



v7.10

Vijeo Citect 用户指南

November 2008

法律声明

免责条款

Schneider Electric SA 不对本手册以及法律所允许的最大延伸做任何陈述或担保，将其因违反任何可能默示的保证之责任，明确限制为对本手册予以更换。此外，Schneider Electric SA 保留在不承担任何通知义务的情况下随时修改此出版物的权利。

版权

© Copyright 2008 Schneider Electric SA 保留所有权利。

商标

Schneider Electric SA 已经尽最大努力提供了在本手册中提及的有关公司名称、产品和服务的商标信息。

Citect、CitectHMI 和 CitectSCADA 是 Citect Pty.Ltd. 的注册商标。

IBM、IBM PC 和 IBM PC AT 是国际商业机器有限公司的注册商标。

MS-DOS、Windows、Windows NT、Microsoft 和 Excel 是 Microsoft Corporation 在美国和/或其他国家/地区的注册商标或商标。

DigiBoard、PC/Xi 和 Com/Xi 是 Digi International Inc. 的商标。

Novell、Netware 和 Netware Lite 是 Novell, Inc. 在美国和其他国家的注册商标或商标。

dBASE 是 dataBased Intelligence, Inc. 的商标。

本文档中提及的其他商标和产品都是其相应所有者的商标或注册商标。

一般声明

本手册中使用的一些产品名称仅用于识别的目的，它们可能是其相应公司的商标。

November 2008 版，适用于 Vijeo Citect 版本 v7.10

手册修订版本 v7.10。

请注意

只有合格的技术人员才能安装、操作、维修和维护电气设备。Schneider Electric SA 对使用本材料所产生的任何后果概不负责。© 2008 Schneider Electric SA。保留所有权利。

有效性注意事项

本文档供负责实施、操作和维护所述产品的合格技术人员使用。它包含产品正确使用所必需的信息。尽管如此，对于希望更“高级”使用我们产品的技术人员，可能会发现很有必要咨询我们离您最近的分销商，以便获取其他信息。

本文档的内容不是合同规定的保修条款，不得将其解释成保修合同条款的延伸或限制。

有关更多信息，请联系您本地的 Schneider Electric 代表。

内容

法律声明	2
入门.....	23
安全信息	24
危险类别与特殊符号.....	24
请注意	25
开始前的准备工作	25
Chapter: 1 获得技术支持.....	1
联络信息.....	1
Vijeo Citect v7.x 中的新增功能	1
Chapter: 2 Vijeo Citect v7.0 中的新增功能	2
迁移工具.....	2
集群	2
本地变量.....	3
发行警报属性.....	3
内存模式.....	3
客户端在线更改	3
发行者-订阅者模式.....	4
双向网络支持.....	4
基于工程的网络配置.....	4
Vijeo Citectv7.10 中的新增功能.....	4
Vijeo Citect 安全性增强功能	5
Windows 集成安全性.....	6
多签名支持	6
在 Microsoft® Excel 中编辑 .dbf 文件.....	6
增强的驱动程序安装.....	6
新增图形按钮字体选择功能	7

Microsoft Windows Vista 支持	7
配置文件和工程文件的新位置	7
报警字段增强功能	8
新时间同步服务	10
v7.10 中的 Citect.ini 参数	10
新参数	10
不再使用的参数	15
v7.10 中的 Cicode 函数	19
新函数	19
已不再使用的函数	24
修改的函数	25
v7.10 中的 CtAPI 函数	29
已不再使用的函数	29
版本v7.10 中的内核命令	30
Chapter: 3 升级到 Vijeo Citect v7.10	31
极少使用和删除的功能	31
升级过程	31
升级 CTAPI 应用程序	32
配置 I/O 设备	32
运行 Citect 安装程序	33
启动Vijeo Citect	33
迁移工具	34
定义集群	38
配置网络地址	38
配置服务器	38
配置标签以使用簇	39
编译工程	41
运行计算机设置向导	41
故障排除	41
Chapter: 4 关于 Vijeo Citect	43
配置 Vijeo Citect 工程	43
部署 Vijeo Citect	43
运行工程	43
Chapter: 5 工具	44
配置工具	44
运行时工具	45
驱动程序	45

Chapter: 6	工程的组件	47
	图形组件	47
	标签	48
	报警	48
	系统组件	49
	通讯组件	50
	I/O 服务器组件	50
	Cicode / CitectVBA	51
Chapter: 7	典型系统方案	52
	单机系统	53
	分布式 I/O 系统	53
	客户端-服务器系统	54
	冗余服务器系统	55
	集群式控制系统	56
	冗余和分布式控制系统	57
	负载分担系统	58
	使用 Vijeo Citect	60
Chapter: 8	规划工程	61
	工厂的物理布局	61
	操作要求	62
	架构	62
	安全性	62
	可靠性	62
	监视	63
	数据收集	63
	工程设计	63
	命名标准	63
	页面模板	64
	精灵和超级精灵	64
	集群	64
	包含工程	65
	冗余	66
	构建工程	66
	工程	66
	设置 I/O 设备通信	66
	图形组件	67
	报警	67
	数据收集	67

用户和区域.....	68
系统组件	68
部署	69
将 Vijeo Citect 设置为 OPC 数据源.....	69

Chapter: 9 管理工程 70

管理工程.....	70
创建工程.....	70
编辑现有工程的属性.....	72
复制工程.....	74
打印工程详细信息.....	75
删除工程.....	76
链接工程.....	77
时间同步.....	77
时间同步对话框.....	78
归档工程.....	79
备份工程.....	79
备份INI文件	80
配置具有密码加密的备份	81
从命令行运行备份.....	81
恢复工程.....	83
包含工程.....	84
在当前工程中包含另一个工程.....	85
包含工程.....	85
使用工程编辑器.....	86
设置工程编辑器选项.....	86
粘贴标签对话框.....	88
粘贴函数对话框.....	89
查找用户函数对话框.....	89
在工程中使用查找和替换功能.....	89
“查找和替换”对话框.....	90
指定搜索范围.....	91
使用结果列表.....	91
删除结果.....	92
导出结果.....	92
跳转到结果（转到）.....	93
替换结果.....	93
查找与替换提醒信息.....	94
故障排除搜索.....	96

Chapter: 10 保护工程 97

概述.....	97
---------	----

保护顶级工程.....	98
保护包含工程.....	99
将工程变为只读.....	101
工程的只读权限.....	102
保护运行时计算机.....	105
客户端启动限制.....	105
运行客户机作为命令解释程序.....	106
禁止 Windows 键盘命令.....	106
禁止控制菜单命令.....	106
删除“取消”按钮.....	106
Chapter: 11 使用 Vijeo Citect 安全性.....	107
维护用户记录.....	107
用户记录和项目恢复.....	107
添加用户记录.....	108
附加字段.....	109
定义用户权限.....	110
使用等级权限.....	112
定义区域.....	112
将区域用于安全性.....	114
使用标签命名区域.....	115
使用区域组.....	115
组属性.....	116
使用带权限的区域.....	117
指定安全要求.....	118
查看工厂区域.....	119
使用 Vijeo Citect 与 Windows 安全性集成.....	120
添加组和用户.....	121
添加角色.....	122
附加字段.....	123
方案和使用情况.....	124
Chapter: 12 配置您的系统.....	125
运行计算机设置向导.....	125
工程配置.....	126
计算机角色配置.....	126
网络模式.....	127
互连网服务器配置.....	127
报警配置.....	128
报表配置.....	129
趋势配置.....	130
CPU 配置.....	130

事件配置.....	131
启动函数配置.....	132
集群连接配置.....	132
控制菜单安全配置.....	133
键盘安全配置.....	134
其它安全配置.....	134
常规选项设置.....	134
完成.....	135
Chapter: 13 实施集群	136
集群的规则.....	136
集群定义.....	138
网络地址定义.....	138
警报服务器定义.....	138
报表服务器定义.....	139
趋势服务器定义.....	140
I/O 服务器定义.....	141
在运行时为集群分配标签.....	142
Chapter: 14 在系统中建立冗余.....	144
I/O 服务器冗余.....	144
I/O 设备提升.....	145
冗余和持续.....	147
数据路径冗余.....	148
多重设备冗余（备用数据路径）.....	149
网络冗余.....	151
配置网络冗余.....	152
报警、报表和趋势服务器冗余.....	153
报警服务器冗余.....	154
报表服务器冗余.....	155
趋势服务器冗余.....	155
文件服务器冗余.....	155
FTP服务器冗余.....	156
单机系统的冗余.....	157
Chapter: 15 I/O 设备通讯.....	158
I/O 服务器的角色.....	159
I/O 设备的角色.....	159
传输媒介的角色.....	160
协议的角色.....	161
通讯配置.....	162

设置通信.....	164
准备设备	164
准备要通信的 I/O 服务器	165
COMX 驱动程序特殊选项参考	167
TCP/IP 驱动程序特殊选项参考	169
创建通信测试工程.....	171
使用通讯快速向导.....	172
快速通信向导 - 简介	173
快速通信向导 - 服务器选择	173
快速通信向导 - 设备选择.....	173
快速通信向导 - I/O 设备类型.....	173
快速通信向导 - I/O 设备通讯选择	173
快速通信向导 - TCP/IP 地址	174
快速通信向导 - I/O 设备地址.....	174
快速通信向导 - I/O 设备连接调度	174
快速通信向导 - 链接到外部数据库	176
快速通信向导 - 串行设备.....	177
快速通信向导 - 摘要	178
手动配置通信.....	178
I/O 服务器属性.....	178
板卡属性	179
端口属性	180
I/O 设备属性	181
使用设备驱动程序.....	186
确定要使用的驱动程序.....	187
安装驱动程序包.....	188
使用驱动程序参考帮助.....	188
使用 Citect.ini 参数自定义通讯.....	189
使用磁盘 I/O 设备	189
磁盘 I/O 设备的设定	190
冗余的磁盘（虚拟）I/O 设备.....	191
使用内存模式	192
故障排除设备通讯.....	193
错误消息.....	194
协议特定错误	197
调试 I/O 设备和协议.....	200
调试 COMx 驱动程序	200
调试 TCP/IP 驱动程序.....	204
调试使用串行通讯的协议驱动程序.....	205
调试专用板卡驱动程序.....	206
串口回环测试.....	207
联络技术支持部门.....	208
性能考虑.....	209

缓存数据	209
将寄存器分组	210
在 I/O 设备中重映射变量	211
重映射示例	214
高级驱动程序信息	216
变量（数字）限制	216
验证为基于标签的驱动程序分发的工程数据	217
写入延迟效果	217
使用远程设备通过调制解调器通讯	219
I/O 服务器端调制解调器	219
I/O 设备端调制解调器	220
I/O 服务器端的调制解调器配置示例	221
I/O 设备的多路限制	224
配置多路远程 I/O 设备	225
可拨号远程 I/O 设备的 I/O 服务器冗余	228
可拨号远程 I/O 设备通讯的故障排除	229
永久连接的选择方式（向后兼容）	230
调度通讯	230
指定时间表	230
写入调度的 I/O 设备	231
从调度的 I/O 设备进行读取	232
Chapter: 16 标签过程变量	234
标签命名	234
标签名称语法	235
使用结构化标签名称	235
配置本地变量	240
配置变量标签	241
变量标签属性	242
格式化数字变量	246
使用数组	249
Chapter: 17 链接、导入和导出标签	253
链接标签	253
断开与外部数据源的链接	254
删除 I/O 设备	254
导入标签	255
导入标签变量对话框属性	256
Unity 链接标签导入	258
OPC 数据访问服务器标签浏览器	259
OPC 工厂服务器 (OFS) 标签导入	260
OPC 工厂服务器 (OFS) 标签导入 限制	261

TagGen XML 模板	262
导出标签	272
导出标签变量属性	273
外部数据源	274
格式文件	275
格式文件布局	276
域转换	279
识别格式文件	284
联合支持矩阵	286
导入的标签	286
导出的标签	288
链接的标签	289
Chapter: 18 定义和绘制图形画面	291
创建新图形画面	291
新建对话框	291
使用页面	292
用户模板（新画面/模板）对话框	293
打开/另存为对话框	294
使用页面模板	295
选择画面风格	295
链接模板	296
创建自己的模板	296
新建风格对话框	297
使用浏览顺序	297
指定启动画面	298
调整页面大小	298
画面（屏幕）分辨率	299
运行期间页面的大小	299
定义画面属性	300
页面属性—通用	300
页面属性—外观	302
画面属性 - 键盘命令	303
页面属性 - 事件	305
页面属性—环境	306
设置缺省页面设置	306
画面默认值	307
了解绘图环境	308
网格	308
网格设置对话框	308
引导线	309
引导线设置对话框	310
选项	310

颜色	311
“编辑收藏颜色”对话框	312
交换颜色对话框	314
“调整颜色”对话框	315
缩放	316
使用库	317
使用符号	318
位图	319
“导入”对话框	321

Chapter: 19 使用对象 322

使用组	323
重塑对象	323
重塑线条对象	323
使用位图	324
导入图形	324
对象属性	324
外观	324
移动	325
缩放	325
填充	325
输入	326
滑钮	326
访问	326
操作对象	327
选取对象	327
移动对象	328
改变对象大小	328
删除对象	329
锁定/解锁对象	329
将对象分组	330
复制和粘贴对象	330
改变对象的重叠	331
对齐排列对象	332
旋转对象	333
镜像对象	334
定位对象	334

Chapter: 20 了解对象类型 335

手画线对象	335
手画线属性 - 外观（一般）	336
直线对象	337

直线属性 - 外观 (通用)	337
长方形对象	338
长方形属性 - 外观 (通用)	338
椭圆对象	341
椭圆属性 - 外观(通用)	342
多边形对象	346
多边形属性 - 外观 (通用)	347
管道对象	350
管道属性 - 外观 (通用)	351
文本对象	351
文本属性 - 外观 (通用)	352
文本属性 - 外观 (显示数值)	353
数字对象	357
按钮对象	358
按钮属性 - 外观(通用)	358
符号集对象	360
符号集属性 - 外观通用 (开/关)	361
符号集属性 - 外观通用 (多状态)	362
符号集属性 - 外观通用 (阵列)	363
符号集属性 - 外观通用 (动画)	364
趋势对象	365
趋势属性	366
插入趋势对话框	368
Cicode 对象	368
Cicode 对象属性 - Cicode (通用)	369
使用动点	369
粘贴的符号对象	370
粘贴符号对话框	370
符号属性 - 外观 (通用)	371
粘贴的精灵对象	372
ActiveX 对象	372
管理关联数据源	372
ActiveX 对象属性	373
标签关联	373
ActiveX 对象属性 - 外观 (标签关联)	373
对象标识	375
对象属性 - 访问 (标识)	375
数据库交换控制对象	376
Vijeo Web Gate 控制对象	376

Chapter: 21 定义通用对象属性..... 377

3D 效果	377
对象属性 - 外观 (3D 效果)	377

可见性.....	380
对象属性 - 外观 (可见性)	381
移动.....	381
对象属性 - 移动 (水平).....	382
对象属性 - 移动 (垂直)	383
对象属性 - 移动 (旋转).....	384
组和对象运动 - 范例	386
缩放.....	388
对象属性 - 缩放 (水平)	388
对象属性 - 缩放 (垂直)	391
填充颜色.....	393
对象属性 - 填充颜色 (开/关)	394
对象属性 - 填充颜色 (多状态)	395
对象属性 - 填充颜色 (阵列)	397
对象属性 - 填充颜色 (阈值)	399
对象属性 - 填充颜色 (梯度)	401
填充高度.....	403
对象属性 - 填充 (高度)	403
触击命令.....	406
对象属性 - 输入 (鼠标)	407
键盘命令.....	408
对象属性 - 输入 (键盘命令)	409
滑钮.....	410
对象属性 - 滑动 (水平)	411
对象属性 - 滑动 (垂直)	412
对象属性 - 滑动 (旋转)	413
访问.....	414
对象的通用访问	414
对象属性 - 访问 (通用)	415
禁止访问对象	416
对象属性 - 访问 (禁用)	417

Chapter: 22 定义命令和控件..... 418

触击命令.....	418
键盘命令.....	418
滑钮控制.....	419
系统键盘命令.....	419
系统键盘命令属性.....	419
键盘键码.....	420
键盘键码属性	420
键盘.....	421
为命令定义键序列.....	422
使用热键.....	423

使用变量数据输入.....	424
传递多个参数.....	425
向函数传递键盘参数.....	426
Chapter: 23 配置和处理报警.....	428
配置报警.....	428
使用报警延迟.....	429
使用自定义报警过滤器.....	429
实施使用自定义报警过滤器的查询.....	429
效率注意事项.....	431
报警分类.....	431
报警分类属性.....	432
数字量报警.....	436
数字量报警属性.....	437
多数字量报警.....	440
多数字量报警属性.....	441
带时间标签的报警.....	446
带时间标签的报警属性.....	446
模拟量报警.....	450
模拟量报警属性.....	450
高级报警.....	455
高级报警属性.....	455
带时间标签的数字量报警.....	457
带时间标签的数字量报警属性.....	458
带时间标签的模拟量报警.....	461
带时间标签的模拟量报警属性.....	461
设置报警显示格式.....	466
包括Vijeo Citect数据.....	466
包括固定文字.....	467
显示列表和表.....	467
报警信息中的变量数据.....	468
报警显示字段.....	468
报警汇总字段.....	471
改变报警摘要的显示顺序.....	473
将报警属性作为标签使用.....	473
支持的报警属性.....	473
报警属性的写入.....	476
设置报警属性.....	478
处理运行时的报警.....	478
使用系统字体.....	479
字体属性.....	480
配置自定义颜色字体.....	482

Chapter: 24	配置事件	483
	事件属性.....	483
	运行事件.....	484
	指定时间和周期.....	485
	使用触发器.....	486
Chapter: 25	使用累积器.....	487
	累积器属性.....	487
Chapter: 26	记录和趋势化数据.....	490
	趋势化数据	490
	配置趋势标签	491
	趋势标签属性	491
	趋势图表.....	497
	创建趋势画面	498
	趋势内插	499
	打印趋势数据.....	499
	导出趋势数据.....	500
	使用趋势历史文件.....	501
	存储方式	502
	计算磁盘存储	502
	重新配置历史文件.....	504
	使用路径替换.....	504
	缺省路径定义	505
	调试趋势.....	505
Chapter: 27	了解统计过程控制.....	507
	过程变差.....	507
	统计控制.....	508
	过程性能.....	509
	XRS 控制图表	509
	CL、UCL 和 LCL	510
	解释图表	510
	性能图表.....	511
	USL 和 LSL.....	511
	Cp 指数	511
	Cpk 指数	511
	Pareto 图	511
	使用统计过程控制 (SPC).....	512
	SPC 标签.....	512
	SPC 标签属性.....	513

SPC 控制图表	518
XRS 控制图表	519
配置 XRS 图表	519
性能图表	519
配置性能图表	519
Pareto图	520
配置排列图	520
SPC 报警	520
SPC 公式和常量	522
控制图线条常量	525
Chapter: 28 报表信息	528
配置报表	528
报表对话框	529
运行报表	530
启动时运行报表	531
指定时间和周期	531
使用触发器	532
使用命令	532
报表格式文件	533
报表示例	535
处理报表中的通信错误	536
报告 I/O 设备数据错误	536
压缩报表	537
Chapter: 29 使用标签	538
在标签中使用参数	540
将数值转换为字符串	541
替换字符串	542
定义标记符	543
Chapter: 30 使用设备	544
使用设备组	546
使用设备读取数据	546
配置设备	547
格式化设备中的数据	551
打印机和 ASCII 设备格式	552
dBASE 和 SQL 数据库设备格式	553
使用数据库设备	555
使用设备历史文件	558
使用命令字段	560

关于打印管理.....	561
Chapter: 31 与其他应用程序交换数据	562
使用 DDE (动态数据交换)	562
DDE 会话和客户机语法.....	563
设置 DDE 会话	564
DDE 函数类型	565
通过 DDE 交换数据.....	566
使用 DDE 连接到标签数据库	566
使用 DDE 公布特定数据.....	567
将值写入 DDE 应用程序.....	568
读取 DDE 应用程序参数值.....	569
将 DDE 用于 Microsoft Office 应用程序	569
网络 DDE.....	570
启动网络 DDE 服务.....	570
设置网络 DDE 共享.....	572
DDE 共享.....	572
使用 DDE 受信任共享.....	573
使用网络 DDE	573
连接到网络 DDE 共享程序.....	574
使用 Citect 标签 Excel 宏	575
使用外部数据库	576
dBASE 数据库.....	576
SQL 数据库	577
使用结构化查询语言.....	578
连接到 SQL 数据库	578
执行 SQL 命令	579
使用事务	580
在 SQL 中表示日期和时间.....	581
使用 ODBC 驱动程序	581
安装 ODBC 驱动程序.....	582
关于 ODBC 驱动程序	583
设置 ODBC	583
获得正确的 ODBC 语法.....	584
ODBC 的编程风格	584
比较 DDE 与 ODBC	585
ODBC 兼容性	585
将 Vijeo Citect 用作 ODBC 服务器.....	587
使用 ODBC 读取来自访问表的数据.....	589
使用 ODBC 附加数据	590
使用 ODBC 编辑数据	590
删除“访问”表中的行	591
通过 ODBC 调用动作查询	591

参数查询	592
Access 和 Cicode 日期/时间转换	593
使用 Microsoft Excel 编辑 .dbf 表	593
功能	595
Chapter: 32 使用精灵和超级精灵	597
了解精灵	597
创建精灵	598
打开精灵	599
保存精灵	599
为精灵定义替换	599
使用精灵	600
粘贴精灵对话框	601
精灵属性	602
在模板中使用精灵替换	602
使用超级精灵	602
为超级精灵定义替换	605
不通过精灵使用超级精灵	607
通过超级精灵使用常量和数组	607
创建精灵控制器	609
“绑定超级精灵”对话框	610
“选择超级精灵”对话框	610
嵌套超级精灵	610
超级精灵区域	611
超级精灵环境变量	611
通过精灵和超级精灵使用结构化标签名称	611
通过精灵使用结构化标签	612
通过超级精灵使用结构化标签	613
隐藏图形对象	614
IFDEF 宏	614
Chapter: 33 使用多国语言工程	616
改变语言	616
标记要改变语言的文字	617
语言数据库	618
多语言	619
多工程	619
在运行期间改变语言	619
用不同语言来记录数据	620
ASCII 和 ANSI 字符集	620
OEM 字符集	621

Chapter: 34	使用 OPC 服务器 DA2.0	622
	OPC 概述.....	622
	Vijeo Citect OPC 服务器.....	623
	Vijeo Citect OPC 服务器安装	623
	配置对 OPC 服务器的远程访问	626
	配置 OPC 服务器.....	626
	配置 OPC 客户机.....	630
	创建数据组.....	634
	向组中添加数据项.....	634
	故障排除.....	634
Chapter: 35	构建工程	635
	编译工程.....	635
	增量编译	636
	调试编译	636
	编译选项	637
	编译错误属性	637
	编译错误消息	638
	运行系统.....	643
	启动和运行时配置.....	643
	独立运行服务器.....	644
	通过 Internet 运行系统.....	644
	互联网显示客户端.....	645
	互联网服务器	645
	启动和运行时配置.....	646
	服务器 - 客户端文件更新.....	646
	使用备用 INI 文件	648
	调试运行期系统	648
	硬件报警	648
	SysLog.DAT 文件.....	649
	使用地址转发进行服务器重定向.....	650
	在线重新启动系统.....	651
	在线重新启动网络上的系统	651
	软件保护.....	654
	CiUSAFE 对话框属性	654
	演示模式	655
	使用内核.....	656
	显示内核窗口	657
	内核内部	658
	使用内核命令	660
	内核命令.....	661
	收集 Runtime 信息	684

系统调整	684
使用 Web 客户机	686
Chapter: 36 Web 客户端	687
系统架构	687
入门	688
选择要使用的 Web 服务器平台 - IIS 或 Tomcat	689
准备用于部署的项目	689
Web 客户机平台的功能限制	690
准备要交付的工程用户文件	691
运行“Web 部署准备”工具	692
配置部署	693
创建新部署	694
从 Vijeo Citect 内部部署工程	696
显示部署	698
编辑现有部署	698
更新部署以反映工程更改	699
删除部署	699
实现多语言支持	700
如何设置缺省语言	700
使用不同于当前系统区域设置的语言	701
设置非缺省语言	702
Web 客户机升级注意事项	703
常见问题	704
Chapter: 37 Windows 语言代码	709
使用 CSV_Include 工程	712
Chapter: 38 CSV_Include 简介	713
在何处查找信息	713
Chapter: 39 使用页面和模板	714
普通画面模板	715
报警画面模板	715
通用功能	716
趋势画面模板	716
通用功能	717

文件画面模板.....	720
管理工具页面模板.....	720
通用工具栏.....	721
漫游工具条.....	722
报警工具栏.....	723
Chapter: 40 创建新工程.....	725
创建特权用户.....	725
运行计算机设置向导.....	726
事件设置画面.....	726
安全设置—控制菜单页面.....	726
设置瞬时趋势.....	726
在多个监视器上显示工程.....	727
实现声音报警.....	727
创建画面.....	728
创建新画面.....	729
向页面中添加用户帮助.....	729
创建自定义菜单.....	730
菜单配置工具.....	731
构建自定义菜单.....	732
编辑项目.....	732
创建报警组.....	733
创建趋势组.....	733
使用环境变量.....	734
词汇表.....	735
索引.....	766

入门

本部分包含有关 Vijeo Citectv7.10 的信息，并介绍以下内容：

[获得技术支持](#)

[升级到 Vijeo Citect v7.10](#)

[关于 Vijeo Citect](#)

[工具](#)

[工程组件](#)

[典型系统方案](#)




安全信息

危险类别与特殊符号

下述符号和特殊消息可能出现在本手册中或出现在产品上，以对潜在的危险进行警告或提醒对阐释或简化某个过程的信息的注意。

“危险”或“警告”安全标记符中的闪电或 ANSI man 符号表示电气危险，正如下述所表明
的，如果不遵守其中的说明，可能会导致人身伤害。

手册的安全信息中的感叹号符号表示潜在的人身伤害危险。谨遵此符号所引出的所有安全
信息，以避免可能的人身伤害或死亡。

符号	名称
	闪电
	ANSI man
	感叹号



危险表示存在严重的危险情况，如果忽视可导致死亡或严重的人身伤害。



警告表示存在潜在的危险情况，如果忽视可导致死亡或严重的人身伤害。

**小心**

小心表示存在潜在的危险情况，如果忽视可导致轻微或中度人身伤害。

小心

小心（不带安全提示符号）表示潜在的危险情况，如果忽视可能导致财产损失。

请注意

只有合格的技术人员才能安装、操作、维修和维护电气设备。Schneider Electric SA 对使用本材料所产生的任何后果概不负责。

开始前的准备工作

Vijeo Citect 是一个数据采集与监控 (SCADA) 解决方案。它便于创建软件以管理和监控工业系统及处理。由于 Vijeo Citect 在控制系统及处理中的核心作用，在操作设置中实施 Vijeo Citect 工程之前，您必须对它进行相应的设计、授权和测试。遵循下列信息：

**警告****失去控制**

- 任何控制方案的设计者必须考虑到潜在的控制路径失效问题，对某些关键的控制功能，还需要提供保证在路径失效过程中或之后实现安全状态的方法。例如，紧急制动和超程制动等关键控制功能。
- 必须为关键控制功能提供单独或冗余控制路径。
- 系统控制路径可以包含通信链接。必须对暗含的无法预料的传输延迟或链接失效问题加以考虑。*
- 对于使用 Vijeo Citect 创建的各个控制系统的实施，在投入使用之前必须逐一进行全面测试以保证工作正常。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

* 有关其他信息，请参考 NEMA ICS 1.1（最新版）中的“应用、安装和维护固态控制的安全原则”。

Chapter: 1 获得技术支持

以下多个支持选项能帮您获得最多的产品信息。

- 如果有关于使用 Vijeo Citect 的问题，请从内容丰富的在线帮助中寻找答案。您可以使用**内容**列表查找感兴趣的部分，向**索引**中输入项目，或使用**搜索**标签输入项目。
- 如果希望得到在线帮助中提供的更多技术信息，请查看知识库。

如果无法找到所需信息，请联系本地 Schneider 电子支持办事处。

另请参见
[联系信息](#)

联络信息

欲获得贵地区的联络信息，请访问如下网址：

<http://www.schneider-electric.com/>

Vijeo Citect v7.x 中的新增功能

本节描述 v7.10 的新增 Vijeo Citect 功能和增强功能。为了保持连续性，本节同时描述在 Vijeo Citect v7.0 发行版中新添的功能。

[在 v7.0 中引入](#)

[在 v7.10 中引入](#)

有关以下内容的变化，请参见：

- Citect.ini 参数，请参见 [版本 7.x 中的 Citect.ini 参数](#)。
- Cicode 函数，请参见 [版本 7.x 中的 Cicode 函数](#)。
- CtAPI 函数，请参见 [版本 7.x 中的 CtAPI 函数](#)。
- 内核命令，请参见 [版本 7.x 中的内核命令](#)。

有关如何配置一个现有的工程以便在 v7.10 中运行的详细信息，请参见 [升级到 V7.10](#)。

Chapter: 2 Vijeo Citect v7.0 中的新增功能

Vijeo Citect v7.0 包含下列新功能。

在 v7.0 中引入：

- [迁移工具](#)
- [集群](#)
- [本地变量](#)
- [发行警报属性](#)
- [内存模式](#)
- [客户端在线更改](#)
- [发行者-订阅者模式](#)
- [双向网络支持](#)
- [基于工程的网络配置](#)

有关以下内容的变化，请参见：

- Citect.ini 参数，请参见 [版本 7.x 中的 Citect.ini 参数](#)。
- Cicode 函数，请参见 [版本 7.x 中的 Cicode 函数](#)。
- CtAPI 函数，请参见 [版本 7.x 中的 CtAPI 函数](#)。
- 内核命令，请参见 [版本 7.x 中的内核命令](#)。

有关如何配置一个现有的工程以便在 v7.10 中运行的详细信息，请参见 [升级](#)。

另请参见

[Vijeo Citect v7.10 中的新增功能](#)

迁移工具

迁移工具是单独的应用程序，应在执行自动更新后手动运行，并在用户为最后的迁移准备好工程后由用户初始化。此工具会提供版本 7.0 中结合的工程功能中的重要更改。

另请参见

[迁移工具](#)

集群

通过集群，可以将不同的运行期间组件组合在一个工程中，从而能够监控多个独立的系统。

在配置集群式系统的方式上，可以采用无数变体。最适当的配置将取决于要部署的解决方案的要求以及部署它的环境。有关详细信息，请参见 [典型系统方案](#)。

另请参见
[包含工程
实施集群](#)

本地变量

当您启动运行时系统时，可通过本地变量在内存中存储数据。本地变量在每次启动系统时创建，因此当您关闭系统时本地变量不保留它们的值。

当您需要每个过程具有单独的数据副本时，本地变量非常有用。对于在工程中配置的每个本地变量，每个过程都有其自己的副本，而且本地变量中的值仅对写入这些值的过程可用。

另请参见
[配置本地变量](#)

发行警报属性

警报设备是指其“协议”字段被设置为“警报”的设备。这些设备的功能可以通过警报服务器进行配置，方法是将“发行警报属性”的属性设置为 True。

另请参见
[警报服务器定义](#)

内存模式

I/O 设备现在可以配置为以内存模式运行：

- 以内存模式运行的 I/O 设备在内存中创建，其值在运行时存储于内存中。
- 采用内存模式的设备不连接到任何硬件，并且将其值写入缓存。第一次配置系统时，内存模式非常有用，因为可以在连接物理 I/O 设备之前设计和测试系统。

注意：内存模式已替代不再支持的内存 I/O 设备。配置为内存 I/O 设备的设备在升级到 v7.10 过程中将被转换为本地变量。

另请参见
[使用内存模式](#)

客户端在线更改

现在，可以在不重新启动客户端的情况下对工程进行下列实时更改：

- I/O 设备（重新启动 I/O 服务器）
- 标签（重新启动 I/O 服务器）
- 报警（重新启动报警服务器）
- 趋势（重新启动趋势服务器）

- 报表（重新启动报表服务器）
- 累积器（重新启动报表服务器）

客户端只需要为它们部署图形、代码和通讯配置。其他配置信息部署到适当的服务器即可。

工程更改的部署仍与以前一样

- 手动
- 运行/备份运行/复制
- FTP (IDC)
- HTTP（Web 客户端）

发行者-订阅者模式

Vijeo Citect 现在使用发行者-订阅者数据收集模式。客户端计算机订阅已配置好的标签，并在标签参数值更改时接收通知。也可以通过更改某个标签，避免轮询的需要，并改善系统效率来触发 Cicode 函数。

另请参见

[TagSubscribe](#)

[TagUnsubscribe](#)

双向网络支持

之前的 Vijeo Citect 版本已通过 NetBIOS 支持冗余网络。从 7.0 版开始，用户就可以仅使用 TCP/IP 对网络进行配置，并可以为每个服务器指定多个 IP 地址，对网络冗余提供本地支持。

基于工程的网络配置

在版本 7.0 中，工程中嵌入了工程拓扑，可以从工程编辑器执行网络配置。服务器及其 IP 地址是通过工程编辑器中的**网络地址**对话框设置的。

这意味着可以轻松更改系统中的物理计算机。只要新计算机的 IP 地址或计算机名与被替换的计算机相同，新计算机将可以立即充当相同角色。

Vijeo Citectv7.10 中的新增功能

Vijeo Citectv7.10 包含下列新功能。

在 v7.10 中引入：

- [Windows® 集成安全性](#)
- [Vijeo Citect 安全性增强](#)
- [多签名支持](#)
- [在 Microsoft® Excel 中编辑 DBF 文件](#)

- [增强的驱动程序安装](#)
- [新增图形按钮字体选择功能](#)
- [Microsoft® Windows Vista™ 支持](#)
- [配置文件和工程文件的新位置](#)
- [新报警字段增强功能](#)
- [新时间同步服务](#)

有关以下内容的变化，请参见：

- Citect.ini 参数，请参见 [版本 7.x 中的 Citect.ini 参数](#)。
- Cicode 函数，请参见 [版本 7.x 中的 Cicode 函数](#)。
- CtAPI 函数，请参见 [版本 7.x 中的 CtAPI 函数](#)。
- 内核命令，请参见 [版本 7.x 中的内核命令](#)。

有关如何配置一个现有的工程以便在 v7.10 中运行的详细信息，请参见 [升级到 v7.10](#)。

另请参见

[Vijeo Citect v7.0 中的新增功能](#)

Vijeo Citect 安全性增强功能

Vijeo Citect v7.10 发行版包括旨在降低产品对外部网络威胁安全性暴露的更改。下面详述受到影响的产品功能。请检查该列表以了解升级和设计过程对您的系统有何影响。

管理表面区域

添加了一组新的配置参数，以提供对 Vijeo Citect 网络接口的控制。这些参数允许您对产品中未使用的功能进行控制，从而帮助您保护系统。可以启用/禁用下列服务：DDE、远程 CTAPI、ODBC、OLEDB 和 FTP。这些服务在缺省情况下被禁用。

控制操作要求用户登录

现在要求对用户进行配置并且用户需要登录 Vijeo Citect 才能让显示过程执行标签写入（控制）操作。在设计 Vijeo Citect 工程时应避免 Cicode 任务出现在执行标签写入（非用户发布）的显示过程中。

我们建议，在配置工程时应尽可能提高系统安全性保护。如果系统已经具有网络安全性保护而且不需要其他安全性保护，则可使用下列参数关闭保护功能以避免受更改的影响：

客户端/显示节点的参数：[\[LAN\]SecureLogin](#)

服务器节点的参数：[\[LAN\]AllowLegacyConnections](#)

如果在运行本系统的节点中混有 Vijeo Citect 的新、旧版本，则升级过程可能需要这些参数。

另请参见

[系统 参数](#)

Windows 集成安全性

在 Vijeo Citect 中，现在您能够将 Vijeo Citect 的用户和安全性选项结合到标准的 Windows 安全系统中。如果更愿意在工程中定义用户并登录到 Vijeo Citect 运行期，当然仍然能够使用 Vijeo Citect 本地安全性。

使用集成的 Windows 安全功能，Windows 用户可以使用在工程中配置的运行时限登录到 Vijeo Citect 运行时间。

另请参见

[使用 Windows 安全性](#)

多签名支持

利用新的 Cicode 函数 MultiSignatureForm 和 MultiSignatureTagWrite，Vijeo Citect 现在提供的工具可供多达四名用户对某一操作或标签写入操作进行审批。

进一步利用两个 Cicode 函数（VerifyPrivilegeForm 和 VerifyPrivilegeTagWrite），您能够通过特定的权限集限制某个用户对特定操作或标签写入操作的访问。

在 Microsoft® Excel 中编辑 .dbf 文件

Vijeo Citect 允许您编辑和保存 Vijeo Citect 中使用的 .dbf 文件（表），方法是在 Microsoft® Office Excel® 中打开它们。

使用 Microsoft Office Excel® 2007，您无法以 .dbf 的格式保存文件，即使您使用“文件”>“打开”命令进行打开和编辑也是如此。为了克服这一限制，Vijeo Citect 包括了一个针对 Microsoft Excel 的插件，其名称为 ProjectDBFAddIn。在 Excel 中加载了此插件后，您将可以使用正确的格式浏览、打开、编辑并保存 Vijeo Citect .dbf 文件。

另请参见

[使用 Microsoft Excel 编辑 .dbf 表](#)

增强的驱动程序安装

安装早于 v7.10 的 Vijeo Citect 时，所有可用的通信驱动程序都将随产品一起自动安装。从 v7.10 起，这些驱动程序的安装将在产品安装的最后阶段通过一个独立的安装过程完成。利用这个安装过程，您能够选择希望安装的且特定于系统及其 I/O 设备的单个驱动程序。

请注意，有些驱动程序是必须安装的，否则 Vijeo Citect 将无法正常运行。与以前的版本相同，这些驱动程序都将自动安装。

如果您使用的是 Microsoft® Windows Vista™ 操作系统，应该确认您选择安装的所有驱动程序都能够与该操作系统相兼容。如果您选择的任意驱动程序未被识别为与操作系统相兼

容，或已识别为无法与操作系统相兼容，则安装进程将提供一个警告，说明其影响，并允许您取消选择该驱动程序，然后您才能继续进行安装。

注意：如果您选择忽略警告，则将安装驱动程序，但无法保证该驱动程序的正常操作。

在产品安装完成后，如果要安装其他驱动程序，可以随时单独调用通讯驱动程序安装。

新增图形按钮字体选择功能

在以前版本的 Vijeo Citect 中，您无法在图形编辑器中更改按钮上的文本属性，如字体、大小、风格。无法配置按钮文本属性使文本来自不同源对象的图形在同一页面上显示出不同的字体设置，不但不美观不整洁，而且运行时会造成显示不一致。

从 v7.10 开始，可以使用图形编辑器和自动界面对按钮对象上显示的文本进行配置，使其与其他 Vijeo Citect 文本对象的属性保持一致。这使您能够得到一个更加精致且同一的 GUI，从而满足某个工程运行时的显示需求。

在从以前的版本进行迁移时，按钮对象文本属性将被保留，并更改为新的按钮对象文本属性，方法是：将相应的缺省属性参数值自动更改为新的配置，如字体=Arial、大小=12、对齐=居中、风格=常规，等。

另请参见

[按钮属性 - 外观](#)

Microsoft Windows Vista 支持

Vijeo Citectv7.10 已获得 Microsoft®“使用 Windows Vista™”证书。但是，仅符合此证书的要求还不足以使 Vijeo Citect 在 Vista 中完全发挥作用。要在 Vista 操作系统中达到令人满意的功能，还需要进行很多其它更改。

v7.10 也符合“Windows Vista™ 认证”证书的很多要求，通过此级别的限定，我们相信用户会发现在 Vista 操作系统中运行产品与在以前的操作系统中运行产品时的细微差别。

配置文件和工程文件的新位置

根据 Windows Vista 中安全性的变更，对 Vijeo Citectv7.10 使用的配置文件和用户文件的位置做了一些修改。

如果在 Windows XP 或更早版本中安装，配置和工程文件缺省存储在下列文件夹中：Documents and Settings/All Users/Application Data/Schneider Electric/Vijeo Citect 7.10/。如果在 Windows Vista 中安装，配置和工程文件缺省存储在下列文件夹中：ProgramData/Schneider Electric/Vijeo Citect 7.10/。安装位置如下：

文件类型	平台	安装路径
------	----	------

文件类型	平台	安装路径
配置文件（如 citect.ini 文件）	Vista 之前	Documents and Settings/All Users/Application Data/Schneider Electric/Vijeo Citect 7.10/Config
	Vista	ProgramData/Schneider Electric/Vijeo Citect 7.10/Config
用户目录	Vista 之前	Documents and Settings/All Users/Application Data/Schneider Electric/Vijeo Citect 7.10/User
	Vista	ProgramData/Schneider Electric/Vijeo Citect 7.10/User
数据目录	Vista 之前	Documents and Settings/All Users/Application Data/Schneider Electric/Vijeo Citect 7.10/Data
	Vista	ProgramData/Schneider Electric/Vijeo Citect 7.10/Data
日志文件	Vista 之前	Documents and Settings/All Users/Application Data/Schneider Electric/Vijeo Citect 7.10/Logs
	Vista	ProgramData/Schneider Electric/Vijeo Citect 7.10/Logs
所有由驱动程序生成的日志文件都写入名为“Drivers”的子文件夹。		

报警字段增强功能

有两个对报警字段的增强功能：

- 运行时写入自定义报警字段
- 报警摘要字段的更改
- 报警显示字段的更改
- 报警呼叫

运行时写入自定义报警字段

现已能够在运行过程中写入八个自定义报警字段。在以前的发行版中，这些字段事实上仅能用于报警过滤。

报警摘要字段的更改

报警摘要字段现已能够用于对报警显示或报警记录设备进行格式化。另外，除了状态之外，任何报警显示字段都可用在您的报警摘要中。

新报警摘要字段

域名	描述
{SumType,n}	报警摘要的类型（与报警显示“类型”类似）。

报警显示字段的更改

现在，任何报警类型都可以使用所有的显示字段。对于不适用的特殊报警类型，则会显示零或空白字符。

新报警显示字段

域名	描述
{TagEx,n}	带有集群名称前缀的报警标签
{AlarmType,n}	报警类型（字符串），未本地化。
{TypeNum,n}	报警类型编号（改用 AlarmType 获取字符串值）。V
{AlmComment,n}	在报警属性对话框的“注释”字段中输入的文本信息。
{Cluster,n}	集群名称
{CUSTOM1,n} {CUSTOM2,n} {CUSTOM3,n} {CUSTOM4,n} {CUSTOM5,n} {CUSTOM6,n} {CUSTOM7,n} {CUSTOM8,n}	配置的报警自定义字段。
{LocalTimeDate,n}	报警日期和时间。
{Paging,n}	指示是否应该呼叫该报警。
{PagingGroup, n}	指示报警所属的呼叫组。

报警呼叫

Vijeo Citect 报警机制对设备数据进行持续的监测并且在设备出现任何故障或者处于任何报警条件下对操作人员进行提示。如果报警被触发，则会显示在标准报警显示画面上。操作员必须始终坚守在 HMI 之前以监控系统。Vijeo Citectv7.10 提供了报警和操作人员远程呼叫系统的联结机制。

添加了两个报警属性，从而使 Vijeo Citect 能够与任何第三方呼叫系统接口。“呼叫”属性是一个标志，表示改报警即将被呼叫；**PagingGroup** 属性是一个无格式的文本字段，表示在发生报警的事件中将要通知的人员顺序。

有关如何与 Vijeo Citect 接口的信息，请参见第三方呼叫系统的说明文档。

新时间同步服务

为了维持时间同步，Vijeo Citectv7.10 将安装一种称作 TimeSyncService 的 Windows 服务，可在内置 LocalSystem 帐户中运行。此服务将取代与 Windows Vista 不兼容的现有时间同步服务器。本服务的目的在于维持本地计算机上的时间，而不使用其他（一个或多个）时间源。

Vijeo Citect 提供的时间同步实用程序可帮助您配置时间同步，并将服务作为管理环境的一部分进行控制。此实用程序要对 Windows 服务进行配置和控制，因此需要管理员权限。在 Windows Vista 中，如果开启用户访问控制 UAC，运行时系统将会提示您提升到管理员级别。而在较早的操作系统中运行时，如果当前用户在本地计算机上不是管理员，则在显示错误后将退出该实用程序。

另请参见

[时间同步](#)

v7.10 中的 Citect.ini 参数

下列各节详细介绍了对 Vijeo Citectv7.10 中的 Citect.ini 参数所做的改变：

- [新参数](#)
- [不再使用的参数](#)

新参数

以下参数为 7.0 版和 v7.10 中的新参数。有关所有系统参数的完整列表，请参见 帮助文件。

报警参数：

[Alarm.ClusterName.ServerName]Clusters	设置此报警服务器进程在启动时所连接的群集。
[Alarm.ClusterName.ServerName]CPU	设置分配给报警服务器进程的 CPU。
[Alarm.ClusterName.ServerName]Events	此报警服务器进程启用的事件列表
[Alarm.ClusterName.ServerName]ShutdownCode	确定报警服务器进程关闭时所运行的 Cicode 函数。
[Alarm.ClusterName.ServerName]StartupCode	确定报警服务器进程启动时所运行的 Cicode 函数。

备份参数：

[Backup]SaveiniFiles	确定备份时是否缺省选中了“保存 ini 文件”复选框。
----------------------	-----------------------------

客户机参数:

[Client]Clusters	选择客户机在启动时要连接的群集。
[Client]ComputerRole	指定计算机的角色。
[Client]Events	设置客户机上要启用的事件。
[Client]ForceClient	仅启动客户机进程，并使用网络连接连接到配置好的 Citect 服务器。
[Client]FullLicense	指定控制客户机使用完整的许可证
[Client]ShutdownCode	确定 DisplayClient 组件关闭时要运行的 Cicode 函数。
[Client]StartupCode	确定 DisplayClient 组件启动时要运行的 Cicode 函数。
[Client]WaitForConnectAtStartup	指定必须在启动之前先建立与服务器进程的连接，然后才能连接服务器。

CtCicode 参数:

[CtCicode]FastFormat	控制是否在 Cicode 编辑器中使用快速格式化。
----------------------	---------------------------

CtEdit 参数:

[CtEdit]Config	Vijeo Citect 配置文件（如 citect.ini）所在的目录。
[CtEdit]Logs	Vijeo Citect 日志文件所在的目录。

拨号参数:

[Dial]MissedScheduleTolerance	指定在缓存稳定前，丢失了多少连续计划拨号尝试
-------------------------------	------------------------

驱动程序参数:

[<驱动程序名称>]OverriderOSProtection	确定是否在与 Windows Vista 不兼容的驱动程序的 I/O 服务器中覆盖内置的保护机制。
---------------------------------	---

常规参数:

[General]Multiprocess	确定 Vijeo Citect 的运行方式为多进程应用程序，还是单进程应用程序。
-----------------------	--

IO 服务器参数:

[IOServer.ClusterName.ServerName]Clusters	设置 I/O 服务器进程在启动时要连接的集群
[IOServer.ClusterName.ServerName]CPU	设置报警服务器进程分配到 CPU
[IOServer.ClusterName.ServerName]Events	此 I/O 服务器进程启用的事件列表
[IOServer.ClusterName.ServerName]ShutdownCode	确定要在 I/O 服务器进程关闭时运行的 Cicode 函数
[IOServer.ClusterName.ServerName]StartupCode	确定要在 I/O 服务器进程启动时运行的 Cicode 函数

报表参数:

[Report.ClusterName.ServerName]Clusters	设置报表服务器进程在启动时要连接的群集。
[Report.ClusterName.ServerName]CPU	设置报表服务器进程分配到的 CPU
[Report.ClusterName.ServerName]Events	此报表服务器进程启用的事件列表
[Report.ClusterName.ServerName]ShutdownCode	确定报表服务器进程关闭时所运行的 Cicode 函数。
[Report.ClusterName.ServerName]StartupCode	确定报表服务器进程启动时所运行的 Cicode 函数。

趋势参数:

[Trend.ClusterName.ServerName]Clusters	设置趋势服务器进程在启动时要连接的群集
--	---------------------

[Trend.ClusterName.ServerName]CPU	设置趋势服务器进程分配到的 CPU
[Trend.ClusterName.ServerName]Events	此趋势服务器进程启用的事件列表
[Trend.ClusterName.ServerName]ShutdownCode	确定趋势服务器进程关闭时所运行的 Cicode 函数
[Trend.ClusterName.ServerName]StartupCode	确定趋势服务器进程启动时所运行的 Cicode 函数

以下参数为 7.10 版中的新参数。

报警参数:

[Alarm]ArgyleTagValueTimeout	定义报警服务器在启动多状态报警扫描之前等待多状态标签称为可用的时间长度。
------------------------------	--------------------------------------

代码参数:

[Code]HaltOnInvalidTagData	如果启用，则在任何标签读操作返回无效数据时会导致 Cicode 终止。
----------------------------	-------------------------------------

客户端参数:

[Client]AutoLoginMode	设置后可启用自动登录。用户可选择七种模式之一。
-----------------------	-------------------------

CtAPI 参数:

[CtAPI]AllowLegacyConnections	启用时，当前版本的 CTAPI 服务器可接受来自以前版本的 CTAPI 客户端的连接。
[CtAPI]AllowLegacyServices	启用时，Citect Web 服务和 Citect OLEDB 提供商可连接至 CTAPI 服务器。

DDE 参数:

[DDE]AllowCicode	允许 Cicode 通过 DDE Execute 命令在 Citect 服务器上运行。
[DDE]AllowWrites	允许通过 DDE Poke 命令将标签写入 Citect 服务器。

内核参数:

[Kernel]ErrorBuffers	可记录到 syslog.dat 文件的错误缓冲最大数目。
----------------------	------------------------------

LAN 参数:

[LAN]AllowLegacyConnections	设置后可允许以前版本的客户端连接到该服务器。
[LAN]AnonymousLoginName	用于控制客户端缺省登录的名称。
[LAN]SecureLogin	设置为 0 时将禁用安全措施，系统则按早于 7.10 的版本中的行为运行。
[LAN]ServerLoginEnabled	设置后可禁用缺省的服务器登录。
[LAN]ServerLoginName	用于服务器到服务器缺省登录的名称。

ODBC 参数:

[ODBC]Server	设置后可启用 ODBC 连接。
--------------	-----------------

页面参数:

[Page]AllowHScroll	定义水平滚动时的缺省行为。
[Page]AllowHScrollBar	定义显示水平滚动条时的缺省行为。
[Page]AllowVScroll	定义垂直滚动时的缺省行为。
[Page]AllowVScrollBar	定义显示垂直滚动条时的缺省行为。

不再使用的参数

在版本 7.0 中，以下参数已不再受支持：

报警参数：

[Alarm]CPU	设置报警服务器组件被分配到的 CPU
[Alarm]Primary	确定此报警服务器是否为主报警服务器
[Alarm]Process	设置报警服务器组件被分配到的 Vijeo Citect
[Alarm]Server	确定此计算机是否为报警服务器

客户端参数：

[Client]Display	将 Vijeo Citect 计算机设置为控制客户机
[Client]Manager	将 Vijeo Citect 计算机设置为仅查看客户机
[Client]Primary	主 Vijeo Citect 服务器的名称
[Client]Process	设置控制客户机组件被分配到的 Vijeo Citect 进程
[Client]Standby	备用 Vijeo Citect 服务器的名称

代码参数：

[Code]AlarmShutdown	确定要在报警服务器组件关闭时运行的 Cicode 函数
[Code]AlarmStartup	确定要在报警服务器组件启动时运行的 Cicode 函数
[Code]AutoReRead	控制 ReRead() 函数是否被自动调用
[Code]IOServerShutdown	确定要在 I/O 服务器组件关闭时运行的 Cicode 函数
[Code]IOServerStartup	确定要在 I/O 服务器组件启动时运行的 Cicode 函数
[Code]ReportShutdown	确定要在报表服务器组件关闭时运行的 Cicode 函数
[Code]ReportStartup	确定要在报表服务器组件启动时运行的 Cicode 函数

[Code]Shutdown	确定要在控制客户机组件关闭时运行的 Cicode 函数
[Code]Startup	确定要在控制客户机组件启动时运行的 Cicode 函数
[Code]TrendShutdown	确定要在趋势服务器组件关闭时运行的 Cicode 函数
[Code]TrendStartup	确定要在趋势服务器组件启动时运行的 Cicode 函数

DNS 参数:

[DNS]<Server name>	确定主 I/O 服务器的 IP 地址（或者具有完全权限的主机名称）。
--------------------	------------------------------------

事件参数:

[Event]Alarm	要由报警服务器启用的事件类
[Event]IOServer	要由 I/O 服务器启用的事件类
[Event]Name	为要启用的名称条目分配的事件类
[Event]Report	要由报表服务器启用的事件类
[Event]Trend	要由趋势服务器启用的事件类

常规参数:

[General]BadOptimise	确定是否在编译时用标签来替换特定的字符串。
[General]CitectRunningCheck	检查在触发编译操作时工程当前是否正在本地计算机上运行

IOServer 参数:

[IOServer]BlockWrites	确定 Vijeo Citect 是否将阻断对 I/O 设备的优化写入。IOserver 不会阻止写入
[IOServer]CPU	设置报警服务器组件分配到 CPU

[IOServer]Name	缺省 I/O 服务器的名称
[IOServer]Process	设置 I/O 服务器组件被分配到的 Vijeo Citect 过程
[IOServer]SaveBackup	本参数已经被 SaveNetwork 所取代
[IOServer]Server	确定此计算机是否为 I/O 服务器

LAN 参数:

[LAN]Bridge	确定桥级别
[LAN]CancelOnClose	针对在使用 Novell NetBIOS 仿真器时遇到问题的用户
[LAN]Disable	从 LAN 启用/禁止 Vijeo Citect
[LAN]GroupName	确定 Vijeo Citect 是采用"CITECT STATION50"还是采用由 [Lan]Node 参数所指定的计算机名称作为组名。
[LAN]KillPiggyBackAck	控制 Vijeo Citect 是否对支持 piggyback ACK 的网络协议进行优化
[LAN]LanA	定义 Vijeo Citect 所采用的用于 NetBIOS 通讯的协议堆栈
[LAN]NetBIOS	启用/禁止 NetBIOS
[LAN]NetTrace	确定是否在启动时启用 NetBIOS 窗口
[LAN]NetTraceBuff	对跟踪 (trace) 缓冲区的数目进行设置。
[LAN]NetTraceErr	启用 NetBIOS 窗口的错误模式
[LAN]NetTraceLog	启用 NetBIOS 窗口的记录模式
[LAN]Poll	将 Vijeo Citect LAN 通讯置为轮询模式
[LAN]RemoteTimeOut	从某个控制客户端向 I/O 服务器发出的对某个 I/O 设备的写入请求的超时时间
[LAN]Retry	在超时后重试建立通讯的次数 - 在生成提醒信息之前
[LAN]SendTimeOut	跨网络发送网络数据包的超时时间
[LAN]SesRecBuf	Vijeo Citect 对所有会话所使用的接收 NetBIOS 控制数据块 (NetBIOS Control Block, NCB) 的数目。
[LAN]SesSendBuf	Vijeo Citect 对所有会话所使用的发送 NetBIOS 控制数据块的数目。

[LAN]TimeOut	跨网络发送网络数据包的超时时间
--------------	-----------------

代理参数:

[Proxi]<I/O Server name>	定义代理服务器关联列表
--------------------------	-------------

报表参数:

[Report]Primary	确定此报表服务器是否为主报表服务器
[Report]Process	设置报表服务器组件被分配到的 Vijeo Citect 过程
[Report]Server	确定此计算机是不是报表服务器。

服务器参数:

[Server]Name	Vijeo Citect 服务器的名称
--------------	---------------------

趋势参数:

[Trend]BlockByIODevice	确认导致一个趋势标签出现间隙的 I/O 问题不会导致所有趋势标签出现间隙
[Trend]CPU	设置趋势服务器组件被分配到的 CPU
[Trend]Process	设置趋势服务器组件被分配到的 Vijeo Citect 过程
[Trend]Redundancy	启用/禁止趋势冗余动作
[Trend]Server	确定此计算机是不是趋势服务器
[Trend]StaggerRequestSubgroups	通过隔开趋势采样请求降低网络浏览量

Win 参数:

[Win]CtrlEsc

确定在运行期间的系统中是否可以使用 Windows 键盘命令 [Ctrl]+[Esc]（来显示开始菜单）。

Com 参数:

[Com]StartTimeout

确定在对任何数据进行显示之前所有 I/O 设备进入在线状态的等待时间。

v7.10 中的 Cicode 函数

以下各节详细介绍了对 Vijeo Citectv7.10 中的 Cicode 函数所做的改变:

- [新函数](#)
- [已不再使用的函数](#)
- [修改的函数](#)

新函数

以下函数为 7.0 版中的新函数:

其它函数

AccControl	控制累积器，如马达运行时间。
AccumBrowseClose	关闭累积器浏览会话。
AccumBrowseFirst	获取最旧的累积器条目。
AccumBrowseGetField	获取浏览会话中光标位置指示的字段。
AccumBrowseNext	获取浏览会话中的下一个累积器条目。
AccumBrowseNumRecords	返回当前浏览会话中的记录数量。
AccumBrowseOpen	打开累积器浏览会话。
AccumBrowsePrev	获取浏览会话中的前一个累积器条目。
ProcessIsClient	确定当前执行进程是否包含客户机组件。
ProcessIsServer	确定当前执行进程是否包含特定服务器组件。

ServiceGetList	获取有关在此组件上运行并调用此函数的服务的的信息。
----------------	---------------------------

报警函数:

AlarmDspLast	显示最新的未确认报警。
AlmSummaryAck	确认激活数据浏览会话中当前光标位置的报警。
AlmSummaryClear	清除激活数据浏览会话中当前光标位置的报警。
AlmSummaryClose	关闭报警摘要浏览会话。
AlmSummaryCommit	将报警摘要记录提交到报警摘要设备。
AlmSummaryDelete	删除浏览会话中的报警摘要条目。
AlmSummaryDeleteAll	删除浏览会话中的所有报警摘要条目。
AlmSummaryDisable	禁止激活数据浏览会话中当前光标位置的报警。
AlmSummaryEnable	启用激活数据浏览会话中当前光标位置的报警。
AlmSummaryFirst	获取最旧的报警摘要条目。
AlmSummaryGetField	获取浏览会话中光标位置指示的字段。
AlmSummaryLast	将数据浏览光标放置在可用浏览集群列表的最后一个集群中的最近摘要记录处。
AlmSummaryNext	获取浏览会话中的下一个报警摘要条目。
AlmSummaryOpen	打开报警摘要浏览会话。
AlmSummaryPrev	获取浏览会话中的前一个报警摘要条目。
AlmSummarySetFieldValue	设置浏览会话中光标位置指示字段的参数值。
AlmTagsAck	确认激活数据浏览会话中当前光标位置的报警标签。
AlmTagsClear	清除激活数据浏览会话中当前光标位置的报警标签。
AlmTagsDisable	禁止激活数据浏览会话中当前光标位置的报警标签。
AlmTagsEnable	启用激活数据浏览会话中当前光标位置的报警标签。

AlmTagsFirst	获取最旧的报警标签条目。
AlmTagsGetField	获取浏览会话中光标位置指示的字段。
AlmTagsNext	获取浏览会话中的下一个报警标签条目。
AlmTagsNumRecords	返回当前浏览会话中的记录数量。
AlmTagsOpen	打开报警标签浏览会话。
AlmTagsPrev	获取浏览会话中的前一个报警标签条目。

超级精灵函数

AssGetProperty	从数据源获取有关当前超级精灵的信息。
AssGetScale	从数据源获取有关当前超级精灵的缩放信息。
AssInfoEx	替换 AssInfo 函数，并支持在线更改。

集群函数

ClusterActivate	允许用户激活未被激活的群集。
ClusterDeactivate	允许用户取消激活已激活的群集。
ClusterFirst	允许用户获取在工程中最先配置的群集。
ClusterIsActive	允许用户确定某群集是否已被激活。
ClusterNext	允许用户获取工程中配置的下一个群集。
ClusterServerTypes	允许用户确定为给定群集定义了哪些服务器。
ClusterStatus	允许用户确定客户机与群集上服务器间的连接状态。
ClusterSwapActive	允许用户在激活无效集群的同时，取消激活有效的集群。

I/O 设备函数

SubscriptionAddCallback	在标签订阅中添加回调函数。
SubscriptionGetAttribute	读取标签订阅的属性参数值。
SubscriptionRemoveCallback	删除标签订阅中的回调函数。
TagGetProperty	获取数据源的变量标签属性。
TagGetScale	获取数据源中某标签的指定缩放参数值。
TagSubscribe	订阅周期监视和事件控制的标签。
TagUnsubscribe	取消订阅周期监视和事件控制的标签。

标签函数

TagInfoEx	替换 TagInfo 函数，并支持在线更改。
TagWriteEventQue	打开标签写入事件队列。

任务函数

TaskCluster	获取执行当前任务的群集关系的名称。
-------------	-------------------

趋势函数

TrnBrowseClose	关闭趋势浏览会话。
TrnBrowseFirst	获取最旧的趋势条目。
TrnBrowseGetField	获取浏览会话中光标位置指示的字段。
TrnBrowseNext	获取浏览会话中的下一个趋势条目。
TrnBrowseNumRecords	返回当前浏览会话中的记录数量。
TrnBrowseOpen	打开趋势浏览会话。
TrnBrowsePrev	获取浏览会话中的前一个趋势条目。
TrnGetCluster	获取与趋势图线相关的群集名称。

TrnGetPenComment	获取趋势笔的注释。
------------------	-----------

报表函数

RepGetCluster	获取运行报表的群集的名称。
---------------	---------------

以下函数为 7.10 版中的新函数：

安全函数

FormSecurePassword	向当前窗体添加密码提示和编辑字段。
MultiSignatureForm	显示最多允许 4 位用户在其中确认凭据以批准某个操作的窗体。
MultiSignatureTagWrite	显示一个窗体，最多允许 4 位用户在其中确认凭据，以便批准向特定标签中写入特定的值。
UserLogin	操作员应使用安全密码字符串登录到 Vijeo Citect 系统。
UserVerify	在用户登录系统中使用身份验证功能
VerifyPrivilegeForm	显示允许单一用户输入其凭据的窗体。
VerifyPrivilegeTagWrite	显示一个窗体，允许任一用户在其中输入其凭据，以便批准向特定标签写入特定值。

其它函数

KernelQueueLength	获取队列中的行数。
KernelTableInfo	提供一致的方法，访问“内核表”中的项目。
KernelTableItemCount	获取“内核表”中的行数。
ProcessRestart	重新启动运行 Cicode 的当前进程。
ServerRestart	在不影响同一机器上运行的其他服务器进程的情况下，从系统内任一 Cicode 节点中重新启动任何报警、报表、趋势或 I/O 服务器。

Windows 函数

WinStyle	打开和关闭现有窗口中的滚动和滚动条功能。
----------	----------------------

已不再使用的函数

7.0 以上的版本不支持以下函数。如果您的工程使用这些函数，编译时会发生错误。

集群函数

ClusterGetName	返回主集群服务器和备用集群服务器的名称。
ClusterSetName	连接到特定集群服务器。

显示函数

DspCol	在 AN 处显示颜色。
--------	-------------

任务函数

ReRead	<p>使 Vijeo Citect 重新读取与当前 Cicode 任务关联的 I/O 设备数据。</p> <p>现在，标签在函数开始处被订阅，更新的函数值会被发送到订阅函数。</p> <p>更新速度满足以下条件时进行标签订阅：</p> <ul style="list-style-type: none">• 图形画面（如果从画面调用）• 缺省订阅速度由参数联机帮助 [Code]TimeData。如果从 Cicode 中调用（缺省 250 毫秒）• 使用 TaskNewEx 为任务创建要求的更新速度• 使用 TagSubscribe 为订阅创建要求的更新速度 <p>应确认订阅的更新速度与函数的要求相匹配。</p> <p>在从回路代码中删除 ReRead 后，您可能需要展开睡眠周期函数。</p> <p>这是要在它读取所有标签值时暂停创建的 ReRead。</p>
--------	--

7.10 以上的版本不支持以下函数。

窗口函数

WndGetProfile	获得 WIN.INI 参数的值。如果调用它，将返回 0。
WndPutProfile	在 WIN.INI 中更新参数。如果调用它，将返回 0。

时间和日期函数

TimeSet	设置新的系统时间
---------	----------

修改的函数

以下函数已在版本 7.0 中进行了修改：

报警函数

AlarmAck	确认报警。
AlarmAckRec	按记录号确认报警。
AlarmActive	确定任何报警是否在用户区域中激活。
AlarmClear	从激活报警列表中清除已确认的非激活报警。
AlarmClearRec	按记录号清除报警。
AlarmDelete	删除报警汇总条目。
AlarmDisable	禁止报警。
AlarmDisableRec	按记录号禁止报警。
AlarmDsp	显示报警。
AlarmDspLast	显示最近未确认的报警。
AlarmEnable	启用报警。
AlarmEnableRec	按记录号启用报警。
AlarmFirstTagRec	搜索报警标签、名称和描述的第一次出现。

AlarmGetDelayRec	通过报警记录号获取报警的延迟设置。
AlarmGetFieldRec	从报警记录号获取报警字段数据。
AlarmGetThresholdRec	通过报警记录号获取模拟量报警的阈值。
AlarmNextTagRec	搜索报警标签、名称和描述的下一次出现。
AlarmNotifyVarChange	激活时间戳数字量报警或时间戳模拟量报警。
AlarmSumAppend	将新的空白记录附加到报警汇总。
AlarmSumCommit	将报警汇总记录提交到报警汇总设备。
AlarmSumDelete	删除报警汇总条目。
AlarmSumFind	及时查找报警记录和报警的报警汇总索引。
AlarmSumFirst	获取最早的报警汇总条目。
AlarmSumGet	从报警汇总条目获取字段信息。
AlarmSumLast	获取最近的报警汇总条目。
AlarmSumNext	获取下一个报警汇总条目。
AlarmSumPrev	获取上一个报警汇总条目。
AlarmSumSet	在报警汇总条目中设置字段信息。
AlarmSumSplit	复制报警汇总条目。
AlarmSumType	检索表示指定报警类型的参数值。

I/O 设备函数

DriverInfo	为特殊 I/O 设备提供有关驱动程序的信息。
IODeviceControl	提供各个 I/O 设备的控制。
IODeviceInfo	获取有关 I/O 设备的信息。

其它函数

AccControl	控制累积器，如马达运行时间。
------------	----------------

ServerInfo	获取客户机和服务器信息。
ServerInfoEx	从多处理器环境中的指定过程获取客户机和服务器信息。
Shutdown	结束 Vijeo Citect 的操作。

报表函数

RepGetControl	获取报表控制信息。
Report	运行报表。
RepSetControl	设置报表控制信息。

SPC 函数

SPCAlarms	返回指定 SPC 报警的状态。
SPCProcessXRSGet	获取过程平均值、范围和标准偏差替代。
SPCProcessXRSSet	设置过程平均值、范围和标准偏差替代。
SPCSpecLimitGet	获取指定标签的规则界限（USL 和 LSL）。
SPCSpecLimitSet	设置指定标签的规则界限（USL 和 LSL）。
SPCSubgroupSizeGet	获取指定 SPC 标签的子集大小。
SPCSubgroupSizeSet	设置指定 SPC 标签的子集大小。

超级精灵函数

Ass	将变量标签与超级精灵关联。
AssPage	最多将八个变量标签与超级精灵关联，并在当前窗口中显示超级精灵。
AssPopUp	最多将八个变量标签与超级精灵关联，并在弹出式窗口中显示超级精灵。
AssTag	将变量标签与当前超级精灵关联。只为当前超级精灵创建关

	联，并且只在重新显示超级精灵之后才产生影响。
AssVarTags	最多将八个变量标签与超级精灵关联。仅对显示的下一个超级精灵创建此关联（在当前窗口或新窗口中）。可以重复使用此函数将多于 8 个的变量标签与超级精灵关联。
AssWin	最多将八个变量标签与超级精灵关联，并在新窗口中显示超级精灵。

标签函数

TagGetProperty	此函数从数据源读取变量标签的属性。
TagGetScale	从数据源的指定刻度处获取标签的参数值。
TagRamp	此函数将按 iPercentInc 定义的量递增标签。
TagRead	从 I/O 设备读取变量。
TagScaleStr	在指定刻度处获取标签的参数值。
TagWrite	通过指定变量标签写入 I/O 设备变量。

任务函数

MsgOpen	使用 Vijeo Citect 服务器或客户机打开信息进程。
---------	--------------------------------

趋势函数

TrendDspCursorTag	显示当前笔的标签名称。
TrnAddHistory	将早期历史文件恢复到趋势系统。
TrnDelHistory	从趋势系统删除早期历史文件。
TrnEventSetTable	为指定趋势标签设置来自表格的趋势数据。
TrnEventSetTableMS	为指定趋势标签设置事件趋势数据和时间数据（包括毫秒）。
TrnFlush	将趋势转存到磁盘。

TrnGetDefScale	获取趋势标签的缺省工程零刻度和满刻度。
TrnGetPen	获取笔的趋势标签。
TrnGetTable	将趋势数据存储到数组中。
TrnInfo	获取趋势标签的配置参数值。
TrnNew	在运行时创建新的趋势。
TrnSelect	设置趋势的画面。

窗口函数

WinCopy	将激活窗口复制到 Windows 剪贴板。
WinFile	将激活窗口写入文件。
WinNewAt	在指定位置为显示的所选页面打开一个新的显示窗口。
WinPrint	打印激活窗口。

v7.10 中的 CtAPI 函数

下节详细介绍了对 Vijeo Citectv7.10 中的 CtAPI 函数所做的更改：

- [已不再使用的函数](#)

已不再使用的函数

之前可用的“点”相关函数现在已不再可用，如果使用这些函数，将检测并返回错误，指出它们已不再受支持。为获得与之前使用这些函数进行调用时相同的结果，将其替换为基于平等使用相应标签参数和条件的标签。

以下为不再可用的“点”函数及其替换函数：

函数	替换
ctPointGetProperty	ctTagGetProperty
ctPointNew	ctListNew 或 ctTagWrite 和 ctTagRead
ctPointRead	ctTagRead 或 ctListRead 和 ctListData
ctPointWrite	ctTagWrite 或 ctListWrite

如果在单独的标签上使用点函数，应使用 `ctTagRead`、`ctTagWrite` 函数代替。如果要将多个标签聚积到一个点，应该使用 `ctListNew` 并通过 `ctListAdd` 将标签添加到列表中。然后使用 `ctListWrite`、`ctListRead` 和 `ctListData` 针对标签执行读写操作。

以下函数与基于标签的操作不相关。它们已不再使用并且没有替换函数。

- `ctPointBitShift`
- `ctPointClose`
- `ctPointCopy`
- `ctPointDataSize`
- `ctPointToStr`
- `ctStrToPoint`
- `ctTagToPoint`

版本v7.10 中的内核命令

版本 7.0 的 Vijeo Citect 中已不包括下列内核命令：

- Kernel Alarm
- Kernel Trend
- Kernel Report
- Kernel IOserver
- Kernel Client
- Probe
- NetBIOS
- PageNetstat

Chapter: 3 升级到 Vijeo Citect v7.10

如果是从 v7.0 升级到 v7.10，您不需要执行下列过程。

要升级您的工程，以从 v6.x Vijeo Citectv7.10 中运行，您需要：

- **升级 CTAPI 应用程序**
确认在升级和运行任意 Vijeo Citect V7.x 工程前，升级 CTAPI 应用程序。
- **配置 I/O 设备**
升级前，确认按照工程运行的要求配置 I/O 设备。
- **定义群集**
可立即定义群集。工程必须配置为至少使用一个群集。
- **配置网络地址**
在工程中定义服务器时使用的网络地址和计算机端口。
- **配置服务器**
在工程中定义的报警、报表、趋势和 I/O 服务器。
- **配置标签为使用群集**
可配置报警、报表、趋势、SPC 标签和累积器在特定群集中运行。

注意： 如果运行的是 CitectSCADA 5.5 版本，确认将您的工程升级到 6.x 版本，然后再升级到 v7.10。

另请参见

[升级过程](#)

[极少使用和删除的函数](#)

极少使用和删除的功能

一些 Cicode 函数已经极少使用，或者被删除和修改了。有关详细信息，请参考 [Vijeo Citect 的新增内容](#)。

升级过程

要将 v6.x 中现有工程升级到 7.0 或 v7.10 版本，应执行以下每个步骤：

- [升级 CTAPI 应用程序](#)
- [配置 I/O 设备](#)
- [运行 Citect 安装程序](#)
- [启动 Vijeo Citect](#)
- [定义集群](#)
- [运行迁移工具](#)
- [配置网络地址](#)

- [配置服务器](#)
- [配置标签以使用群集](#)
- [编译工程](#)
- [运行计算机设置向导](#)

升级 CTAPI 应用程序

要设置 Vijeo Citectv7.10 与系统中的 CTAPI 应用程序（包括 Vijeo Citect 报告和 Ampla）通讯，需确认在运行任意升级的 Vijeo Citectv7.10 工程之前，将这些产品升级为其最新的版本。

同样，如果您使用的是自定义的 CTAPI 应用程序，可升级计算机上的 Vijeo Citect，在升级其他任意 Vijeo Citect 计算机前，必须先安装这些应用程序。

另请参见

[配置 I/O 设备](#)

[Vijeo Citect 技术参考中的“CtAPI 函数”](#)

配置 I/O 设备

升级过程可以确认您工程的功能升级到 7.0 版本。为简化升级，在升级过程中您工程配置的某些部分可以更改。因此，确认在升级之前将工程配置成您希望其运行的方式十分重要。

特别是，因为已经不再支持内存 I/O 设备，所以工程（指定 MEMORY 作为端口）中定义的内存 I/O 设备在升级后的工程中将配置为本地变量。本地变量无需进一步配置即可提供与内存 I/O 设备相同的功能。

然而，这也意味着如果有 I/O 设备暂时定义成内存 I/O 设备用于模拟或测试，也将会被升级过程错误配置为本地变量。请确认在升级到 7.0 版本之前按照要求设备运行的方式来进行配置。



升级提醒输出配置

在安装 Vijeo Citectv7.10 之前，请确认并调整所有 I/O 设备的配置。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

在因为升级过程引起变更而尝试配置 I/O 设备之前，首先阅读 [配置本地变量](#)和 [使用内存模式](#)中的信息。其中包括关于本地变量及代替内存 I/O 设备的所有其他 I/O 设备选项的详细信息，以便您适当地选择和配置工程。

由于协议属性设置为“报警”的报警设备不再受支持且将被迁移工具移除，因此还应该配置您的警报服务器，以便在警报服务器上使用发布报警属性的属性。



警告

升级会删除部分报警

在安装 Vijeo Citectv7.10 之前，应对工程中的所有报警服务器进行配置以便使用报警服务器中的“发布报警属性”属性。忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

当运行迁移工具时重新配置将生效。有关此工具的详细信息，请参考 [迁移工具](#)。

另请参见

[运行 Citect 安装程序](#)

[配置本地变量](#)

[使用内存模式](#)

运行 Citect 安装程序

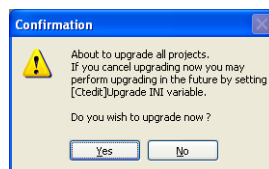
要开始升级，请运行 Vijeo Citectv7.10 安装程序。安装程序将引导您通过大量步骤，直到安装完成。

注意： 在安装 v7.10 之前先卸载全部现有版本，因为 Vijeo Citect 不支持并行运行的不同版本。此外，要使用 v7.10 示例和 CSV_Example 工程，建议您在启动安装之前先使用 Citect 工程管理器删除现有示例和 CSV_Example 工程。

启动Vijeo Citect

当您初次启动 Vijeo Citect 时，将自动更新您的工程。

1. 要启动 Vijeo Citect，请点击**开始 | 所有程序 | Schneider Electric | Vijeo Citect 7.10 | Vijeo Citect 工程管理器**。随即将显示以下信息：



2. 点击**是**确认升级。

另请参见

[定义集群](#)

[迁移工具](#)

迁移工具

首次启动 Vijeo Citect v7.10 时发生的自动更新不会全面升级您的工程，如果是从 v6.x 进行迁移，必须先使用迁移工具完成迁移工作。此更新是被动的动作，用于更新两个版本之间已更改的任何数据库字段定义并复制 v7.10 中所需的新文件。进行自动更新之前，系统会提供能够取消更新的选项。用户可以在日后调用升级（通过在 Citect.ini 文件中将 [CtEdit]Upgrade 参数设置为 1 (True)）。

自动更新完成后，应在运行迁移工具之前准备好工程。

迁移工具是单独的应用程序，应在执行自动更新后手动运行，并在用户为最后的迁移准备好工程后由用户初始化。此工具会提供对结合在版本 7.0 和 v7.10 中的工程功能的重要更改。

要使用此工具成功进行更新，准备好现有工程非常重要。

Vijeo Citect 的 v7.10 中引入的某些特点需要更改版本 6.x 中的项目数据。

另请参见

[内存设备](#)

[报警设备](#)

[包含工程](#)

[使用迁移工具](#)

内存设备

在以前版本的 Vijeo Citect 中，可通过将端口值设置为“内存”而将 I/O 设备定义为内存设备。这通常是为了实现下列一种目的：

- 提供当前不连接到系统而将来要使用的设备，但需要在工程的此阶段对它们进行配置。
- 用作不存在相应物理 I/O 设备但您需要进行 I/O 变量功能（如报警）相关数据存储时的虚拟设备。
- 用作替代 Cicode 全局变量而作为进程的本地变量使用的变量。

在版本 7.0 中，仍可以将 I/O 设备作为未来备用或虚拟设备，但应当手动将“端口”参数设置为“内存”之外的未使用值，并在运行迁移工具之前将设备的“内存”属性设置为 True，以指示它是一个离线的内存形式设备。

您需要重新审查工程，确定哪些内存 I/O 设备是本地变量占位符，哪些需要更改为非内存，以便迁移工具不会转换它们的变量。

迁移工具会将标识为“内存”设备的任何 I/O 设备的端口设置为新的本地变量，而将原始设备记录删除。

另请参见

[配置 I/O 设备](#)

[报警设备](#)

[转换内存变量](#)

报警设备

在先前版本的 Vijeo Citect 中，报警设备定义为其“协议”属性设置为“报警”的设备。在版本 7.0 中，用于配置这类设备的功能现在取代为在报警服务器上将“发布报警属性”属性设置为 True。

其“协议”属性设置为“报警”的所有报警设备都将由迁移工具从 I/O 设备表中删除。运行迁移工具之前，应标识这些设备并在报警服务器上将“发布报警属性”属性设置为 True。

另请参见

[报警服务器定义](#)

迁移工具可以删除所有的内存和报警设备记录。如果要在以后转换变量并删除设备，请取消选择“删除过时内存和报警设备”选项

另请参见

[转换内存变量](#)

转换内存变量

内存变量即其 I/O 设备端口属性被设置为 "Memory" 或者 "MEM_PLC" 的变量。

如果存在多个同名的 I/O 设备（可能位于不同的 I/O 服务器上），无论端口的值是什么，都认为此设备非内存设备。换句话说，迁移工具不能处理重名的内存设备变量。

另请参见

[插入新建本地变量](#)

[删除标签变量](#)

插入新建本地变量

当迁移工具运行时，将为每个标识的内存变量插入一个本地变量记录，而且变量数据将被复制到该新建本地变量。

本地变量的字段比变量更少；下表显示了复制数据时从变量到本地变量的映射。

标签变量参数或常数值	本地变量参数
变量标签名称	名称
数据类型	日期类型
(空)	数组大小
工程零刻度	零刻度
工程满刻度	满刻度
注释	注释

除数组大小（它在版本 7.0 中已专门为本地变量引入）外，所有字段都会从相同或类似字段收到它们的值。

另请参见

[删除标签变量](#)

删除标签变量

在创建完本地变量记录之后，迁移工具会将在之前步骤中发生转换的所有标签变量记录插入，然后删除原始标签变量。

如果插入本地变量时检测到错误，则将无法删除标签变量。如果出现这种情况，可能会出现两个具有相同名称和数据的记录，一个位于本地变量（新插入的记录）中，一个位于标签变量（已删除的原始记录）中。您需要手动删除任一变量，或在删除错误起因后恢复备份的工程，然后再次运行迁移工具。

另请参见

[缺省刻度](#)

[删除旧 I/O 设备](#)

缺省刻度

两个标签变量和本地变量中的刻度属性是可选的。如果未指定刻度值，缺省值由 Citect.ini 文件中的参数指出。参数名称为“DefaultSliderScale”，位于 Citect.ini 文件的 [通用] 部分下。刻度的缺省值为 0-32000，但缺省滑钮的刻度为 True 的情况除外，在此情况下，缺省值取决于类型（例如 Integer、String 等）。

迁移工具将读取此参数，如果此参数未设定或设为 False，那么它会将任何空刻度属性显式设置为 0 至 32000 范围内的值。即使零刻度或满刻度参数有值，这也将完成，在此情况下，空刻度参数将收到缺省值。

如果 Citect.ini 文件中的 DefaultSliderScale 被设为 True，刻度参数不会用缺省值填充（如果为空），而是在运行时翻译。

删除旧 I/O 设备

删除旧 I/O 设备是迁移工具中的一个可选步骤，将在转换完所有内存变量后执行。如果已选择删除选项，那么所有旧的内存设备和报警设备都将作为迁移工具的最后一步操作被删除。

另请参见

[包含工程](#)

包含工程

每个工程都可以包含多个包含工程。而且，任何包含工程还可以包含自己的包含工程，从而形成级联工程。

迁移工具需要一步处理原始工程及所有包含工程。因此，可以在一个工程中定义变量，而该工程参考在另一个包含工程中定义的 I/O 设备。

迁移工具对“主”工程按顺序执行此步骤，然后对每个包含工程执行此步骤。

在两个主工程以同一个工程为包含工程的情况下，当您处理具有共享包含工程的工程时，不选中“删除过时内存和报警设备”复选框很重要。这是因为删除会在迁移过程结束时在每个主工程并随之在每个包含工程上进行。这会导致在第一个主工程中删除 I/O 设备，而它又被随后步骤处理的共享包含工程中的某个标签所参考。

如果两个独立的“主”工程包含相同的包含工程，请在不选择删除过时设备的情况下在每个“主”工程上运行迁移工具。



警告

迁移工具会删除已配置的输出和报警

在具有共享包含工程的工程上运行迁移工具之前，应取消选中“删除过时内存和报警设备”选项。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

要删除过时设备，建议在迁移工具成功完成后（没有选中复选框情况下），再选中该复选框运行一次迁移工具。这样就可以安全地删除设备，因为在迁移工具第一次成功完成时就完成了所有标签转换。

使用迁移工具

注意：在使用迁移工具之前，强烈建议您熟悉它执行的进程，并熟悉对于 [迁移工具](#) 下描述的现有工程您需要执行的准备步骤。

运行迁移工具：

1. 备份要迁移的工程。
2. 从“Citect 工程管理器”或“Citect 编辑器”菜单栏中，选择“工具 | 迁移工具”以显示“迁移工具”对话框。
3. 接受编辑框中显示的工程，或浏览要升级的工程。如果工程包含其他工程，只选择“主工程”。任何包含的工程将被自动处理。
4. 如果希望在成功的迁移之后删除设备，请选择“删除过时的内存和报警设备”复选框。

注意：如果一个工程包含其他工程，而这些工程由多个主工程共享，则当您对这些工程首次运行此工具时，请不要选择此复选框。如果要删除这些设备并且首次运行之后迁移成功，请使用此选项第二次运行此工具。

5. 单击“迁移”开始迁移进程，或单击“关闭”退出而不执行迁移。

6. 迁移进程将开始并显示一个进程对话框，指示转换的阶段以及正在迁移的工程的名
称。如果希望在此时取消迁移，请单击“中止”按钮。

注意：中止迁移将停止迁移进程，任何已经完成的更改不会被撤销。您必须从在第一步中创建的备份中还原工程。

7. 当迁移进程完成后，将显示一个确认对话框，指示转换的变量数量以及删除的 I/O 设
备数量（如果在迁移开始时选择了设备删除）。
8. 单击“关闭”按钮关闭对话框。

定义集群

即使不希望在工程中使用，也必须定义至少一个集群。缺省情况下，所有标签和服务器都将在定义的集群中运行。

1. 在工程编辑器中，选择**服务器 | 集群**。将显示“集群”对话框：
2. 在**集群名称**域中，输入集群的名称（最多 16 个字符）。此名称在工程中必须是唯一
的，且不能包含空格。
3. 在**注释**域中，输入有用的注释。此属性是可选的，不会在运行时使用。
4. 点击**添加**。

另请参见

[配置网络地址](#)

配置网络地址

在工程编辑器中，配置要在工程中用作服务器的每台机器的网络地址。

1. 在工程编辑器中，选择**服务器 | 网络地址**。将显示网络地址对话框：
2. 在**名称**域中，输入要配置的网络地址的名称（最多 16 个字符）。名称在工程中必须
唯一且不能包含空格。
3. 在**地址**域中，输入要配置的机器的 IP 地址或计算机名。
4. 在**注释**域中，输入一些有用的注释。此属性可选且不用于运行时环境。
5. 点击**添加**。

另请参见

[配置服务器](#)

配置服务器

所有主报警和备用报警、报告、趋势及 I/O 服务器现在都是使用工程编辑器定义。其中包括为每台服务器指定网络地址和簇。

默认端口

每台服务器都有为其指定的一个唯一默认端口。该默认端口只能与该类服务器配合使用。尝试在其他类型的服务器上使用默认端口将导致下列编译错误：

“保留了无效端口号 (2073-2082,20222,21)。”

下表列出了默认端口号及其关联的服务器类型。

默认端口	服务器类型	服务器角色
21	FTP 服务器	IDC 画面下载
2073	CTAPI	CTAPI 通讯
2074	客户机	Cicode 调试
2075	报表服务器	报表服务器通讯
2076	报警服务器	报警服务器通讯
2077	趋势服务器	趋势服务器通讯
2078	输入输出服务器	旧 I/O 通讯
2079	IDC	Internet 显示服务器/客服机通讯
2080	报警服务器	报警属性连接器
2082	10 服务器	发布预订 I/O 服务器通讯
20222	ODBC	ODBC 服务器

有关配置每类服务器的详细信息，请参考：

- [报警服务器定义](#)
- [报表服务器定义](#)
- [趋势服务器定义](#)
- [I/O 服务器定义](#)

配置标签以使用簇

如果您定义了多个簇，则可以配置要运行特殊簇的报警、报表、趋势、SPC 标签和通讯。使用工程编辑器，现在可以为每个标签指定相应的簇。

- [配置报警标签以使用簇](#)
- [配置报表标签以使用簇](#)
- [配置趋势标签以使用簇](#)
- [配置 SPC 标签以使用簇](#)
- [配置累积器以使用簇](#)

配置报警标签以使用簇

在工程编辑器中，选择**报警**，然后选择要配置的报警类型。将显示已选择报警类型的对话框：

在**簇名称**域中，为每个报警选择要运行报警的簇的名称。如果工程仅定义了一个簇，可以保留此域为空。报警将默认为定义的簇。

如果工程有多个簇，而您未在此对话框中选择簇名称，Vijeo Citect 会认为在所有定义的簇上运行该报警。

如果希望该报警属性被发布并被视为正常的变量标签，而且在它作为 I/O 连接器时能够被报警服务器侦听，请在“报警服务器定义”对话框中使用 **F2** 键打开 [扩展属性页面](#) 并将“发布报警属性”设置为 “True”。

点击**替换**保存更改。

另请参见

[配置报告标签以使用簇](#)

配置报表标签以使用簇

在工程编辑器中，选择**系统 | 报表**。将显示报表对话框：

在**簇名称**域中，为每个报表选择将运行该报表的簇。如果工程仅定义了一个簇，可以保留此域为空。报表将默认为定义的簇。

如果工程有多个簇，而您未在此对话框中选择簇，Vijeo Citect 会认为在所有定义的簇上运行该报表。

点击**替换**保存更改。

另请参见

[配置趋势标签以使用簇](#)

配置趋势标签以使用簇

在工程编辑器中，选择**标签 | 趋势标签**。将显示趋势标签对话框：

在**簇名称**域中，为每个趋势选择将运行该趋势的簇。如果工程仅定义了一个簇，可以保留此域为空。趋势将默认为定义的簇。

如果工程有多个簇，而您未在此对话框中选择簇，Vijeo Citect 会认为在所有定义的簇上运行该趋势。

点击**替换**保存更改。

另请参见

[配置 SPC 标签以使用簇](#)

配置 SPC 标签以使用簇

在工程编辑器中，选择**标签 | SPC 标签**。将显示 SPC 标签对话框：

在**簇名称**域中，为每个 SPC 标签选择将运行该 SPC 标签的簇。如果工程仅定义了一个簇，可以保留此域为空。SPC 标签将默认为定义的簇。

如果工程有多个簇，而您未在此对话框中选择簇，Vijeo Citect 会认为在所有定义的簇上运行该标签。

点击**替换**保存更改。

另请参见

[配置累积器以使用簇](#)

配置累积器以使用簇

在工程编辑器中，选择**系统 | 累积器**。将显示累积器对话框：

为每个累积器，在**簇名称**域中选择将运行该累积器的簇。如果工程仅定义了一个簇，可以保留此域为空。累积器将默认为定义的簇。

如果工程有多个簇，而您未在此对话框中选择簇，Vijeo Citect 会认为在所有定义的簇上运行该累积器。

点击**替换**保存更改。

另请参见

[编译工程](#)

编译工程

配置完工程后，对其进行编译并确认没有错误。

在此阶段，您可以重新配置某些 Cicode 以支持在线更改。尤其是建议在以后的软件版本中使用的 AssInfo 和 TagInfo 函数。大多数情况下，可以将其替换为 AssInfoEx 和 TagInfoEx 函数。

运行计算机设置向导

针对每个运行工程的计算机运行计算机设置向导。在向导的各个阶段，为该计算机配置适当的设置。

另请参见

[运行计算机设置向导](#)

故障排除

升级到 Vijeo Citectv7.10 时，应注意并仔细考虑以下问题：

- [编译器错误](#)
- [升级使用分布式服务器的工程](#)

编译器错误

在将工程配置为可在版本 7.0 中运行之前，编译工程将产生多个编译器错误。这可能包括有关建议使用或删除函数的信息，以及检测到的“未定义集群”错误。在工程中定义集群之后，此检测到的错误将得以解决。

另请参见
[故障排除](#)

升级使用分布式服务器的工程

如果已在使用分布式服务器、全局包含工程和群集工程的 6.x 版本中实现了群集，则应在 7.0 版本中将工程配置为使用群集。

全局显示工程和群集工程的现有结构可保留不变（即，全局显示工程包括每个群集工程）。

以下各点说明了 7.0 版本中群集的建议工程结构：

- 在全局显示工程中包括独立的通讯工程。在此工程中，建议您仅定义网络地址、群集和服务器。

注意： 定义独立的通讯工程意味着编译全局显示工程时，通讯信息不需要加载来自群集工程的所有数据。

- 在每个群集工程中，指定警报、趋势、报告、SPC 标签和累积器的适当群集。
- 您可能需要修改全局显示工程中的按钮和画面，特别是，当这些按钮和画面使用了已修改和极少使用的群集函数时。

另请参见
[极少使用和删除的函数](#)

Chapter: 4 关于 Vijeo Citect

Vijeo Citect 是一种管理控制和数据采集 (SCADA) 解决方案，用于管理和监视制造、主要生产、工具传递和设备管理中的流程。

与 Vijeo Citect 安装关联的图形、控件、配置数据和编程通过项目配置和实现。项目作为与工厂基础设施一同部署的生产设备的数字表示形式，允许以实时方式监视和控制整个系统。

配置 Vijeo Citect 工程

一开始，通过为与每个设备和数据源关联的变量加标签，使用 Vijeo Citect 的配置环境来识别和确定设备和数据源机器地址。

然后，可以使用模板设计引用这些标签的图形画面，创建您的对象可以用来查看和控制系统的接口。

使用这些图形画面，您可以：

- 使用动画显示工厂的操作状态和运行状况
- 使用键盘命令和图形工具为操作员提供产品设备的集中或本地控制
- 开发出多层安全系统，可以根据功能组或地理区域控制用户访问
- 以图形格式实施标签数据的历史和毫秒趋势。

此外还包括了一种功能强大的脚本语言，可以实现自定义、可编程功能。

部署 Vijeo Citect

工程将在客户端-服务器网络架构间进行部署。服务器用于管理与现场设备的通信和比较产品数据，而客户端则提供操作员和管理员访问系统并与系统进行交互的接口。

此架构可灵活地调整 Vijeo Citect 以适应任何生产方案，它还支持缩放性、服务器群集和系统冗余。

运行工程

在运行时完整编译和实施工程后，生产员工可以直观地监视系统、初始化生产过程和响应报警条件。

可以整理历史数据和趋势数据并将其分布到评估工作性能评量，如生产容量、效率和维护要求。

Chapter: 5 工具

Vijeo Citect 的架构可分为三个不同的功能区：

- 配置
- 运行期间
- 驱动程序

“配置”包括准备和构建工程所需的全部任务，而“运行期间”则是工程在实时生产环境中的实施。

驱动程序支持通过多种通讯协议实现与设备的通讯。驱动程序定义 Vijeo Citect 与特定设备进行通讯所需的特定工程设置。

考虑 Vijeo Citect 附带的这些工具时，最容易了解它们在配置中或运行期间的作用。

另请参见

[配置工具](#)

[运行期间工具](#)

[驱动程序](#)

配置工具

使用下列工具，您可以配置工程及其组件，和设置计算机，以便使用 Vijeo Citect:

	Vijeo Citect 浏览器	用于创建和管理您的工程的应用程序。它会显示所有工程的列表，并提供对每个工程的组件的直接访问。您可以使用浏览器重命名、备份、恢复或删除工程。 见 管理工程
	Vijeo Citect 工程编辑器	用于创建和管理工程的配置信息的应用程序，包括标签、报警、系统组件及通讯组件 见 工程组件
	Vijeo Citect 图形编辑器	用于设计、创建和编辑工程图形组件的应用程序，包括模板、图形对象、符号、精灵和超级精灵 见 定义和画图形画面
	计算机设置编辑器	用于编辑配置文件和生成要比较和分析文件的报表的工具 请参见 计算机设置编辑器在线帮助

	计算机设置向导	<p>可让您自定义计算机的设置和定义其角色及功能的向导</p> <p>见 运行计算机设置向导</p>
---	---------	--

运行时工具

用户可以通过以下工具运行、监视和控制运行期间的工程：

	Vijeo Citect Web 客户机	<p>在 Web 浏览器中显示实时 Vijeo Citect 工程。</p> <p>请参见 Vijeo CitectWeb 客户机</p>
	Internet 显示客户机	<p>用于通过 Internet 从远程位置运行 Vijeo Citect 工程的计算机。</p> <p>请参见 通过 Internet 运行系统</p>
	过程分析器	<p>允许用户对比和分析运行期间的历史和实时趋势数据和报警数据的 Active X 控件。</p> <p>有关详细信息，请参见 过程分析器用户向导中的“配置过程分析器”</p>
	Vijeo Citect 运行时管理器	<p>用于管理和控制工程的 CPU 配置和每个组件的运行状态的应用程序。</p> <p>有关详细信息，请参见 运行管理器在线帮助中的“启动运行管理器”</p>

驱动程序

Vijeo Citect 可以与 I/O 设备的阵列进行通讯，这些设备包括 PLC（可编程逻辑控制器）、循环控制器和离散控制系统（DCS）。

这些 I/O 设备可能是本地的（直接连接到 I/O 服务器），也可能是远程的（通过某种中间通讯方式，例如电话，连接到 Vijeo Citect）。

驱动程序可以通过各种通讯协议（包括以太网、TCP/IP 以及串行）与这些设备进行通讯。驱动程序定义 Vijeo Citect 与特定设备进行通讯所需的特定工程设置。该设置包括以下信息：

- 板卡
- 端口
- 设备

- 标签地址

有关驱动程序以及如何在系统中使用这些驱动程序的详细信息，请参见 [与 I/O 设备进行通讯](#)

Chapter: 6 工程的组件

跨以下类别对可以纳入工程的组件进行了逻辑划分：

- [图形组件](#)
- [标签](#)
- [报警](#)
- [系统组件](#)
- [通讯组件](#)
- [I/O 服务器组件](#)
- [Cicode / CitectVBA](#)

这些分类通过与每一工程关联的文件夹集显示在 Citect 浏览器中。

构建工程时，加入的组件将列在相关的工程文件夹中。从这些文件夹中选择一个项目后，所需工具中的选定组件将开始编辑其属性。

另请参见

[图形组件](#)

[标签](#)

[报警](#)

[系统组件](#)

[通讯组件](#)


[I/O 服务器组件](#)

[Cicode / CitectVBA](#)

图形组件

工程的图形组件提供了用于创建在客户端上显示的屏幕的内容。包括：

	画面	屏幕布局的基础 参见 定义和绘制图形画面
	模板	用于使显示屏幕标准化的画面布局集合 参见 使用画面模板
	符号	存储在库中以备重新使用的图形对象 参见 使用符号
	精灵	为便于复制而将图形和功能元素分组的对象 参见 了解精灵

	超级精灵	可以在运行时向其传递设备专有信息的精灵 参见 使用超级精灵
---	------	--

当在图形编辑器中创建工程画面时，加入的组件会被添加到当前工程的图形文件夹中的相关子目录内。

标签

标签用于标识使用 Vijeo Citect 监控的基础结构的终点。您向标签提供的名称将成为寄存器地址的标记符，使其可以直观地应用于多个图形页面和报警通知。

Citect 浏览器中的工程的标签文件夹中包含三种类型的标签：




	变量标签	用于标记寄存器地址 参见 标签过程变量
	趋势标签	用于标记数据趋势的标签 参见 趋势数据
	SPC 标签	用于根据统计过程控制原则标记标签 参见 SPC 标签

从 Citect 浏览器中选择上述标签类型的一种将调用工程编辑器中与之关联的配置对话框。

报警

报警用于确定系统中需要注意的情况。Vijeo Citect 支持七种不同的报警类型：

	数字量 请参见 数字量报警
	模拟量 请参见 模拟量报警
	带时间戳的 请参见 带时间戳的报警
	高级 请参见 高级报警

	多-数字量 请参见 多-数字量报警
	带时间戳的数字量 请参见 带时间戳的数字量报警
	带时间戳的模拟量 请参见 带时间戳的模拟量报警

您还可以在工程中使用报警分类来帮助识别和管理报警。

系统组件

可以通过工程的系统组件自定义、管理和跟踪自己的 Runtime 系统。包括：

	键盘键	分配给键盘键的有意义的名称 参见 键盘键
	键盘命令	带有相关说明的键序列 参见 系统键盘命令
	报表	Runtime 数据和特殊条件的自定义演示 参见 报表信息
	事件	响应特定 Runtime 触发器（如 Cicode 表达式或变量标签）而执行的命令。当事件触发器为 True 时，此命令执行 参见 配置事件
	累积器	跟踪连续 Runtime 数据的变量标签。可以采取将变量标签制成动画和趋势的方式监视并显示数据 参见 使用累积器
	设备	可以将高层数据传输到其它组件（如 RTF 文件、ASCII 文件）的组件和打印机 参见 配置设备
	用户	用户个人资料，用于限制和授予对 Runtime 系统的访问权。 参见 维护用户记录
	组	系统区域组，用于简化用户个人资料的管理 参见 定义区域
	标签	对常用命令和表达式的系统范围的替换 参见 使用标记符

	字体	用于显示报警和对象的字体 参见 使用系统字体
	参数	用于精确调整 Runtime 系统的内置操作设置 参见 Vijeo Citect 技术参考中的“Citect.ini 文件参数”
	包含工程	数据库记录自动包含在用户工程中的预定义工程 参见 包含工程


通讯组件

项目的通讯组件是系统中通讯硬件的配置表示。包括：

	板	通过 I/O 设备启用各种类型通讯的硬件 参见 板属性
	端口	板和 I/O 设备间的物理连接 参见 接口属性
	调制解调器	用于将 Vijeo Citect 连接到拨号远程 I/O 设备的硬件。 参见 在 Vijeo Citect 中设置调制解调器：
	I/O 设备	与现场控制或监视设备通讯的设备项 参见 I/O 设备属性
	I/O 设备地址	I/O 设备 Vijeo Citect 的唯一通讯地址

I/O 服务器组件

工程的服务器组件是系统中服务器计算机的配置表示。包括：

	集群	跨多个物理计算机运行的服务器的逻辑组 请参见 实施集群
---	----	--

	网络地址	主要服务器或备用服务器的 IP 地址或计算机名 请参见 网络地址定义
	警报服务器	监视报警并在适当的客户机中显示这些报警的服务器
	报表服务器	控制报表处理的服务器
	趋势服务器	控制趋势信息累积和记录的服务器
	I/O 服务器	在 I/O 设备和客户机之间交换数据的专用通讯服务器

Cicode / CitectVBA

Vijeo Citect 提供了两种编程语言，可以用它们来控制 and 操作 Vijeo Citect 的组件：

	Cicode	一种结构化的编程语言，设计用来在 Vijeo Citect 中监控设备。 参见 Cicode 简介 。
	CitectVBA	一种兼容 Visual Basic for Applications (VBA) 和 VBScript 的基本脚本语言。 参见 CitectVBA 简介 。

Chapter: 7 典型系统方案

本章中说明的方案说明了如何使用 Vijeo Citect 来支持主要产品、实用程序发行以及生产中的常用进程。

实际上，工程可以与本方案相结合，实现高度自定义功能和可伸缩性。但是，已对这些采样进行简化，以说明如何对 Vijeo Citect 进行配置及部署，以符合某产品系统的具体要求。

单机系统

所有系统组件都运行于一台计算机上。请参见 [单机系统](#)。

分布式 I/O 系统

Vijeo Citect 可用于监视和管理连接到远程 I/O 服务器的分布式设备。请参见 [分布式 I/O 系统](#)。

冗余服务器系统

与系统相关联的一个或多个服务器重复，并定义为主单元和备用单元，允许系统在某个服务器不正常时继续运行。请参见 [冗余服务器系统](#)。

客户端-服务器系统

与系统相关联的服务器和客户机独立分布于网络中的各个计算机中，提供强大的可访问性和性能优势。请参见 [客户端-服务器系统](#)。

冗余和分布式控制系统

远程或地理独立的产品系统拥有完全的可操作子系统，可通过本地进行监视和控制。如果这种子系统出现部分或全部不正常（表现为禁止本地控制），这种配置允许远程控制客户端控制受影响的子系统。请参见 [冗余和分布式控制系统](#)。

集群式控制系统

某产品系统组织成多个不连续的区域，由每个区域的操作员进行监视。但是，也有一个控制级别可监控系统的所有区域。请参见 [集群式控制系统](#)。

负载分担系统

系统对负载进行分割，否则会对多台计算机上的系统造成影响，这样可以更好地利用可用的基础架构。请参见 [负载分担系统](#)。

另请参见

[集群连接配置](#)

单机系统

Vijeo Citect 的单机安装程序运行单个计算机中系统的所有服务器和客户机组件。其中包括：

- I/O 服务器
- 报警服务器
- 趋势服务器
- 报表服务器
- 控制客户机

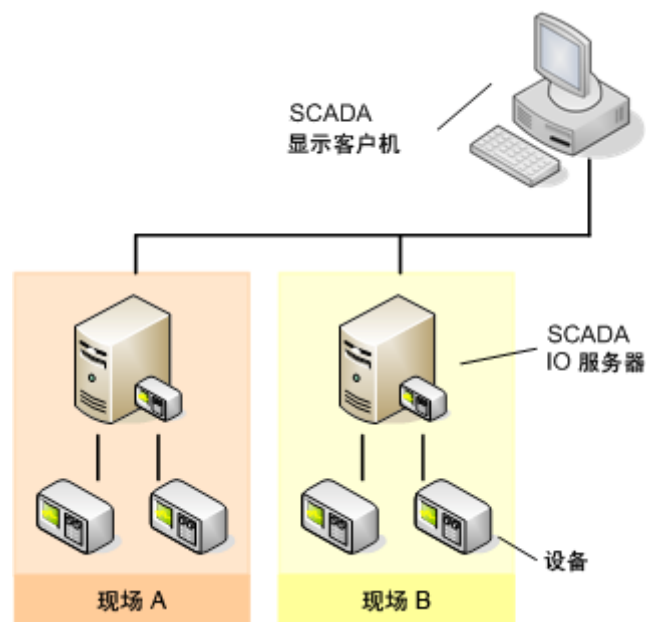
这使 Vijeo Citect 可以作为一个小型自包含系统运行。

注意：可以将单机系统的服务器和客户端组件作为单进程和多进程系统运行。建议只将单进程设置用作控制系统的短期解决方案，或运行演示和测试工程。向系统添加冗余可使系统更加可靠和更有效率。

分布式 I/O 系统

此方案演示了将 Vijeo Citect 连接到一些设备的方法，这些设备分布在广泛地理区域内的若干位置。

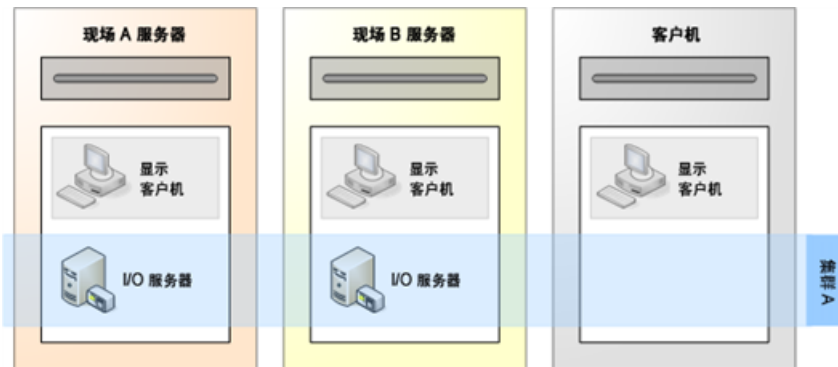
不要尝试直接通过远程连接来连接设备，而是在每个地点放置一台 I/O 服务器，启用在系统内管理的通讯。



此模型对包含串口或通讯能力有限的设备也很有用。通过在工厂中设置 I/O 服务器以便与这些设备接口，可以优化在慢速或低带宽网络上的通讯并全面改进性能。

虽然 I/O 服务器在地理上分布在许多地点，但由于一个集群可以支持多个 I/O 服务器，因此，此类系统可以被配置为单个集群系统。

下面的图表展示了如何使用单个集群跨服务器计算机访问此类系统的部署。

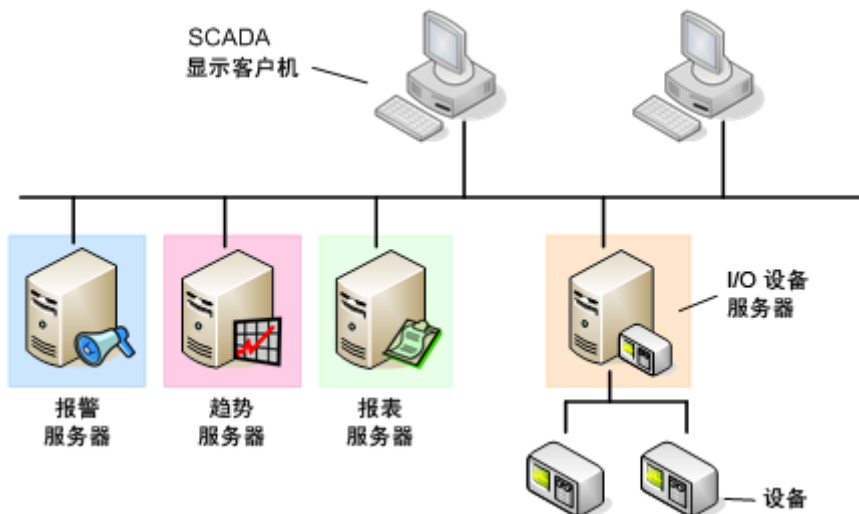


只有当工程要求调用多个冗余的报警、趋势和报告服务器组合时，才需要其他集群。

客户端-服务器系统

Vijeo Citect 的客户端-服务器架构使得系统的组件可以分布在局域网的众多计算机上，从而形成一个具有地域灵活性和性能优势的系统。

每个组件在工程内只由一个地址唯一标识，因此每个组件的位置和硬件要求都可以单独考虑。



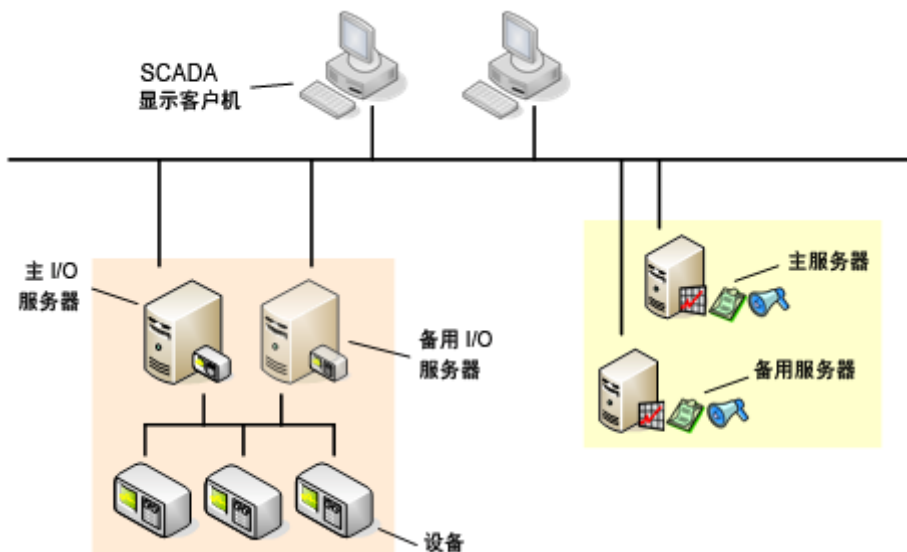
下面的示意图示范了如何在单一集群中配置此示例。



注意，每个服务器都作为整个系统架构中的控制客户端。

冗余服务器系统

在工程中定义主服务器和备用服务器的能力允许在系统基础架构中构建硬件冗余。这有助于防止下述状况的发生，即某个服务器上的错误导致整个系统不可用。这类系统在服务连续性和/或安全数据采集很重要时尤为有用。



在 I/O 服务器冗余的情况下，备用服务器将平行保存在主服务器中。如果检测到硬件错误，则备用服务器可以重新控制设备通信，而系统几乎不会中断。您也可以使用冗余 I/O 服务器分解处理负载。

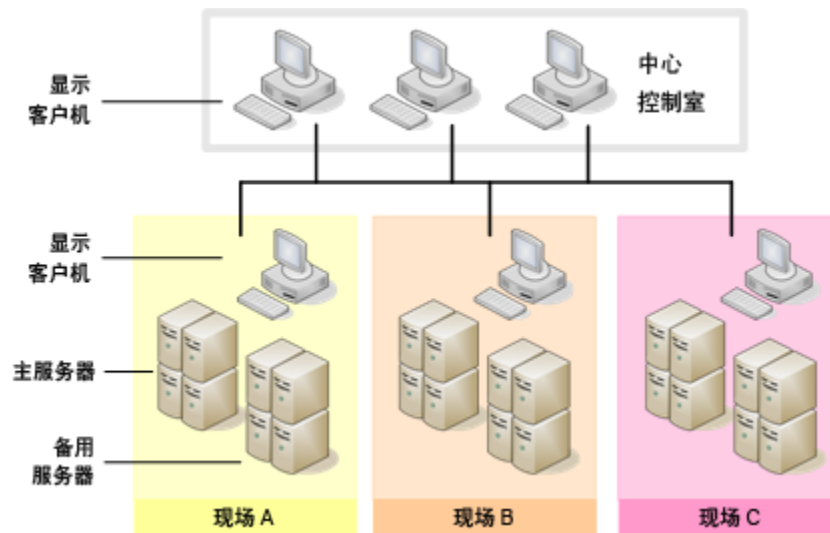
报警、报告和趋势服务器也可以作为冗余服务器实现。这可保证在主服务器不正常的情况下，客户机继续访问来自备用服务器的数据。Vijeo Citect 将在两个服务器上保存相同的数据。

在下图中，主 I/O 服务器和备用 I/O 服务器独立进行部署，同时，报警、趋势和报告服务器作为独立进程在通用主计算机和备用计算机上运行。在这种情况下，整个系统将作为一个群集进行配置。



集群式控制系统

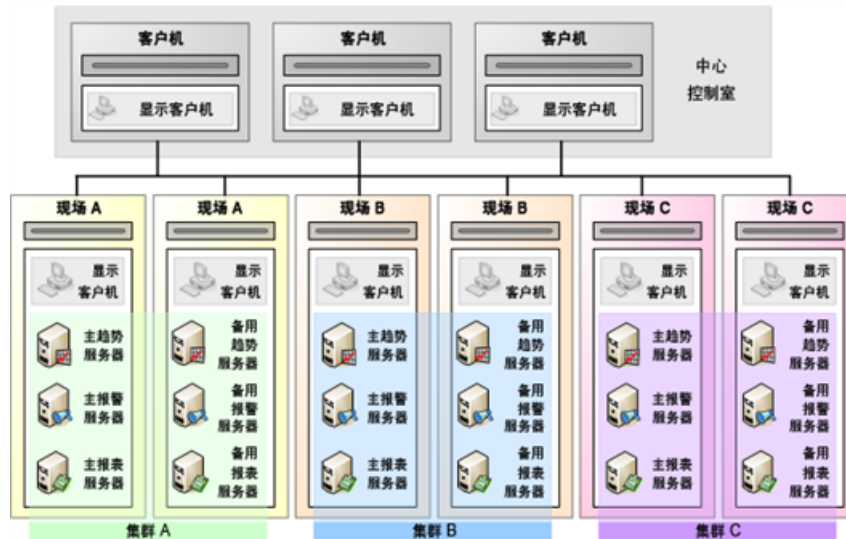
在此方案下，系统被组织为由本地操作人员控制并且由本地冗余服务器支持的分散站点。同时，存在一个管理级别，要求系统中的所有站点要从一个中心控制室同时进行监控。



每个站点在工程中都表现为组合了其主服务器和备用服务器的一个独立集群。每个站点的客户端只关心其本地集群，而中心控制室的客户端则能够查看所有集群。

中心控制室方案的部署相当简单，因为每个站点都可以独立面向自己的集群。控制室本身只需要控制客户端。

服务器的部署可以按下列方式安排：



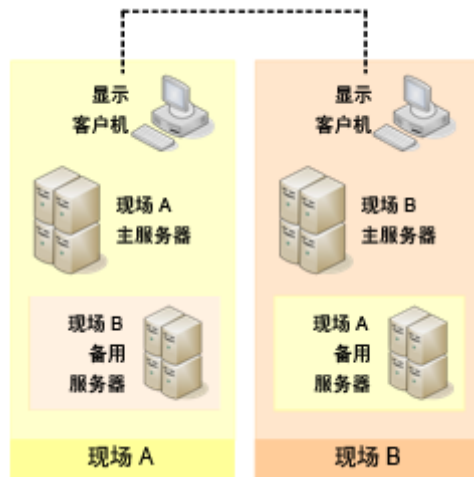
Vijeo Citect 对动态集群的支持意味着每个站点在需要时都可以从中心控制室进行监控。例如，如果特定站点的操作人员只在正常上班时间工作，则监控在下班后就可以切换到中心控制室。

冗余和分布式控制系统

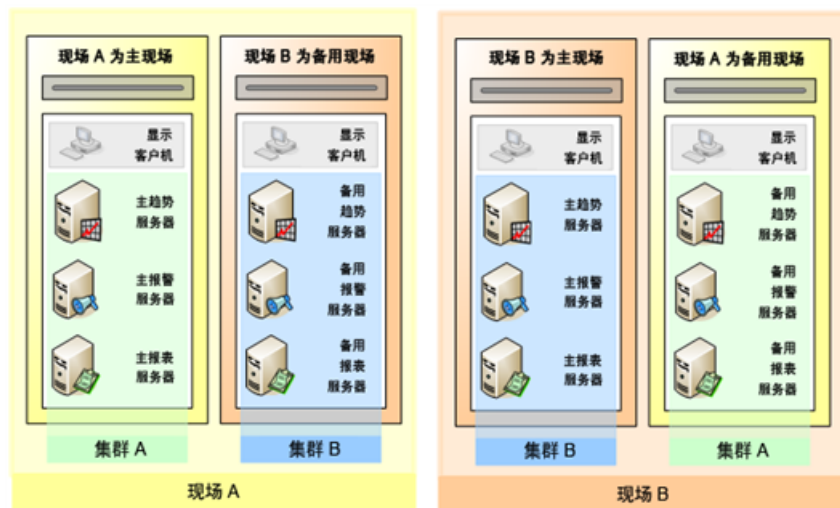
在此方案中，工程代表一系列本地操作站点，每个都包含其自身的服务器和客户机。例如，某个水分布系统中的一些泵站，或生产流程中的多条生产线。但是，要求对其中的某个站点进行监控，以便在系统发生故障时能够继续。

要实现这一要求，可通过以下方法：将主服务器和备用服务器分布于多个不同站点，或将所有备用服务器置于中心位置。

可以使用群集来定义每个站点上的不同服务器的角色，而这些服务器可以通过在每个客户机上运行的普通工程来查看。也就是说，如果在某个站点上发生系统故障，则可以通过站点 B 来监控站点 A，反之亦然。



上面的示例需要创建两个群集，这样，工程就可以包含两组主服务器和备用服务器。这两个群集代表服务器的冗余对，可以通过以下两个站点进行部署：



这些群集有益于在配置时保持工程的逻辑架构，而不管冗余服务器对的分布是否异常。

负载分担系统

系统组件在不同计算机和 CPU 之间的负载分担意味着可以在多台计算机之间分摊可能给一个系统造成压力的工作负荷，从而更好地利用可用的基础设施。

例如，管理报警会给 CPU 的性能造成严重压力，而趋势数据可以使用大量的磁盘空间。通过将趋势和报警服务器分配给共享计算机上的不同进程，可以将报警服务器作为备用趋势服务器，从而有效利用空闲磁盘空间。

可采用此方法提高网络性能、减少数据存取时间以及提高一般系统稳定性。

如果引入了集群，还可以灵活地在一台计算机上运行相同类型的多个服务器。只要客户机能够访问工程中配置的所有集群，一系列服务器是否分布在许多集群上则无关紧要。

在下面的示意图中，两个服务器被配置为相互的备用单元，从而支持两个系列的冗余趋势和警报服务器。



两台机器都有均匀平衡的趋势和警报服务器，从而确保了 CPU 和磁盘空间的有效利用。通过在两个集群上分布这些服务器，它们相互之间也能够作为彼此的冗余单元。这样可将所需计算机的数量从最多八台减少为只需要两台。

使用 Vijeo Citect

本部分包含有关使用 Vijeo Citect 的信息，并介绍以下内容：

[计划工程](#)

[管理工程](#)

[保护工程](#)

[配置系统](#)

[实现集群](#)

[将冗余内置于系统](#)

[标记过程变量](#)

[链接、导入和导出标签](#)

[定义和绘制图形页面](#)

[配置和处理报警](#)

[配置事件](#)

[使用累积器](#)

[记录和趋势数据](#)

[了解统计过程控制](#)

[报告信息](#)

[使用安全性](#)

[使用标签](#)

[使用设备](#)

[与其它应用程序交换数据](#)

[使用精灵和超级精灵](#)

[使用多语言工程](#)

[使用 OPC 服务器 DA2.0](#)

[与 I/O 设备通讯](#)

[使用通讯快速向导](#)

[生成项目](#)

Chapter: 8 规划工程

本章介绍 Vijeo Citect 系统的规划阶段。

通过有计划地设计和配置系统，可以实现产品功能和性能的最优使用，同时可帮助您满足生产设备的所有要求。这还可以避免配置工程过程中出现不必要的返工。

规划系统时考虑到以下几个方面很重要：

1. [工厂的实际布局](#)
2. [操作要求](#)
3. [工程设计](#)
4. [构建工程](#)
5. [部署](#)

工厂的物理布局

需要考虑要实施 Vijeo Citect 的工厂的物理布局。

此信息将帮助确定工厂的架构，以及它的许多操作要求。还可以通过此信息评估可用设备，以确定是否需要添加或修改。

在检查工厂的物理布局时，应考虑以下因素：

地理位置

工厂的物理布局，包括工厂是否跨越多个地理位置或特定功能区域，如平行运行的多个生产线。

机器

将由系统监控的现场设备（机器、物理连接和设备）。

现有计算机硬件

当前您的设施中存在的、能够支持 Vijeo Citect 的客户机-服务器架构的计算机。必须考虑现有设备的规格和限制，以便确定是否能够支持您的系统的操作要求。

网络配置

支持系统及其与工程设置通讯的网络的当前配置。其中包括使用的协议、系统的性能和安全性。

另请参见

[操作要求](#)

操作要求

您可以通过为工程开发一个操作要求集来定义一个需求和目标列表，您的系统必须支持其中的需求和目标才能有效监控生产。

确定操作要求时需要考虑的因素包括：

- 架构
- 安全性
- 可靠性
- 监视
- 数据收集

架构

生产流程

需要考虑生产设备中的操作过程，以确定如何以逻辑方式表示这些过程并在工程中支持这些过程。如果这些过程彼此依赖，则还需要考虑如何管理它们之间的交互，特别是在出现意外情况时。

安全性

用户访问

用户需要考虑谁要使用系统以及他们需要对工程的哪个部分具有访问权限如果关注的是安全性，则可以对以个人为基础或通过创建用户组（例如，操作员、经理、管理人员）的访问进行限制。

另请参见

[维护用户记录](#)

控制区域

根据工程的构造方式，您可能需要根据功能区域（例如，收货、分派和处理）或地理位置区域定义工程中的区域。控制区域与 Vijeo Citect 的基于用户的安全性一同工作来有效管理生产流程的控制。

另请参见

[定义区域](#)

可靠性

生产过程的本质将决定系统可靠性的重要性。请注意以下各项：

- 对不间断操作的需求
- 宕机的影响和成本
- 对收集和保护系统数据的需求
- 报警情况的严重性。

这将有助于确定您的工程是否应该包含冗余、所需的冗余类型以及实现冗余的最佳方案。有关详细信息，请参考 [冗余](#)。

监视

系统监视是 SCADA 系统的关键功能，需要根据人员和生产流程之间的交互考虑。

用户需要考虑数据传输是否对时间要求很急。例如，报警条件应实时出现，趋势数据可以稍微延迟后传递，而维护数据可以根据需要进行累积和查看。

系统还需要在不同级别进行监视：从机器操作员到控制室人员管理工厂范围的流程。对于每个监视级别，请考虑需要显示的数据和需要标记的特定性能和诊断条件。

数据收集

考虑需要从生产流程中收集的数据的类型及其使用方式。根据您的要求，Vijeo Citect 可以收集：

- 生产数据
- 购买要求
- 批量处理统计数据
- 设备的状态和性能数据
- 维护调度信息
- 过程的性能数据
- 可视性动态分析数据

还应认真考虑累积数据的近似数量评估，因为它将显著影响计算机硬件和网络性能要求。

有关详细信息，请参见 [日志和趋势数据](#)。

另请参见
[工程设计](#)

工程设计

在开发出一组清晰的操作要求之后，您需要规划如何设计工程以最好地满足这些要求。设计工程时，应考虑以下问题：

命名标准

通过采用命名标准，可以使用传递有用信息（如组件的位置或类型）的有意义名称配置工程组件。要使用的标准取决于对系统操作员有用的信息类型。命名标准有助于提高整个工程的一致性，更容易快速识别组件，以及减少重复和用户培训。命名标准对设备、变量标签、报表、图形对象及页面非常有用。

页面模板

页面模板是可以用来建立工程显示屏幕（图形页面）的预定义的页面布局。模板让使用者快速创建新页面，并允许运行系统有一致的外观和感觉。它们能合并标准导航并且支持所有页面通用的工具。Vijeo Citect 包括许多标准模板，并且使用者可以设计适合于系统需求的新模板。

另请参见
[使用页面模](#)

精灵和超级精灵

精灵与对象模板相同，您可以在上面放置图形画面以帮助简化许多类似设备的配置。它们会对功能和图形元素进行分组，并使用可配置的字符串替换将设备信息显示为特定标签或表达式的占位符。随后，信息将在运行时显示。

超级机灵是可以在运行时向其传递设备专有信息的精灵。超级精灵对于动态控制很有用，如可用于控制许多设备的弹出式开关。

另请参见
[使用精灵和超级精灵](#)

集群

通过集群，可以将互不相干的 Vijeo Citect 服务器组件组合在一个工程中，从而能够同时监控多个系统。

最适当的配置将取决于要部署的解决方案的**要求**以及部署它的**环境**。

一些典型的集群配置如下：

- [单机系统](#)
- [分布式 I/O 系统](#)
- [客户端-服务器系统](#)
- [冗余服务器系统](#)
- [集群式控制系统](#)
- [冗余和分布式控制系统](#)
- [负载分担系统](#)

Vijeo Citect的集群实施允许灵活部署可以从不同集群动态访问数据的图形画面。调用画面时，可以为它分配一个集群环境，而且该画面上的任何注释都将指定到同一集群，除非明确为它们指定了集群。参见 [关于集群环境](#)。

另请参见
[典型系统方案](#)
[集群的规则](#)

关于集群上下文

许多Vijeo Citect项要求指定集群，以使它们可以正确发挥作用。可以在调用或显示项时显式指定此 `ClusterName`，或者项可以使用调用进程或页面的缺省集群上下文。如果您的系统只有一个集群，则不需要指定任何集群名称。请参阅 [集群上下文规则](#)。

因此，集群上下文是用于解析标签以及执行表达式和 `Cicode` 函数的缺省集群。如果使用标签，则系统通过为标签名称添加集群名称前缀来显式提供集群。对于内置的 `Cicode` 函数，`ClusterName` 通常为可选参数。

服务器进程（报警、趋势、报告）将其缺省集群上下文设置为其各自的集群，因此，包含变量标签（不显式提供集群）的报警定义会尝试将这些标签解析到各自的集群中。从服务器代码启动的 `Cicode` 任务将继承服务器集群，除非显式提供了集群。

缺省情况下，图形页面不具备集群上下文。页面的集群可以在页面外观属性中静态提供，可以从前面的上下文中继承，也可以使用 `Cicode` 函数（用于显示页面）中的可选参数动态应用。从页面启动的所有 `Cicode` 任务都将继承该页面的集群上下文。

另请参见

[集群上下文规则](#)

集群环境规则

1. 单集群系统始终使用该集群，并且不需要使用集群前缀。
2. 报警、报表或趋势服务器环境中引用的变量标签隐含解析到该集群，除非使用集群前缀另外明确规定，如 `ClusterName.TagName`。
3. 在报警、报表或趋势服务器环境中调用的 `Cicode` 缺省为在该服务器的集群环境下运行。
4. 服务器到服务器连接（例如，SPC 报警）缺省限于该集群内。
5. 客户端画面可以有与其关联的集群环境。
6. 可以为客户端画面配置集群环境，也可以从以前（或父级）画面继承集群环境。
7. 已为客户端画面配置好的集群环境可以在运行期间动态覆盖。
8. 客户端画面没有缺省集群，而且缺省情况下不继承多集群系统下的环境。
9. `TaskNew Cicode` 函数会继承呼叫者的（画面或任务）集群环境，除非明确覆盖。

另请参见

[关于集群环境](#)

包含工程

如果您有一个大型生产环境，您可以通过将您的工程设计为更小“包含”工程的集合来简化您系统的配置和管理。

包含工程可以独立运行，不过，它们在运行期间需要共享资源并在工作上相互依存。这意味着，您可以创建并测试表示工厂的各个功能或物理部分的工程，然后逐步将它们联系起来。随后的维护管理对生产的影响就会很小。

有关详细信息，请参考 [包含工程](#)。

冗余

可在系统的不同级别实现冗余，具体取决于工程的可靠性要求。以下冗余类型可用：

设备冗余

可在 Vijeo Citect 中为设备配置多个数据路径。因此，如果主路径变得不可用，仍然可以通过辅助路径监视数据。

服务器冗余

可以对主要和备用报警、报表以及趋势服务器进行配置，在主要服务器不可用于处理客户机请求的时候，将请求发送给备用服务器进行处理。

局域网 (LAN) 冗余

为了避免在主网络不工作时造成服务中断，可以实现冗余 LAN，提供一个指向服务器的备用路径。

另请参见

[在系统中建立冗余](#)

构建工程

确定工程需要和设计后，便可以开始在 Vijeo Citect 中实现设计。

下列主题将帮助您确定需要实现的工程组件和选项。

工程

首先需要创建一个新工程，并熟悉如何针对该工程执行存储、包含和归档等任务。

运行此工程之前，它的编译过程会提醒您注意配置中的任何错误。

另请参见

[构建 Vijeo Citect 工程](#)

[管理工程](#)

[编译工程](#)

设置 I/O 设备通信

Vijeo Citect 可配置为与许多不同供应商提供的 I/O 设备进行通信。要建立与 I/O 设备的通信，您需要执行下列步骤：

- 安装相关设备驱动程序
- 配置设备所需的硬件和软件
- 设置测试工程来测试通信信道
- 为您要与之通信的 I/O 设备上的每个数据点配置变量标签。
- 手动或使用通信快速向导在工程中配置设备

另请参见
[使用通信快速向导
与 I/O 设备通信
标记过程变量](#)

图形组件

操作员可通过图形组件查看 Runtime 系统并与之交互。可将图形画面设计为通过相关的监控选项在不同系统区域或层次提供操作员。

要创建符合操作要求的图形组件，您应该熟悉如何创建图形画面、使用画面模板以及配置精灵和超级精灵等图形对象。

另请参见：
[定义和绘制图形画面](#)
[使用精灵和超级精灵](#)
[使用对象](#)
[定义通用对象属性。](#)
[了解对象类型](#)

报警

Vijeo Citect 报警系统监视您的生产流程并提示操作人员留意可能需要注意的意外事件。

您可能需要配置两种类型的报警：

- **硬件报警** - 提醒您设备不正常或部分不正常
- **已配置的报警** - 允许您为设备指定相关报警条件（例如，用于监视特定设备的级别、温度或状态的变量标签值）。根据需要设置的报警条件类型，有七种类型的已配置报警。

为了帮助操作人员对报警进行处理，可以创建可提供报警信息（如操作人员为改正情况所应采取的措施）的图形页面。

另请参见
[配置和处理报警](#)
[已配置的报警](#)
[设置报警显示格式](#)

数据收集

位于 Vijeo Citect 中的数据并入了以下两个主要方面：

- **趋势** - 可通过趋势系统收集和监视现场数据。可以定期收集数据，也可以在发生特定事件时收集，具体取决于您的要求。随后，可将数据存入磁盘以备分析之用或显示在图形或报表中。要在系统中使用趋势，需要熟悉如何配置趋势标签以及如何如何在图形或报表中显示趋势数据。
- **报表** - 报表提供有关工厂和过程状态的信息。为使报表符合操作要求，可以使用以下信息对其进行配置：

- **周期/触发:** 报表可以针对请求运行、定期运行或在发生特定事件时运行。
- **报表格式:** 可以使用文本编辑器创建指定报表显示方式的文件。
- **报表输出:** 报表可以输出到文件、设备或显示在图形画面上。

另请参见

[日志和趋势数据](#)
[报表信息](#)

用户和区域

您可以设计系统的安全性，包括以下安全功能：

- **用户** - 允许您限制访问运行时系统的用户帐户。所有用户必须使用用户名和密码获取访问权限才能登录系统用户帐户可以设置为个人帐户，也可以设置为用户组帐户。
- **区域** - 区域允许您定义系统中的地域边界或功能边界。然后，您就可以控制用户访问工程的不同部分并执行各任务。

另请参见

[利用安全性](#)

系统组件

Vijeo Citect 包含以下系统组件，这些组件提供了更多用于监视、控制和用户交互的选项：

- **命令和控制** - 可配置的键盘命令和滑钮控件，允许操作员与 Runtime 系统交互。
- **事件** - 可以配置事件（如变量标签或表达式）以触发特定动作（如命令）。
- **累积器** - 累积器跟踪递增的 Runtime 数据。数据以变量标签的形式存储在 I/O 设备中，并会在触发器处于活动状态期间定期更新。
- **分析过程控件** - SPC 允许您通过收集和解释与产品关联的过程变量来跟踪质量。
- **标签** - 可针对通常执行的命令和表达式配置系统范围的替换。
- **设备** - 可以将高层 Vijeo Citect 数据（包括报表和日志）传输到其它系统元件，如打印机、数据库或文件。
- **远程访问** - 可以采取以下方式对工程进行远程或无线访问：
 - **Vijeo Citect Web 客户端** - 可通过 Vijeo Citect Web 客户端查看 Web 浏览器中的实时工程。

另请参见

[定义命令和控件](#)
[配置事件](#)
[使用累积器](#)
[了解统计过程控件](#)
[使用标签](#)
[使用设备](#)
[使用设备](#)
[Vijeo Citect Web 客户端](#)
[通过互联网运行系统](#)
[与其它应用程序交换数据](#)

部署

在创建工程之后，您需要对系统中的每台计算机进行配置。配置信息保存在每台计算机的 `citect.ini` 文件中。信息包括：

- 计算机在 Citect 网络中具有的角色
- 正在运行的工程
- CPU 配置
- 为每个组件启用的 Citect 事件
- 启动时为每个组件运行的 Cicode
- 集群配置
- 应用的安全设置

计算机设置向导将显示一系列页面，您可以通过这些页面配置这些设置。选定的选项将写入到 `citect.ini` 文件中。作为运行工程前的最后步骤，应该在每台计算机上运行向导。

另请参见

[运行计算机设置向导](#)

将 Vijeo Citect 设置为 OPC 数据源

Vijeo Citect OPC 服务器允许您通过任何 OPC 客户机应用程序（v1.0 或 v2.0）访问 Vijeo Citect 运行环境中所有可用的数据（例如，来自 PLC 和数据库的）。

使用 Vijeo Citect 来监控工厂时，就会从 PLC 收集数据并显示在运行环境中。OPC 客户机可通过与 OPC 服务器的接口访问设备和标记信息，而该接口又与到运行时的 CtAPI 接口进行交互。有关如何配置 OPC 服务器的详细信息，请参阅 [使用 OPC 服务器 DA2.0](#)

Chapter: 9 管理工程

Vijeo Citect 是基于工程的应用程序。帮助的这一部分说明与创建、存储和维护工程相关联的管理任务。其中包括：

- [管理工程](#)
- [存档工程](#)
- [包含工程](#)
- [使用工程编辑器](#)
- [在工程中使用查找和替换功能](#)

管理工程

本节的帮助内容介绍如何使用 Citect 浏览器来实现对工程的管理。包括下列主题：

- [创建工程](#)
- [编辑现有工程的属性](#)
- [复制工程](#)
- [打印工程详细信息](#)
- [删除工程](#)
- [链接工程](#)
- [时间同步](#)

创建工程

创建新工程：

1. 启动 Citect 工程管理器。
2. 点击**新建工程**按钮，或者从文件菜单中选择**新建工程**。
3. 为要创建的工程输入**名称**（必需）。
4. 输入**描述**以及存放新工程文件的**位置**。
5. 选择一个**模板风格**和**模板分辨率**以设置图形画面的外观。
6. 点击**确定**即可创建工程，也可以点击**取消**。

另请参见

[新建工程对话框](#)

新建工程对话框

使用此对话框，您可以 [创建新工程](#)。要创建新工程，请在**名称**字段（其他字段可选）中输入值，然后点击**确定**。

创建工程后，用户就可使用“工程属性”对话框来查看并编辑其属性，如下所示。

名称

工程所独有的一个名称。工程的名称长度限制为 64 个字符。它可以包含除分号（;）或者单引号（'）之外的任何字符。由于工程名称是一个唯一标识符，因此 Vijeo Citect 不允许用户再使用相同的名称创建或者恢复另一个工程。

描述

对于工程的说明。此字段可用于解释工程的作用。我们强烈希望用户能够填写此字段。

保存位置

保存工程文件的目录路径。在“名称”字段输入名称后，会在“保存位置”字段中自动生成该目录。用户可以采用手动输入，或者点击浏览的方法来修改该字段中的内容。

[画面缺省值] 模板风格

系统在运行时，图形画面的风格（外观）。用户所选择的风格也是用户添加到该工程中的所有新画面的缺省风格。用户可以通过图形编辑器中的画面属性，来更改现有画面和模板的风格。

多数用户更愿意采用标准风格。用户可以在“图形”、“模板”下的“包含工程”中查看预定义的风格。

[画面缺省值] 模板分辨率

标准图形画面（诸如报警画面以及标准趋势画面）的缺省屏幕分辨率：

屏幕类型	屏幕宽度（按像素）	屏幕高度（按像素）
VGA	640	480
SVGA	800	600
XGA	1024	768
SXGA	1280	1024
User	****	****

[画面缺省值] 显示模板的标题栏

确定是否显示 Windows 的标题栏（在图形画面的上方）。标题栏中包括窗口的标题，最大化、最小化和关闭按钮（在标题栏的最右端），以及控制菜单按钮（在标题栏的最左端）。

要以全屏的方式来显示画面（不显示标题栏），画面的大小必须同屏幕的大小相同（或者要大些）。如果画面小于屏幕，即使在全屏模式下，将仍然显示标题栏。标准的模板风格对于两种画面大小都适用。

[画面缺省值] 背景颜色

所有新建图形画面显示的背景颜色。

编辑现有工程的属性

要编辑现有工程的属性，请执行以下操作：

1. 打开 Citect 工程管理器。
2. 从列表中选择一个工程。
3. 点击**属性**按钮，或从**文件**菜单中选择**工程属性**。
4. 在“工程属性”对话框中对属性进行编辑。
5. 点击**确定**以保存更改，或点击**取消**中止。

属性

工程具有 [常规工程属性](#)和 [画面工程属性](#)。

工程常规属性

（常规）名称

工程的名称。此名称与创建工程时使用的名称相同。工程的名称长度限制为 64 个字符。它可以包含除分号（；）或者单引号（'）之外的任何字符。由于工程名称是一个唯一标识符，因此 Vijeo Citect 不允许用户再使用相同的名称创建或者恢复另一个工程。最大长度为 64 个字符。

（常规）状态

工程的状态。该状态可以是**已编译**或**未编译**中的一种。

（常规）位置

保存工程文件的目录路径。该域无法被修改。

（常规）说明

对于工程的说明。此域可用于解释工程的作用。我们强烈希望用户能够填写此域。最大长度为 255 个字符。

（常规）主要修订

当工程首次被创建时，Vijeo Citect 将该属性设置为一 (1)。用户可以利用该属性来对工程的重要修改进行跟踪。您可以使用递增的修订历史（如 1、2、3，或 . . A、B、C. .）。最大长度为 4 个字符。

（常规）次要修订

当工程首次被创建时，Vijeo Citect 将该属性设置为零 (0)。用户可以将该属性同主要修订（Major Revision）相结合来对工程的整个开发过程进行跟踪。最大长度为 4 个字符。

（常规）日期和时间

在开始阶段，Vijeo Citect 会把这些域设置成为该工程创建时的日期和时间。当这些域同修订域相结合时就会变得十分有用了。最大长度为每项 20 个字符。

（常规）工程 ID

为工程建立的一个专有的编号。工程编号可以从 1 到 1022。

如果用户输入一个已经被另一个工程使用的 ID 号，Vijeo Citect 将会在对工程进行编译时检测到这一情况。

工程的编号是 OPC 驱动程序从标签读取数据或者向标签写数据时所使用的独有的标识（对象 ID (OID)）的一个组成部分。

如果用户没有指定某个工程编号，Vijeo Citect 将会在用户下次在 Citect 浏览器中选择该工程时，或者下次对工程进行编译时自动产生一个编号。最大长度为 4 个字符。

注意：如果您键入 0，您的工程 ID 将会在下次对工程进行编译时自动设置。

（常规）只读

指定不能对工程进行改动。当该选项被选中时，如果试图对工程进行修改，系统将会弹出一个消息框提示用户将该选项禁用。

注意：如果您对任何属性进行更改，必须点击**确定**按钮将更改保存到工程中去。

工程页面属性

（页面缺省值）[模板] 分辨率

标准图形页面（诸如报警页面以及标准趋势页面）的缺省屏幕分辨率：

显示器类型	屏幕宽度（按像素）	屏幕高度（按像素）
VGA	640	480
SVGA	800	600
XGA	1024	768
SXGA	1280	1024
User	****	****

注意：您可以在创建页面的时候更改其页面的默认设置或者在此后的任何时候进行更改。

（页面缺省值）[模板] 风格

Runtime 系统中的图形页面的风格（外观）。用户所选择的风格也是用户添加到该工程中的所有新页面的缺省风格。用户可以通过图形编辑器中的页面属性，来更改现有页面和模板的风格。

多数用户更愿意采用标准风格。您可以在**图形 | 模板**下的包含工程中查看预定义风格。

注意：您可以在创建页面的时候更改其页面的默认设置或者在此后的任何时候进行更改。

（页面缺省值）[模板] 显示标题栏

决定是否显示 Windows 的标题栏（在页面的上方）。标题栏包括窗体的名称、最小化、最大化和关闭按钮（在标题栏的最右端），以及控制菜单按钮（在标题栏的最左端）。

如果要查找一个自己建立的模板，用户必须确保输入了正确的要以全屏的方式来显示页面（不显示标题栏），页面的大小必须同屏幕的大小相同（或者要大些）。如果页面的尺寸小于屏幕的大小，即使在全屏的模式下，标题栏将仍然被显示出来。标准的模板风格对于两种页面大小都适用。

注意：您可以在创建页面的时候更改其页面的默认设置或者在此后的任何时候进行更改。

（页面缺省值）背景颜色

所有新图形画面背景中显示的颜色。

复制工程

可以将一个工程的内容复制到现有工程或新工程中。

复制工程：

1. 打开 Citect 工程管理器。
2. 选择**复制**图标，或从**文件**菜单中选择**复制工程到**。
3. 在“复制工程”对话框中，从**工程名称**下的下拉列表中选择源工程。
4. 选择要复制到的现有目标工程或选择一个新工程。
5. 点击**确定**以复制该工程，或点击**取消**。

另请参见

[复制工程对话框](#)

复制工程对话框

使用此对话框，用户可以将某个工程中的全部内容 **复制**到另一个工程中去。要对一个工程进行复制，指定源[从]以及目标[到]工程，再点击**确定（OK）**按钮。

[从]工程名称（[From] Project name）

需要进行复制的源工程的名称。如果存在多个工程，用户可以从下拉列表中选择工程名称。

[到]（已经存在或者新建的）工程（[To] (Existing or New) project）

用户可以复制到一个已经存在的或者一个新建工程名称以及位置中。

- **已存在工程：**源工程覆盖（替换）一个已存在工程名称之下已存在工程位置。
- **新建工程：**源工程将被复制到新建工程名称的新建位置。一个新建工程必须被赋予一个当前没有使用过的新名称，该名称应遵循以下所列出的详细要求。

[到]名称

被复制到的目标工程的名称。

当复制到某个已存在工程中时，用户必须从下拉列表中选择已存在工程名称。

当复制到某个新建工程中时，用户必须为该工程新建一个唯一的名称。该工程名称的长度限制为 64 个字符，而且可以包括除分号 (;) 或者单引号 (') 之外的任何字符。由于工程名称是一个唯一标识符，因此 Vijeo Citect 不允许用户再使用相同的名称创建或者复制另一个工程。

当创建了新工程后，用户可以通过工程属性来更改其名称。

在将工程复制到一个已存在工程位置中时，在开始对源工程进行复制之前，用户可以通过同时选择 **复制前清除 (Clear Location Before)** 以及 **清除子目录 (Clear Sub-directories)** 选项来删除目标工程中已经存在的文件。这样将删除可能对恢复之后的工程进行干扰的许多文件。如果用户在复制之前不对工程所在的位置进行清除，在目标工程中只有通用的文件会被覆盖。

[到]复制之前清除（Clear location before copying）

在对源文件进行复制前首先删除所选的已经存在的目标工程中的文件。这样将删除可能对恢复之后的工程进行干扰的许多文件。

[到]清除子目录（Clear subdirectories）

在对源文件进行复制前首先删除所选的已经存在的目标工程的子目录中的文件。这样将删除可能对恢复之后的工程进行干扰的许多文件。

保存位置（Location）

工程文件所保存在的目录路径。在键入 **名称 (Name)** 字段后，在 **位置** 字段将会自动产生目录。用户可以采用手工键入或者点击 **浏览 (Browse)** 按钮的方法来修改该字段中的内容。

检查确认对话框中的工程名称和位置是否正确。单击 **是** 开始复制工程，或者单击 **否** 取消复制。

打印工程详细信息

您可以对当前工程的配置元素（数据库记录、页面、Cicode 文件等）进行打印。Vijeo Citect 将打印到 Windows 缺省打印机。

要打印工程数据库详细信息，请执行以下操作：

1. 打开 Citect 工程编辑器

2. 从**文件**菜单中选择**打印**。
3. 使用**打印选择列表**选择要打印的元素。
4. 点击**确定**开始打印，或点击**取消**中止。

在开始打印数据库之前，先打印一小部分来测试打印效果。可以通过从**工具**菜单中选择**选项**来更改默认字体、字号和页面大小。有关其它打印选项，请参见 Windows 文档。

另请参见

[打印（工程详细信息）对话框](#)

打印（工程详细信息）对话框

利用此对话框，您可以 [打印](#) 当前工程中的配置元素（数据库记录、图形页面、Cicode 文件等）。点击**确定**保存选择，或点击**取消**中止打印。

[打印选择]

列出了工程中所有能够被打印输出的元素。要选择（或者取消选择）某元素进行打印，点击相应的复选框；复选标记表明它将被打印。

点击**全选**选择列表中的所有项目，或点击**全部取消选择**清除所有选择。

[选项]图形页包含在打印选择区域

指定某个特定的页面进行打印。使用下拉列表从工程中选择某个页面。选择<**所有页面**>条目来打印工程中所有的页面。

[选项] 根据图形页面对打印输出分组

将对对象数据库信息连同相关页面一起打印。如果没有对该选项进行设置，对象数据库中的信息将只作为连续的列表被打印，其中的页面仅是一些相关信息。

用户只能打印当前工程的内容。合并的工程将不会被打印。用户可以在工程编辑器的**选项**部分（位于**工具**菜单中）指定打印字体、字号和页面大小。

删除工程

要删除现有工程，请执行以下操作：

1. 打开 Citect 工程管理器。
2. 从列表中选择工程。
3. 从**文件**菜单中选择**删除工程**。
4. 会出现一个对话框，要求您确认是否继续。点击**是**删除工程，或点击**否**取消。

无法删除当前打开的或已安装的工程。也无法删除 Vijeo Citect 提供的包含工程。

注意：您无法恢复没有备份的已删除工程。

另请参见

[链接工程](#)

链接工程

安装在同一网络中不同计算机上的 Vijeo Citect 可以共享相同工程。在一台计算机上创建工程后，同一网络中的其他计算机可以链接到该同一工程，但是该工程必须位于共享或网络驱动器上。一旦链接之后，就可以在本机 Citect 浏览器中查看该远程工程，并且可以通过网络进行编辑和编译。只有一个工程的版本是一直保留的，并且该版本应保留在最初创建该工程的计算机上。

注意：即便工程位于网络中的远程计算机上，开发人员也可以通过与该工程的链接正常访问和控制该工程。

注意：虽然链接的工程可能位于网络中的远程计算机上，但也是有可能将其删除的。您应该解除通过网络链接的工程而不是将其删除。

链接的工程将不包括在任何其他工程的编译中，除非已经专门从工程编辑器中将它们包括在该工程内。

有关详细信息，请参见 [包含工程](#)。

链接到一个工程：

1. 打开 Citect 浏览器。
2. 点击**添加链接**按钮，或从**文件**菜单中选择**添加工程链接**。
3. 利用“选择工程目录”对话框选择一个工程保存位置。
4. 点击**确定**链接到该工程，或点击**取消**。

如果新建工程与某个已经存在的工程具有相同的名称，在进行下一步操作之前，系统会提示更改该名称。在“工程属性”对话框中对属性进行编辑。

删除与某个工程的链接：

1. 打开 Citect 浏览器。
2. 从列表中选择工程。
3. 点击**删除链接**按钮，或从**文件**菜单中选择**删除工程链接**。
4. 继续删除操作之前系统会进行提示。要删除链接请单击**是**，否则单击**否**取消删除操作。

另请参见

[包含工程](#)

时间同步

Vijeo Citect 以前的版本使用了一种基于消息的时间同步服务器，确认运行 Vijeo Citect 工程的计算机上的时钟始终保持时间同步。要在 Windows Vista 中开启用户访问控件 (UAC) 的情况下，使用标准用户权限运行 Vijeo Citect，需要替换现有的时间服务器功能。

注意：这同样会使 Cicode 函数 TimeSet 过时，因此在现有代码中不应使用该函数。

为了维持时间同步，Vijeo Citectv7.10 将安装一种称作 TimeSyncService 的 Windows 服务，可在内置 LocalSystem 帐户中运行。本服务的目的在于维持本地计算机上的时间，而不使用其他（一个或多个）时间源。时间源是运行时间服务的计算机。

Vijeo Citect 提供的时间同步实用程序可帮助您配置时间同步，并将服务作为管理环境的一部分进行控制。该对话框将存储并读取 TimeSyncConfig.xml 文件中的设置，该文件默认安装在 Vijeo Citect Config 目录。有关配置文件位置的信息，请参见 [配置文件和工程文件的新位置](#)。使用该配置实用程序，您可以为配置文件指定一个备用路径，如网络共享。如果您拥有多台计算机且配置数据相同，则这一实用程序非常有用；如要更改任何设置，只需在一台计算机上更改即可。

要显示“时间同步”对话框，可打开 Citect 浏览器，从“工具”菜单中选择“时间同步”菜单项。

另请参见

[时间同步对话框](#)

时间同步对话框

下表描述了“时间同步”对话框中的可用字段。

字段	描述
当前状态	按照计算机管理的服务属性中的显示方式，显示 TimeSync Windows 服务的状态。如果该服务已停止则可单击“启动服务”按钮，而如果服务正在运行中则可单击“停止服务”按钮。如果服务被标识为禁用，则按钮同样被禁用。要启用服务，请使用 Windows 管理工具来自动或手动启动。
启动类型	确定访问是手动启动还是自动启动。如果访问被禁用，请使用 Windows 管理工具来启用自动或手动启动。
TCP/IP 端口	服务将用来侦听来自客户端的连接的网络端口。
上次同步	显示 LastSyncTime 注册表设置的值。上次成功同步的本地时间。
当前本地时间	显示本地计算机的当前时间，每隔 1 秒更新一次。
记录信息事件	控制服务是否将 'Information' 类型的事件写入到事件日志。缺省是未选中，这样就只记录警告和错误。
保持此计算机时间同步	选中此复选框可将使计算机成为一个时间客户端。这允许您针对要同步的时间输入轮询时间和时间服务器的列表。
启用时间同步	指定是否启用时间同步。
同步间隔	输入一个 1 到 168（含）之间的数字（按小时计），指定同步发生的频率。缺省参数值为 24。
立即同步	点击可立即同步。
与第一个可用的计算机同步	显示一系列计算机以及这些计算机的当前时间（如果可用）。每隔 1 秒更新一次显示。

字段	描述
添加按钮	显示一个对话框，输入要添加的服务器的名称。
删除	从上面的列表中选择一台计算机，并点击“删除”，将其从列表中删除。

注意： 在向列表中添加时间源时，将显示该机器上的当前时间，提供的服务运行在远程机器并侦听同一端口号。如果“不可用”则服务不运行，或运行但使用不同的端口号，或该端口号被防火墙阻止。列表框中该列作为诊断功能提供，以确保输入的机器名可以进行同步。此框中显示的时间仅是个大概的时间。

另请参见

[时间同步](#)

归档工程

配置系统后，应该备份（会归档）工程。这将避免在主存储故障或不可访问时丢失任何配置数据。

注意： 在对工程进行开发的过程中应当采取定期备份策略。执行备份之前，请确认刷新工程中所有链接的标签。

Vijeo Citect 允许您将工程备份到本地驱动器（硬盘驱动器）、网络位置或可移动介质（软盘驱动器、内存条）。

帮助的本部分包括有关以下归档任务的信息：

- [备份工程](#)
- [备份 INI 文件](#)
- [使用密码加密配置备份](#)
- [从命令行运行备份](#)
- [恢复工程](#)


备份工程

Vijeo Citect 备份程序采用标准压缩技术归档文件，产生的是同 PKZip® v2.04g 相兼容的压缩文件。Vijeo Citect 备份文件的默认扩展名为.CTZ，但也可以使用其他扩展名（包括.ZIP）。这意味着用户还可以使用 PKZip 工具从压缩的 Vijeo Citect 备份中提取文件。

注意： 使用此备份程序产生的文件无法用 CitectSCADA 5.10 之前的版本来恢复。

备份工程：

1. 打开 Citect 资源管理器。

2. 单击**备份**按钮， 或选择**工具 | 备份**。随即会显示“备份工程”对话框：
3. 在**名称**字段中，选择要备份的工程名称。
4. 在**备份文件**字段中，输入到备份文件位置的路径，包括文件名。既可以直接键入路径，也可以使用**浏览**按钮。
备份文件名称缺省为<工程>.CTZ。如果省略了后缀，将采用.CTZ 作为后缀。
将工程备份到软盘时，备份程序会询问开始备份之前是否要删除该软盘上的文件。
如果目标驱动器配置为 A: 或 B: 并且经检测为可删除，则可以选择删除该磁盘上的任何现有文件。
5. 在**选项**之下，从以下列表中选择所需选项：
 - **使用压缩**：备份工程时可以使用数据压缩以节省空间。
 - **编译保存**：缺省时，Vijeo Citect 将以**未编译**模式保存工程。如果选择此选项，Vijeo Citect 将同时备份**已编译**和**未编译**工程，这样会使备份文件更大。
 - **保存子目录**：如果选择此选项，Vijeo Citect 将备份工程目录内所有子目录中的全部数据。备份中会保持原来的目录结构，恢复工程时可以选择恢复子目录。例如，如果要备份过程分析器视图，请将它们保存在工程的某个子目录中并选择此选项。恢复工程时，就可以选择同时恢复该过程分析器视图目录。
 - **使用加密**：作为一项追加的安全措施，可以用加密格式备份工程。如果选择此选项，Vijeo Citect 将要求提供密码。Vijeo Citect 向磁盘写入工程时会对受保护的工程采用密码编码。只有输入密码才能恢复该工程。
 - **保存配置文件**：选中此选项可备份 **Config** 文件夹中的全部 *.ini 文件。同时还会备份用于存储时间同步设置（在时间同步实用程序中配置）的 TimeSyncConfig.xml 文件。
6. 单击**确定**。

另请参见
[备份INI文件](#)

备份INI文件

缺省情况下，当选择“保存配置文件”选项时，将备份 "Config" 文件夹中的所有 *.ini 文件。如果使用自定义的 INI 文件（例如，"abc.ini"）而且它放在 "Config" 文件夹中，则它同样会被备份。如果使用自定义的 INI 文件（存储在工程的子目录中），则选择“保存子目录”选项可确保它被备份。

注意： 通过从“快捷”标签上的“工程浏览器属性”对话框将参数一直传递到工程浏览器，可以为 Vijeo Citect 定义非缺省 INI 文件。有关如何进行定义的进一步信息，请参见 [使用备用 INI 文件](#)。

如果通过命令行运行备份程序，并将 INI 文件指定为参数，则会备份指定的 INI 文件而不备份 citect.ini。

另请参见
[配置密码加密备份](#)

配置具有密码加密的备份

当您选择了“使用加密”备份选项时，Vijeo Citect 会以对密码编码的格式将工程写入磁盘。只有在输入密码才能恢复工程。

要使用加密，请执行下列操作：

1. 选中备份工程对话框上的“使用加密”选项。
2. 点击**确定**。将显示备份/恢复加密对话框：
3. 在**输入密码**域中，输入您的密码。在输入字符的位置将会显示星号。
4. 在**重新输入密码**域中，重新输入您的密码。Vijeo Citect 会检查您两次是否输入了相同的密码。
5. 点击**确定**。将会备份工程。

另请参见
[从命令行运行备份](#)

从命令行运行备份

可以从命令行执行 Vijeo Citect 备份程序来备份和恢复 Vijeo Citect 工程以外的文件。

从版本 5 开始，备份程序的名称为 CtBack32.exe。对于早期版本，其名称为 CtBackup.exe。缺省情况下，该文件安装在 Vijeo Citect 工程的“Bin”文件夹中。

Vijeo Citect 备份程序使用标准压缩技术归档文件，产生的与 PKZip® v2.04g 兼容的压缩文件。尽管可以使用任何扩展名（包括 .ZIP），但 Vijeo Citect 备份文件的缺省扩展名为 .CTZ。这表示用户也可以使用 PKZip 从压缩的 Vijeo Citect 备份中提取文件。

从命令行执行备份时，如果将 INI 文件指定为参数，则会将该文件备份，而不是缺省的 citect.ini 文件。

备份程序会使用 [BACKUP] 分类读取任何参数的 citect.ini 文件。这些设置（如果进行了设置，如果没有进行设置则使用缺省值）将被作为命令行选项传递的任何参数值覆盖。

下表介绍备份命令行选项。

选项	描述
-d<name>	数据库名称
-m<ext>	包括扩展名
-x<ext>	排除扩展名
-e	使用密码进行加密

选项	描述
-p<password>	对密码进行加密/取消加密
-s[+/-]	递归子目录
-f<level>	格式化级别, 0 表示仅在需要时格式化, 2 表示始终格式化磁盘[自版本 3.xx、4.xx 后不再使用]
-u[+/-]	保存未编译的数据, 使用 -u- 保存已编译的数据
-g[+/-]	显示配置对话框
-c[+/-]	压缩文件
-b<path>	备份路径
-r<path>	恢复路径
-i<filename>	ini 文件名
-f1	使用旧的文件格式 (将长文件名截断为 8.3)
-a	在自动模式下运行 (注意: 命令行或 INI 文件必须包含所有所需的输入。)

范例

- 要备份 (在版本 3 中) c:\data, 请使用以下命令:

```
CTBACKUP -g- -bc:\data
```

- 要恢复上述数据, 请使用 (在版本 5 中):

```
CTBACK32 -g -rc:\data
```

- 要备份 Vijeo Citect 数据库 (例如备份演示), 请使用:

```
CTBACK32 -dDEMO -b -u- -c+ -d-
```

Ctbackup 也使用 CITECT.INI 文件中的以下参数:

```
[BACKUP]
Database= ! database to backup or restore
BackupPath= ! file to backup to, for example c:\temp\example.ctz.

DrivePath= ! path to backup to or restore from.

FilePath= ! file path, used in not a database
BackupFile= ! file name on backup disk, default CTBACKUP.

Password= ! encryption password
Drive=0/1/2 ! 0=other, 1=A, 2=B
DiskSize=0/1 ! low density=0, high density=1
Encrypt=0/1 ! encrypt backup
FormatLevel= ! format level.

Configure=0/1 ! display configure dialog
Compress=0/1 ! compress backup
Overwrite=0/1 ! overwrite
SaveCompiled=0/1 ! save compiled
Recurse=0/1 ! recurse sub directories
DeleteAll=0/1 ! delete all before restore
SaveIniFiles=0/1! determines whether save ini files is checked
Operation=0/1 ! 0=backup, 1=restore
Include= ! include list
Exclude= ! exclude list, default DBK,_CI
CompiledFiles= ! compiled files, default RDB
FileFormat=0/1 ! 1= use old format (truncates long filenames to 8.3)
```

恢复工程

可以使用“恢复工程”程序恢复已备份和归档的工程。用户可以通过此程序将任何当前工程覆盖为备份版本，或者将已备份的工程恢复为新工程。

注意：将文件恢复为目标中的所有文件时请小心，否则子目录会在恢复之前删除。如果不小心将恢复路径设置为驱动器根目录，则程序将删除整个磁盘驱动器。

小心

硬盘驱动器删除

请勿将恢复工程路径设置到驱动器的根目录（通常是 c:\）。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

如何恢复工程：

1. 打开 Citect 浏览器。
2. 单击**恢复**按钮。



或选择**工具 | 恢复**。随即会显示“恢复工程”对话框。

3. 在**备份文件**字段中，输入要恢复的工程名称。
4. 在**恢复位置**下，选择“当前工程”将工程覆盖为已备份的工程，或选择“新工程”将备份的工程恢复为新工程。
5. 在**名称**字段中，输入恢复的工程名称。
6. 在**位置**字段中，输入要恢复的工程位置，包括文件名。可以直接键入路径或使用**浏览**按钮。
7. 在**选项**下，选择“配置文件”恢复备份的 INI 文件和 TimeSyncConfig.xml 文件，TimeSyncConfig.xml 文件用于存储在“时间同步”工具中配置的时间同步设置。
8. 如果备份了工程下的子目录，则这些目录会在“选择要恢复的子目录”下列出。可以选择恢复所有子目录或不恢复子目录，也可以选择要恢复的特定子目录。
9. 单击**确定**。

另请参见
[归档工程](#)

包含工程

在大型系统中，使用一组较小的工程而不是一个大型工程可以使得系统应用的开发更为方便。例如，用户可以为工厂中的每个部分或者主要生产流程都建立一个独立的工程。这样，用户就可以在将这些工程合并到主工程之前分别对每个较小的工程进行开发和测试。

Vijeo Citect 工程将不会被合并到任何其他工程中去，除非已经在 Citect 工程编辑器中专门对合并进行了指定。

注意：如果某个工程远程存在于与本地安装相同的网络上，并且在共享或网络驱动器上，那么它可以连接到本地 Citect 浏览器。这种连接与合并某个工程是不同的。连接使得在本地的 Citect 浏览器中可以查看某个工程。一旦建立了连接，用户可以通过网络将该工程选中作为当前工程进行编辑。

任何建立了连接的工程（在 Citect 浏览器中可以看见）都可以同某个本地的 Citect 工程进行合并，并且相应地被合并到该本地工程的编译之中。

请注意不要将包含文件与合并工程这两个概念相混淆：

- **包含文件**包含 Vijeo Citect 命令和/或表达式，并且在 Vijeo Citect 命令或表达式属性字段中被用作替代。
- **包含的工程**是独立的，并且（通常是更小的）工程，可以合并到另一个 Vijeo Citect 工程中，因此它们显示为一个工程。

每个 Vijeo Citect 系统都提供了许多包含工程。这些工程都包含预定义的数据库记录。

在当前工程中包含另一个工程

包含另一个工程（到当前工程中）：

1. 打开 Citect 浏览器。
2. 从列表中选择工程。
3. 选择**系统**图标，然后选择**包含工程**。
4. 完成显示的“包含工程”对话框。
5. 单击**添加**附加您创建的记录；如果修改了某个记录，请单击**替代**。

注意：不要定义循环参考。也就是说，如果工程 A 包含工程 B，就不要在工程 B 中包含工程 A。否则会在编译时出现“无法打开文件”错误。相反，创建另一个工程并将 A 和 B 都包括进去。

另请参见

[包含工程对话框](#)

[Vijeo Citect的包含工程](#)

包含工程对话框

该对话框使用户可以在当前工程中 [包含另一个工程](#)。在大型系统中，用户应该使用一系列更小的工程而不是使用一个大工程来开发应用程序。

用户最多可以包含 240 个工程。（为了达到此限制，用户必须将 [CtEdit]DBFiles 参数设置为 310。）每个工程中所有的记录都将变为全局性的（也就是说，在其中任何一个工程中定义的记录均可以被另一个工程使用）。

注意：每个系统都自动有一个包含工程，它包含预定义的数据库记录和图形库。

工程名称

要包括在当前工程中的工程的名称（最多 64 个字符）。

注释

任何有用的注释（最多为 48 个字符）。

包含工程

每个 Vijeo Citect 安装都提供了三个预定义的包含工程，旨在帮助用户更迅速地开发自己的工程。它们是：

- **包含工程** - 带趋势和报警画面的模板工程。
- **CSV_Include 工程** - Windows XP 风格的模板集，带有常用工具栏和高级可视化工具。
- **CSV_Instant 趋势工程** - 创建用于支持 CSV_Include 工程的即时趋势功能。

这些工程包含预定义的数据库记录和图形库，可以将其作为自己工程的内容基础。

注意：请不要修改包含工程以作为运行期间工程使用。它将无法成功编译，应当将它保留，以作为新工程的模板使用。Vijeo Citect 升级会安装 CSV_Include 工程的新版本，因此升级时将丢失对工程所做的更改。

缺省情况下，包含工程在 Citect 资源管理器的工程树中处于隐藏状态。

显示/隐藏 Vijeo Citect 包含工程：

1. 打开 Citect 资源管理器。
2. 从视图菜单中选择显示包含工程。

另请参见

[CSV Include 简介](#)

使用工程编辑器

工程编辑器是用来配置工程的变量地址、通信和系统组件的主要工具。本节着眼于合并到工程编辑器的组件以支持此过程。

- [设置工程编辑器选项](#)
- [粘贴标签对话框](#)
- [粘贴函数对话框](#)
- [查找用户函数对话框](#)
- [在工程中使用查找和替换功能](#)

设置工程编辑器选项

工程编辑器提供了改变工程配置环境运行方式的选项。

如何设置工程编辑器选项：

1. 启动工程编辑器
2. 从工具菜单中选择选项。
3. 进行所需的选项调整。
4. 单击确定。

另请参见

[工程编辑器选项对话框](#)

工程编辑器选项对话框

可通过此对话框调整工程编辑器的功能。

显示删除

使数据库中的已删除记录显示。启用后，数据库表底部的一个复选框指示某记录是否已被删除。

增加的编译

启用工程的增加编译。

扩展的表格

使扩展的数据库表显示。还可以使用键盘上的 F2 键显示扩展的表。

通知记录的改变

在您向数据库中添加数据（或更改其中数据），然后试图关闭表格（在添加或替代记录前）时，使“记录已被更改”消息窗口显示。

注意：如果您在禁止此选项后更改了某数据库记录，但忘记了添加或替换该记录，您将丢失数据。

废除用户函数搜索

当使用组合框选择函数（针对命令或表达式域）时，将显示一个由内置 Cicode 函数和用户写入的函数组成的列表。如果您禁止用户函数，此列表中将只显示内置函数。

确认工程打包

在打包数据库时，使“打包数据库可能需要较长时间”信息窗口显示。

自动打开错误表格

如果在编译工程时检测到错误，则会自动显示编译错误表格。

编译询问信息

在修改工程并从文件菜单中选择运行后，使“要编译吗？”消息框显示。通常，Vijeo Citect 会在运行被选中后自动编译工程（如果此工程已被修改）。

编译成功信息

当编译完工程时，使“编译成功”消息窗口显示。

标签不存在提示

当指定变量标签不在数据库中时，使“未找到变量标签，是否创建此标签？”消息窗口显示。在消息窗口被启用后，您就可以根据需要创建新变量标签了。

准备 Web 部署

每次编译工程时自动运行 Web 部署准备工具。请注意，这将使每次编译所花费的时间显著增加，尤其是对于大型工程。

记录编译期间的不建议使用警告

如果选择此选项，编辑器将生成一个提醒信息，用以识别在工程中检测到的不建议使用的元素，即不再支持的函数、参数或内核命令。

通过取消选中此选项，提醒信息仍将包括在显示的提醒计数中，但不会被添加到错误日志中。

信息弹出时间

从开始搜索数据库到搜索信息窗口显示之间的延迟（以秒计）。搜索信息窗口显示被跟踪的记录的数量，并允许您取消搜索。您可以通过选择信息窗口中的取消按钮来取消搜索。

Cicode 编辑器

用于编辑 Cicode 函数库和报表格式文件的文本编辑器。您必须在此域中输入可执行文件的名称。缺省编辑器为 Vijeo Citect 随附的 Cicode 编辑器 (ctcicode.exe)。

报表编辑器

用于编辑报表格式文件的编辑器。您必须在此域中输入可执行文件的名称。缺省编辑器为 Write (write.exe)。如果您使用的是富文本格式 (RTF) 报表，确认您的编辑器兼容 RTF。

打印页面大小

打印数据库记录时每页上打印的行数（1 至 66）。

打印字号

打印数据库记录时使用的字号。

打印字体名

打印数据库记录的字体名。

最大列表框项目

下拉组合框中显示的最大记录数量。

就全编译期间未使用的标签发出警告

针对未在工程中直接使用的标签生成提醒条目。当全编译运行时，提醒条目包含在工程编辑器的编译错误表中。默认情况下，此选项不会被选中。

注意： 对此选项，提醒条目仅在全编译时生成，不会在增加的编译时生成。

编译期间记录“标签未定义”警告

如果选择此选项，编译器将针对被检测到未在变量数据库中定义的任何标签在错误日志中生成一个“标签未定义”提醒。

由于 Vijeo Citectv7.10 现在允许在图形画面上加入未定义的标签，此警告可能是多余的非实用警告。通过取消选中此选项，提醒信息仍将包括在显示的警告计数中，但不会被添加到错误日志中。

粘贴标签对话框

如果需要将标签变量插入标签或表达式字段，可以使用“粘贴标签”对话框。

在标签或表达式字段中插入变量标签：

1. 选择要插入标签的位置，比如表单中的表达式字段。
2. 从**编辑**菜单中选择**粘贴标签**以显示“插入标签”对话框。
3. 选定标签名称，然后单击**确定**或单击**取消**。

这样，标签将被插入到光标当前位置的标签或表达式字段中。

粘贴函数对话框

如果需要将函数插入标签或表达式字段，可以使用“插入函数”对话框。

在标签或表达式字段中插入函数：

1. 选择要插入函数的位置，比如表单中的表达式字段。
2. 从**编辑**菜单中选择**粘贴函数**以显示“插入函数”对话框。
3. 选定函数名称，然后单击**确定**或单击**取消**。

如需带参数插入函数，请选中**插入参数**框。

函数将被插入到光标位置的当前字段中。

注意：如果函数及其参数的总长度超过 254 个字符，它将不会显示在此对话框中，而是会显示“文本太大”的消息。

查找用户函数对话框

如果您需要在 Cicode 源文件中查找 Cicode 函数，可以使用“查找用户函数”对话框。

要查找 Cicode 源文件中的函数，请执行下列操作：

1. 从**编辑**菜单选择**查找用户函数**。
2. 输入函数名称（或部分函数名称），然后点击**确定**或点击**取消**。

将会显示一系列匹配您搜索条件的函数。

注意：如果保留“查找”字段为空然后点击“确定”，函数的完整列表将显示在该列中。

要编辑包含函数的 Cicode 文件，请执行下列操作：

1. 当搜索函数时（请见上述内容）从出现的列表中选择函数名称，然后点击**编辑**或点击**取消**。

包含选定函数的文件将会在 Cicode 编辑器中打开。

在工程中使用查找和替换功能

可以使用“查找和替换”对话框在工程中查找指定文本。可以在工程中执行全文替换，并到处搜索结果。

以下主题介绍了这些内容：

- [“查找和替换”对话框](#)
- [指定搜索范围](#)
- [使用结果列表](#)
- [删除结果](#)
- [导出结果](#)
- [跳转到结果（转到）](#)
- [替换结果](#)
- [查找与替换提醒信息](#)

还有一个 [搜索疑难解答](#) 的主题，有助于确定是否在返回意外结果时正确配置了搜索。

“查找和替换”对话框

可以采取以下方式打开“查找和替换”对话框：

- **工程编辑器**：可以查找和替换工程和包含工程中的文本字符串。
- **图形编辑器**：可以查找和替换图形页面、模板或精灵中的文本。

可以配置搜索范围、查看结果、替换结果，或打开搜索结果以了解更多信息。

要显示“查找和替换”对话框：

- 在工程编辑器或图形编辑器中，点击 **编辑 | 查找或编辑 | 替换**。选择 **查找** 分页或 **替换** 分页后显示的对话框，具体取决于所选命令。

要搜索文本：

1. 在工程编辑器或图形编辑器的 **编辑** 菜单中，点击 **查找**。
2. 在 **查找** 框中，键入要搜索的文本字符串。搜索不区分大小写，因此输入大写或小写字母都可以。
可以输入要查找的完整字符串或该字符串的一部分。例如，键入 BIT 将返回所有包含 BIT 的字符串，如 BIT_1、BITE、HABIT、HABITS 等。不能输入通配符，但可以根据需要加入特殊字符和空格。
3. 使用 **查找** 和 **搜索选项** 列表指定 [搜索范围](#)。
4. 点击 **查找**。搜索完成后，搜索结果将显示在 [结果列表](#) 中。结果列表下的状态文本指示搜索进度。

注意：启动搜索后，**查找** 按钮变为 **停止** 按钮，您可以使用该按钮退出搜索。如果停止搜索，由部分结果组成的列表将显示。

要替换文本：

1. 在工程编辑器或图形编辑器的“编辑”菜单中，点击 **替换**。
2. 在 **查找** 框中，键入要搜索的文本。
3. 在 **替换为** 框中，输入替换文本。
4. 指定 [搜索范围](#)。
5. 点击 **查找**。
6. 查看 [搜索结果](#)。

7. 用**替换**或**全部替换**项 [进行替换](#)。

指定搜索范围

可以使用**查找**和**搜索选项**列表指定搜索范围，从而确定要搜索的工程中的项目。

当选择**查找**选项时，**搜索选项**将发生改变。每次选择**查找**选项时，所有关联的**搜索条件**都会被默认选中。**查找**选项在以下几种情况下能与**搜索选项**一起使用。

- 选择**当前工程**或**当前工程和包含工程**将显示下述选项。
 - **常规**：搜索与工程以及包含工程关联的所有配置数据库（如果该选项被选中）。
 - **图形**：仅对图形页面执行“快速搜索”（图形页面当前并非必须处于打开状态）。*如果符号、精灵或模板中的文本未被用在页面上，此搜索不会查找这些文本。*如果未找到所需文本，尝试更全面的图形搜索，方法是打开页面、精灵、图形或模板并选择**当前图形**作为**查找**选项（见下文）。
 - **代码文件**：搜索当前工程和包含工程（如果该选项被选中）中的所有 Cicode/CitectVBA 文件。
 - **报表**：搜索工程文件夹和包含工程（如果该选项被选中）中的所有报表文件。
- 选择**当前图形**将使下述选项可用。（相对于在上述**当前工程**：**图形**部分执行的搜索，此搜索更为全面，并会搜索当前处于打开状态的图形文本。）
 - **精灵和符号内部**包括精灵或符号内包含的图形对象。
 - **模板内部**包括页面模板内包含的对象。
- 选择**当前表**：搜索当前表。

使用结果列表

随着匹配项被找到，它们将被列在结果列表中。结果列表将显示匹配在搜索中输入的字符串的项目的概述。

结果列表每个画面最多可以显示 200 个结果，这些结果按照工程和项目排序，您不能更改排序的顺序。结果列表包含以下列：

列	描述
工程	出现找到的文本的工程的名称。
项目	取决于项目出现的文档类型。如果文档类型是： 数据库 - 数据库的用户友好名称。 画面 - 画面的名称。 Cicode/VBA - Cicode/VBA 文件的名称和路径。 报表 - 报表的名称。
域	标识找到的项目出现的文档/数据库的那个部分。例如，如果找到的项目出现在数据库中，这将参考数据库中的列名称。请注意，搜索将覆盖表达式/命令和数值属性。

列	描述
保存位置	显示找到的项目在文档/数据库中出现的特定记录号、AN 或行号。
上下文	找到的项目在工程中出现的上下文的示例。例如，如果文档类型是： 数据库 - BIT* 的搜索结果的上下文可能是 BIT_!。 画面 - BIT* 的上下文可能是 Toggle(BIT_!)。 Cicode/VBA - UserName 的上下文可能是 FUNCTION GetUserName()。 报表 - PUMP* 的上下文可能是 @(Pump A)。

如果返回的结果数超过 200 个项目，请使用**第一个**、**上一个**、**下一个**和**最后一个**按钮以 200 个结果为一组导航您的结果。

您可以切换查找和替换功能而不丢失搜索结果，但如果您关闭“结果”画面，搜索结果将丢失。

注意：您可以重新调整列表列的大小，方法是在列表列之间的分隔符上移动鼠标光标。当鼠标光标变成带箭头的黑条的形状时，拖动列到新的大小。您还可以双击字段之间的垂直条以便重新调整该字段的大小，使其适应最宽的项目。

删除结果

您可以删除“结果”窗口中的搜索结果。删除的结果不包含在导出操作或替换操作中。删除结果并不会真的将其删除，而只是将其从“结果”窗口中删除。

如何删除结果：

- 高亮显示您要删除的结果，然后点击**删除**。该结果将从“结果”窗口中删除。

导出结果

您可以将搜索结果以制表符分隔格式导出到指定的位置。结果以如下格式导出

```
<Project> <Item> <Field> <Location> <Context>
```

如果“结果”窗口包含超过 200 条结果，所有结果都将被导出，而不只是当前显示的结果。如果从结果列表中删除一个项目，该项目将不被导出。（有关删除结果的详细信息，请参见 [删除结果](#)。）

如果导出的项目具有上下文，上下文字符串中将删除制表符和换行字符。

结果以 Unicode 格式导出。由于此原因，文件中将被添加两个开始字符和两个拖尾字符，但在大多数情况下它们将保持隐藏状态。当导出结果时，请使用支持 Unicode 格式的 Excel 2000 或更高版本。

如何导出结果:

1. 当希望导出的搜索结果列在“结果”窗口中时，点击**导出**。
2. 在对话框中指定位置，然后点击**保存**。如果此文件已经存在，您可以选择是否覆盖此文件。结果列表下的状态文本将指示导出进度。

注意: 如果要停止导出，请点击**停止**。您不能执行部分导出，因此点击**停止**将取消整个导出。

跳转到结果（转到）

可以跳转到单个结果以查看结果出现的位置。根据包含查找结果的文档的类型，结果可能出现在以下位置：

- **数据库:** 表单在工程编辑器中打开，文字字符串被高亮显示。
- **Cicode/VBA:** 文档在 Cicode 编辑器中打开，文字字符串被高亮显示。
- **图形:** 画面在图形编辑器中打开，显示包含文字字符串的对象的“属性”对话框。显示包含文字字符串的属性。如果该字符串出现在精灵上或精灵中，则还会显示该精灵的表单。
- **报表:** 所配置的报表编辑器打开并显示报表文件，但不高亮显示文字字符串。

跳转到结果:

- 在“结果”窗口中高亮显示要跳转到的查找结果后，点击**转到**。包含出现内容的文档或表单随即打开。

另请参见
[替换结果](#)

替换结果

您可以使用指定的替换文本字符串替换单个结果或多个结果。您也可以在替换单个结果之前，测试此结果。根据包含查找结果的文档的类型，替换时可能出现以下情况：

- **数据库:** 使用替换文本替换结果，并更新数据库记录。包含搜索结果的表单未打开；要在替换前或替换后查看搜索结果的位置，请使用 [Go To 命令](#)。
- **Cicode/VBA:** 将加载包含匹配文本的 Cicode 文件（如果未加载），进行替换，并保存文件。
- **图形:** 在图形编辑器中打开画面（如果画面还不可用），然后进行替换。如果打开的画面包含未保存的更改，则将会提示您保存或放弃更改，然后进行替换。如果对同一个图形画面做了多个更改，则画面将保持打开，直到完成所有更改。
- **报表:** 使用替换文本替换找到的文本，并保存该文件。

注意: 执行替换后，则无法取消替换。在进行替换前，应小心检查这些替换，尤其是使用了多个替换。

如何删除结果:

1. 高亮显示您要测试的结果，然后单击**测试**。将显示一个对话框，表示文本替换的结果。
2. 单击**接受**，接受文本替换，或单击**退出**。

替换单个结果：

- 高亮显示您要替换的结果，然后单击**替换**。进行替换，然后从“结果”窗口删除结果。然后选择列表中的下个结果。

如果替换多个结果：

1. 当希望导出的搜索结果列在“结果”窗口中时，单击**全部替换**。随即显示确认对话框。
2. 进行替换，然后从“结果”窗口删除替换。（未进行的替换将保留在结果列表中。例如，您尝试替换只读属性时将发生这种情况。）

注意： 在此过程中，单击**停止**不会取消已经做出的任何替换。

尝试进行替换时，您可能会遇到提醒消息，提示您应注意项目相关问题，然后再进行替换。有关详细信息，请参见 [查找和替换提醒消息](#)。

查找与替换提醒信息

查找与替换如果无法替换文本字符串，将显示下列其中一个错误：

- [文件在使用](#)
- [已替换文本被截断](#)
- [未找到原始文本](#)
- [已替换文本超出范围](#)
- [替换文本非数值](#)
- [域只读](#)
- [未知错误](#)

文件在使用

如果需要写入的数据库或文件已经无法使用，则显示此提醒信息。如果数据库/文件在由第三方应用程序使用，也会显示此错误。

需要执行下列其中一种操作：

- 单击**重试**（默认），对数据库/文件重复执行操作。
- 单击**忽略**，跳过对此文件的操作。
- 单击**全部忽略**按钮跳过对在使用文件的所有操作；如果选择此选项，将不显示该信息。

已替换文本被截断

如果在 DBF 字段中执行文本替换时超出字段宽度限制，则出现此提醒信息。例如，如果含有 % 的工程单位字段被替换为“%LONGTEXTSTRING”，则会出现“替换文本被截断”提醒信息，因为此字段的最大宽度为 8。替换文本因此变成“%LONGTEX”。

如果搜索当前图形画面，则不会发生此提醒信息。

请执行下列操作之一：

- 点击**是**，将截断文本提交给数据库。通常，编译工程时，会生成编译提醒信息。
- 点击**全是**，提交对字段进行的所有更改，无论截断是否在未再次显示提醒的情况下存在。
- 点击**否**（默认），将文本保留为原样。
- 点击**否**，**全部**按钮，将所有截断字段保留为原样。

未找到原始文本

如果试图在无法找到动画或文本时替换当前图形画面上的项目，将出现这一提醒信息。如果已删除了动画，或者在找到动画但还未更换项目前对在动画域中找到的文本进行了更改，将会出现这种情况。

在下面的范例中，填充高度最大值包含值 23，搜索框是 23。在更换记录前，它变为了 66。

请执行下列操作之一：

- 点击**忽略**跳过此操作，保留列表中的条目，然后转至下一个替换（如果有的话）。
- 点击**忽略全部**的结果和点击**忽略**按钮类似，只是它将会跳过此替换过程中出现的所有未找到的错误。
- 点击**停止**，停止当前记录中的替换过程。

已替换文本超出范围

此提醒信息显示在当前图形画面上执行替换的两种不同情况：

1. 字段为文本字段，且长于所允许的字段宽度。
2. 字段为数字字段，且超出替换参数值所允许的范围。

在下面的示例中，“最大填充级别”允许的范围为 0-100，参数值为 23，由 101 所替换，超出了范围。

请执行下列操作之一：

- 点击**忽略**跳过此操作，保留列表中的条目，然后转至下一个替换（如果有的话）。
- 点击**全部忽略**的结果和点击**忽略**按钮类似，只是它将会跳过此替换过程中出现的超出范围的错误。
- 点击**停止**，停止当前记录中的替换过程。

替换文本非数值

如果在当前图形画面中执行替换，当字段被替换为数字字段且替换文本中包含非数字参数值时，将显示此提醒信息。

在下面的示例中，“最大填充级别”包含参数值“23”，替换文本为“fred”。

请执行下列操作之一：

- 点击**忽略**跳过此操作，保留列表中的条目，然后转至下一个替换（如果有的话）。

- 点击**全部忽略**的结果和点击**忽略**按钮类似，只是它将会跳过此替换过程中出现的所有非数值错误。
- 点击**停止**，停止当前记录中的替换过程。

域只读

当在当前图形画面上替换某项时，如果要替换的域是精灵或模板等对象的一部分，则会显示此提醒信息。

在下面的范例中，表达式域是属于精灵的一个对象的一部分。

需要执行下列其中一种操作：

- 点击**忽略**跳过此操作，保留列表中的条目，然后转至下一个替换（如果有的话）。
- 点击**全部忽略**的结果和点击**忽略**按钮差不多，只是它将会跳过此替换过程中出现的所有只读错误。
- 点击**停止**，停止当前记录中的替换过程。

未知错误

在当前图形画面上执行替换后检测到一般错误时，将显示此提醒信息，在正常操作中不应发生。

请执行下列操作之一：

- 点击**忽略**跳过此操作，保留列表中的条目，然后转至下一个替换（如果有的话）。
- 点击**全部忽略**的结果和点击**忽略**按钮类似，只是它将会跳过此替换过程中出现的所有未知错误。
- 点击**停止**，停止当前记录中的替换过程。

故障排除搜索

如果您没有找到希望得到的结果，可检查以下各点，然后再次执行搜索：

- 文本字符串是否拼写正确？
- 空格数是否正确？
- 使用的“查找”选项是否适当？
- 使用的“搜索”选项是否适当？
- 搜索的工程是否正确？
- 使用的图形搜索是否正确？
- 如果您使用的是图形画面搜索，打开的图形画面是否正确？

Chapter: 10 保护工程

Vijeo Citect 工程表示相当可观的投资。传递委托工程后，该工程必须保持传递状态，直到修改被获得授权的人员执行。为了防止未授权人员对工程进行修改，可通过 Vijeo Citect 将管理员的权限设置为“只读”来保护工程。

对于大型应用程序或其中必须限制对某些进程或机器的访问权限的应用程序，可将安全性内置于系统。随后可限制对所有操作员应不可用的命令的访问，例如，操作专用机器、确认重要报警或打印机密报告的命令。可以使用 2 个选项配置系统安全性。

这些选项使用 [Vijeo Citect 本机安全性](#) 或 [Vijeo Citect 与 Windows 安全性集成](#)。

本部分说明下列内容：

- 读写工程和只读工程的特点（请参见 [概览](#)）。
- 说明 [保护顶级工程](#) 和 [保护包含工程](#) 的方案。
- 如何保护工程，使其成为只读（请参见 [使工程只读](#)）。
- 保护工程的结果（请参见 [有关工程的只读权限](#)）。
- 使用 [Vijeo Citect 本机安全性](#)。
- 使用 [Vijeo Citect 与 Windows 安全性集成](#)。

概述

Vijeo Citect 有两种类型的工程：

- **读写：**允许当前用户对工程文件夹（或任何工程文件）拥有写入和删除权限。
- **只读：**拒绝当前用户对工程文件夹拥有写入和删除权限的工程。

下表显示只读和读写工程的不同特性：

任务	只读	读写
查看工程中的现有文件	x	x
在工程中创建新文件		x
删除工程中的现有文件		x
修改工程中的现有文件		x
删除工程		x
直接为工程重命名		x

只读工程不能被作为顶层工程编译（即主（根）工程与包含工程相对立），并且不支持在线更改。

注意：如果当前用户对工程文件夹拥有只读权限，而他对其中的一个或多个文件却拥有读写访问权限，该工程将被认为是只读/读写混合工程。Vijeo Citect 不支持此类工程。运行混合工程将导致系统变得无法作出响应。（此注意不包括需要拥有读写访问权以便在运行时操作的文件夹或文件；有关详细信息，请参见 [将 Vijeo Citect 与 Windows Security](#) 一起使用。

启用只读工程时使用的安全模型不会替换现有 Vijeo Citect 用户帐户；相反，它会通过以下途径与这些用户帐户联合工作：

- Vijeo Citect 用户帐户控制工程元素的 Runtime 安全。
- Windows 用户帐户控制配置工程元素的安全。

保护顶级工程

本部分介绍“真实世界”的情况，可能需要对顶级工程应用只读权限。

注意：保护顶级工程之前，请阅读 [有关工程的只读权限](#) 了解有关操作约束的详细信息。尤其要注意 [有关顶级工程的只读权限](#) 部分。

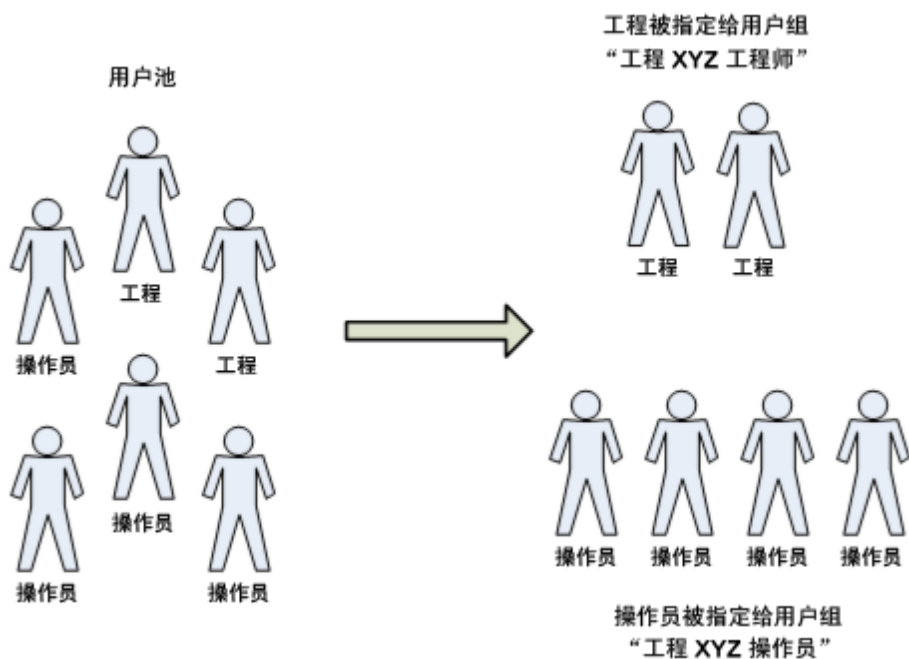
在此方案中，多个实地工程师负责维护顶级工程 ProjectXYZ。因此他们需要对所有工程文件夹的读写权限。

负责监视工厂操作的操作员只在运行时使用工程，因此对工程只具有只读访问权限。

实地系统管理员首先识别所有使用工程的员工，然后将用户池划分为两个用户组：

- **工程工程师** - 负责工程配置。
- **操作员** - 负责工程的运行时操作。

下图对此进行了说明。



为了更容易地管理用户，管理员创建了两个用户组：**ProjectXYZEngineers** 和 **ProjectXYZOperators**，然后将工程师分配给第一组，将操作员分配给第二组。

注意：创建用户组是可选的，但此操作可使多个用户权限的处理更加容易。如果只有几个用户，则不必创建用户组。

随后，管理员会向工程师分配对顶级工程文件夹的读写权限，向操作员分配只读权限，诸如此类。

1. 选择顶级工程的工程文件夹并显示其属性。
2. 选择 **ProjectXYZEngineers** 用户组并允许使用读写权限。（请记住，为了使用读写工程，必须分配读取、写入和删除权限。）
3. 选择 **ProjectXYZOperators** 用户组并拒绝使用写入权限。有关应分配的特定权限，请参见 [使工程只读](#) 部分。
4. 应用和保存更改。
5. 检查更改以确认工程师和操作员对其角色具有正确的权限。

另请参见
[保护包含工程](#)

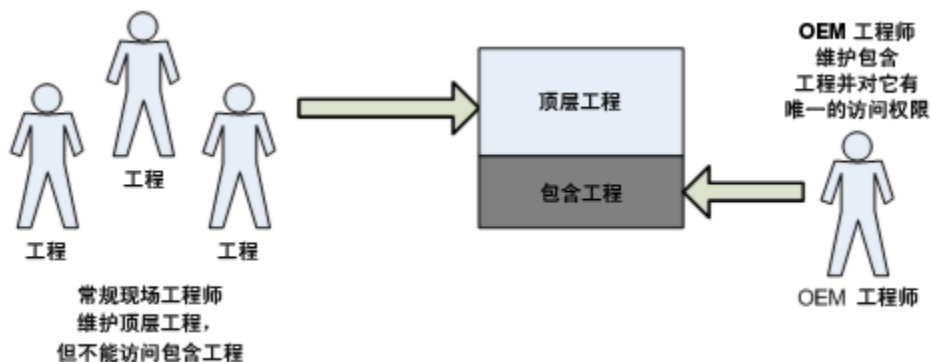
保护包含工程

本部分介绍“真实世界”的情况，可能需要对包含工程应用只读权限。

注意：保护包含工程之前，请阅读 [有关工程的只读权限](#) 了解有关操作约束的详细信息。尤其要注意 [有关包含工程的只读权限](#) 部分。

在此情况中，OEM 已配置和传递了包含工程，该工程是大型（顶级）工程的一部分。由于 OEM 工程师只负责维护包含（并且仅包含）工程，因此实地管理员会为 OEM 工程师分配对包含工程的读写访问权限，但只为其分配对顶级工程的只读访问权限。相反，实地普通工程师可以访问顶级工程，但不能访问包含工程。

此处显示此方案：



要设置此方案，管理员可执行以下操作：

1. 为了便于管理，将实地工程师分配给用户组 **AcmeTopEngineers**。

注意：由于只有一位 OEM 工程师，因此管理员不会为这一个用户创建用户组。

2. 选择顶级工程的工程文件夹并显示其属性。
3. 选择 **AcmeTopEngineers** 用户组并允许对此文件夹使用读写权限。（请记住，为了使用读写工程，必须分配读取、写入和删除权限。）
4. 应用和保存更改。
5. 选择包含工程。
6. 选择 OEM 工程师的用户名并允许对此文件夹使用读写权限。
7. 应用和保存更改。
8. 检查所做的更改以确认分配了正确的权限。尤其要注意的是，管理员应确保分配给 **AcmeTopEngineers** 用户组的权限会拒绝对包含工程的读写访问。

另请参见
[保护顶级工程](#)

将工程变为只读

本节介绍如何通过对选定用户和/或用户组修改文件安全设置来使文件夹变为只读。无论使用的是 Windows 2003 Server、Windows XP 或 Windows 2000，只读配置都是相同的。

注意

- 此过程假定您已经配置了用户和用户组（可选），并将他们添加到要保护工程的工程文件夹列表的用户组中。有关创建用户和用户组的详细信息，请参阅您的 Windows 文档。
- 将工程变为只读之前，确认已恰当地应用了读/写、删除及执行权限，以使 Vijeo Citect 配置和运行时环境能够正常运行。有关详细信息，请参见 [利用 Windows 安全性使用 Vijeo Citect](#)。
- 如果使用的是 Windows XP 或 Windows 2000，而且文件系统是 FAT32，则必须将文件系统转换为 NTFS，才能在工作站上指定正确的用户权限。有关详细信息，请参阅 Windows 文档。
- 为了避免出现不希望的结果，请在将工程变成只读之前阅读 [工程的只读权限](#)。

将工程变成只读：

1. 在 Windows 资源管理器中，选择要使其变成只读的工程文件夹。缺省情况下，工程文件夹位于下面的文件夹中

C:\ProgramData\Schneider Electric\Vijeo Citect 7.10

2. 右键单击该文件夹并从上下文菜单中选择**属性**。随即会显示“属性”对话框。
3. 选择**安全**分页。
4. 选择要修改其安全设置的用户和/或用户组。
5. 点击**高级**。随即会显示工程文件夹的选定用户/用户组的“高级安全设置”对话框。确认选择了要修改其权限设置的用户或用户组。
6. 点击**编辑**显示“权限输入”对话框。
7. 点击**全部清除**清除当前选择，然后选中下列选项的**允许**复选框：
 - **遍历文件夹/执行文件**
 - **列出文件夹/读取数据**
 - **读取属性**
 - **读取扩展属性**
 - **读取权限**
8. 点击**确定**。
9. 点击**应用**将权限应用到选定用户/用户组，然后点击**确定**退出“高级安全设置”对话框。
10. 点击**确定**关闭“属性”对话框。

工程文件夹现在已针对选定用户和/用户组指定为只读。

另请参见

[保护顶级工程](#)

[保护包含工程](#)

工程的只读权限

此部分介绍使工程只读的操作限制。同时，还介绍了保护顶层工程和包含工程的特定问题。

注意：在 [将工程变为只读](#)前，确保您了解此处所介绍的问题。同时，也请确保为配置和操作时运行环境的正常运行设置了正确的权限；有关详细信息，请参见 [使用 Vijeo Citect 与 Windows 安全性](#)。

- [开始](#)
- [概述](#)
- [图形和画面](#)
- [备份和恢复](#)
- [工程升级](#)
- [调试](#)
- [Web 部署](#)
- [运行时问题](#)

上述大多数问题对顶层工程和包含工程都很常见。而下面所列的部分将探讨这些工程类型的一些具体问题：

- [只读顶层工程](#)
- [只读包含工程](#)

启动

在 Citect 工程管理器启动时，工程被确定是否为只读工程。如果在 Citect 浏览器启动后修改了工程文件夹的安全许可，Vijeo Citect 配置应用程序可能无法准确确定工程是否为只读工程。

另请参见

[将 Vijeo Citect 与 Windows 安全一起使用](#)

概述

在使用图形编辑器、工程编辑器和 Citect 浏览器打开只读工程后，标题栏中会显示此工程的名称和一条只读消息，指出该工程为只读工程。

如果为某只读项目打开快速向导，将在第一个画面上显示一条消息，指出该项目为只读项目：

此外，任何菜单命令、工具条按钮和其它执行写函数的操作都将变灰和/或不可用。例如，可以在工程编辑器中对只读工程执行“复制”命令，但不能执行“剪切”和“粘贴”命令。

使用过程分析器时，无法在只读工程文件夹中创建视图，并会显示提醒消息。

图形和画面

您无法更新只读工程中的图形文档或画面。（但如果顶层工程包含一个或多个只读工程，则可以更新读-写工程中的画面。）

在图形编辑器中打开只读工程中的已更新图形画面时，系统将尝试显示已升级符号（仅在演示而非磁盘层）。

注意：不建议在中级包含工程中更新图形文档和/或页面，因为它将造成页面不会在顶级工程中进行更新。但是，可全部通过递归包含工程在顶级发布某个更新页面。（如果这些页面需要更新，请先将所有相关的工程设置为可读写。）

备份和恢复

无论用户的权限如何，均可以备份只读工程。不过，该备份功能并不归档当前安全权限。因此，如果在另一台机器上恢复该工程，则需要重新应用安全设置。

不能将工程恢复到现有只读工程；试图这样做将显示一条提醒消息，告知该工程为只读并且无法恢复。

工程升级

工程在以下任一情况出现时升级：

- [CtEdit]Upgrade=1 将被添加到 citect.ini 文件。
- 工程链接通过 Citect 浏览器添加。
- 当工程被恢复时。

当 Vijeo Citect 检测到包含工程、系统工程或 CSV_Include 工程已就绪并且创建工程时使用的 Vijeo Citect 版本与当前 Vijeo Citect 版本不一致时，将显示向您提出相应建议的消息框。

当 Vijeo Citect 检测到用户工程已就绪并且创建工程时使用的 Vijeo Citect 版本与当前 Vijeo Citect 版本不一致时，将显示向您提出相应建议的消息框。

指向此工程的所有链接都将被删除，Citect 浏览器中的工程树将被更新以指出这一情况。

如果计划升级某顶级工程，您必须以对此工程拥有相应读写安全权限的用户的身份登录，在 Citect 浏览器中向此工程添加一个链接，然后再次执行工程升级。



不希望的设备操作

- 请不要在工程环境中保留未升级的工程。
- 如果安全权限阻止成功升级与您的系统相关的所有工程，在继续操作之前应将系统恢复到其原先的状态。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

调试

在只读工程中，您可以在调试代码时设置断点，但当您退出 Cicode 编辑器时，这些断点不会被保存。

Web 部署

您不能针对只读工程执行 Web 部署。

运行时问题

缺省情况下，运行期间的大多数输入操作都出现在 [DATA] 或 [RUN] 位置（有关详细信息，请参见下文）。缺省情况下，Vijeo Citect 将 [RUN] 位置配置为工程目录。如何打算使工程目录只读，则必须修改路径以适应读写位置。

- **almsav.dat** - 缺省情况下存储在 [RUN] 位置的报警数据，此位置通常是工程文件夹。如果打算使工程文件夹只读，则必须更改此位置。此外，也可以使用 citect.ini 文件中的 [Alarm]SavePrimary 和 [Alarm]SaveSecondary 选项控制输出的位置。
- **磁盘 PLC** - 如果用户对 [RUN] 路径不具有正确的权限，则所有通讯将针对磁盘 PLC 离线。如果打算使工程文件夹只读，则必须更改此位置。
- **用户 Cicode 函数** - 使工程只读可以避免使用以下用户函数：UserCreate、UserDelete、UserEditForm、UserSetPassword、UserSetPasswordForm。尝试使用这些函数会导致错误代码 262 (0x0106)（“无法打开文件”）的出现。
- **报警 Cicode 函数** - 使工程只读可以避免使用以下报警函数：AlarmSetDelay、AlarmSetDelayRec、AlarmSetThreshold、AlarmSetThresholdRec。尝试使用这些函数会导致硬件报警 400 (0x0190)（“工程或文件为只读”）的出现。不能修改报警属性，如阈值或延迟。

顶级工程中需要运行时读写访问权限的任何文件应不在工程文件夹中。

只读顶层工程

在将只读属性应用于顶层工程前，请注意以下内容：

- 在将只读属性应用于顶层工程前，请执行一次完整的编译，然后应用相关的安全性。在应用了只读权限后，您将无法对顶层工程进行编译（请参见以下 bullet）。对于必须在顶层工程中操作文件的工程，您可以将各个文件的安全性修改为可读写；只要工程文件夹拒绝写入权限，该工程就会被视为只读。

注意： 顶层工程中需要运行时读写访问权限的所有文件都应位于工程文件夹外。

- 将只读包含工程作为一个部分的可读写工程通常可以进行编译。
- 如果只读工程是顶层工程，您可以仅运行该工程。
- 将只读属性应用于顶层工程可以防止对警报、用户、趋势以及画面进行在线更改。

另请参见

[将工程变为只读](#)

[保护顶层工程](#)

只读包含工程

在将只读属性应用于包含工程前，请注意以下内容：

- 在编译顶层只读包含工程时，包含工程的对象 ID 将被跳过并更改。
- 不建议在中级包含工程中更新图形文档和/或页面，因为它将造成页面不会在顶级工程中进行更新。但是，可全部通过递归包含工程在顶级发布某个更新页面。（如果这些页面需要更新，请先将所有相关的工程设置为可读写。）

另请参见

[将工程变为只读](#)

[保护包含工程](#)

保护运行时计算机

Vijeo Citect 运行时计算机是基于 Windows 的应用程序，在标准 Windows 环境中运行。通常，Windows 允许您同时运行数个应用程序，不过，如果需要计算机专用于运行期，则可能影响到性能或模糊运行期的消息和信息。

例如，操作员显示用于展示报警通知的面板。这时不希望运行时屏幕最小化或隐藏在其它窗口后面。

限制对软件而不是对 Vijeo Citect 的访问可以使用几个不同的方法。

另请参见

[客户端启动限制](#)

[运行客户机作为命令解释程序](#)

[禁止 Windows 键盘命令](#)

[禁止控制菜单命令](#)

[删除“取消”按钮](#)

客户端启动限制

在启动时，Vijeo Citect 会使用匿名登录建立与服务器的初始通讯。在缺省情况下，客户端与服务器之间的这种通讯使用的是仅查看模式。在这种模式下，用户无法写任何标签、确认任何报警或使用 Cicode 函数。

注意： 仅查看模式被应用到整个显示客户端的进程，包括所有正在运行的 Cicode 任务。

在用户成功登录后，只写访问权限可用。一旦用户退出，则返回到仅查看模式。

用户可以通过修改 [Client]AutoLoginMode 参数来配置匿名登录。

另请参见

[保护运行时计算机](#)

运行客户机作为命令解释程序

为了限制某些操作员在运行时从客户机切换到不同应用程序，可以将 Windows 配置成将 Vijeo Citect 作为命令解释程序在运行时启动。这样会将 Vijeo Citect 运行时部署为计算机唯一可用的接口，限制当前工程环境中的访问。

有关如何设置作为命令解释程序的客户机的信息，请与当地支持办事处联系。

另请参见

[禁止 Windows 键盘命令](#)

禁止 Windows 键盘命令

Windows 环境提供了在计算机上同时运行的应用程序之间进行切换的命令。在使用 Vijeo Citect 时，您可能不希望使用这些命令，因为它们允许操作员访问其他 Windows 设备，并且您无法进行直接控制。可以在 *计算机设置向导* 中禁止其中一部分命令。您可咨询 Citect 资料库，获取有关禁止 Windows 键盘命令的最新信息。

另请参见

[禁止控制菜单命令](#)

禁止控制菜单命令

“控制菜单”（位于应用程序窗口的左上角）提供命令来控制应用程序窗口的位置和大小，在某些应用程序中还控制应用程序。可以设置运行时系统的控制菜单，以访问一些特定于 Vijeo Citect 的命令，例如“关闭”（用于关闭运行时系统）或“内核”（用于显示内核）。

可以在 *计算机设置向导* 中启用或禁止这些命令。

另请参见

[删除“取消”按钮](#)

删除“取消”按钮

当 Vijeo Citect 运行时系统开始时，将会有有一个消息框显示系统的开始状态。这个消息框通常包含一个允许用户取消启动的**取消**按钮。当用户调试或测试系统时，这个按钮是非常有用的。当用户完成测试时，用户可使用 *计算机设置向导* 从此消息框中删除**取消**按钮，这样可防止系统启动被意外取消。

Chapter: 11 使用 Vijeo Citect 安全性

Vijeo Citect 运行时安全系统结合了 8 个权限和 255 区域的组合。权限可以是非等级或等级权限。它非常灵活且功能强大，因为配置工程时，每个对象都可以分配权限和区域。Vijeo Citect 用户也可以分配权限和区域。只有对象的权限和区域与分配给用户的权限和区域匹配，登录用户才能查看和操作对象。Vijeo Citect 安全系统要求必须在工程中定义用户，用于识别用户的用户名和密码对必须在同一工程网络的每个 Vijeo Citect 工作站中一致。用户必须登录到运行时系统才能获得为访问系统配置的权限。

可以为每个操作员（或操作员类别）分配单独的密码，操作员使用系统之前必须输入该密码。

另请参见

[维护用户记录](#)

[定义用户权限](#)

[定义区域](#)

[使用 Vijeo Citect 与 Windows 安全性集成](#)

维护用户记录

您可以为运行时系统的部分（或全部）用户添加登录记录。用户记录强制执行有次序的登录并限制对系统的访问。每个有用户记录的操作员必须输入用户名和口令才能访问运行时系统。

当您配置工程时，可以为每个用户添加用户记录，或者为每类用户（例如，操作员、经理、管理员等）添加一个记录。系统运行时可根据需要（基于定义的类别）添加新用户。每类用户具有共同属性，例如权限。

用户记录和项目恢复

如果从备份重新恢复一个项目，或从一个编译完成而未连线的主工程安装一个新工程，然后用户记录需重新设置来配合工程的原始配置。如果利用运行时建立用户、改变密码或密码过期等功能，运行时的详细资料和未连线的主工程可能不一致。

在此情况下，编译任何未连线工程时，需设置程序确保使用当前 `Users.dbf` 文件（现场运行文件）。这会减少删除运行时建立的用户或在配置并运行一个新系统时锁住期满用户记录的可能性。

注意： 因建立和修改用户而产生的在线变化只反映在本地文件 `_Users.rdb` 和 `Users.dbf` 中。在中央节点上执行所有的用户记录，以便在分布式网络上保持同步。其它节点在 Vijeo Citect 中应使用复制=功能性或自定义类数据库中的复制功能。

另请参见
[添加用户记录](#)

添加用户记录

必须为希望能够使用系统的人员添加用户记录。

添加用户记录:

1. 选择**系统 | 用户**。此时将出现“用户”对话框。
2. 完成“用户”对话框的内容。
3. 点击**添加**以附加新记录，或者点击**替换**以修改现有记录。

使用“用户”对话框为用户定义属性。

用户名

用户的名称。输入一个不超过 16 个字符的值。可以为单个用户分配用户记录，例如：

用户名	J Smith
用户名	John Smith

每个操作员必须输入**用户名**和**密码**才能使用系统。

全名

用户或用户组的全名。输入一个不超过 32 个字符的值。此名称被用作注释，且显示在报警日志和命令日志中。

密码

用户的密码。输入一个不超过 36 个字符的值。输入密码时，将显示星号 (*)，用于替代每个输入的字符。保存用户记录时，密码将在保存到 Users.dbf 之前被加密。

每个操作员必须输入**用户名**和**密码**才能使用系统。

请使用 [General]PasswordExpiry 参数来指定密码过期时间。

确认密码

重新输入用户密码，以确认在**密码**字段中输入的文本。输入一个不超过 36 个字符的值。如果在保存记录时**密码**和**确认密码**字段的内容不同，则将显示一条消息，表明二者不匹配并要求您重试。

全局权限

以全局方式分配给用户的权限类。输入一个不超过 16 个字符的值。

配置系统时，可以将权限分配给各种元素，如图形对象、报警、累积器、命令等。例如，具备全局权限 3 的用户可以发出任何已分配了权限 3 的命令，可以处理任何具备权限 3 的报警，也可以点击任何已分配了权限 3 的按钮等。除非您正在使用区域，否则用户将无法访问任何分配了权限的命令（如果未指定全局权限）。

通过将系统分为多个区域，分配用户权限或对于特定区域分配只查看权限，可以更灵活地进行安全保护。有关详细信息，请使用扩展表单参见此对话框中的 [附加字段](#)（按 **F2** 键）。

注意：全局权限将覆盖已对用户应用的“可查看区域”设置。

类型

通用用户类型。输入一个不超过 16 个字符的值。例如，

类型	操作员
类型	管理员
类型	经理

仅使用此属性来定义那些在运行时通过函数 `UserCreate()` 创建的各个用户所属的用户类。

注释

任何有用的注释。输入一个不超过 48 个字符的值。

注意：

1. [附加字段](#)是在扩展的表单中实现的（按 **F2** 键）。
2. 要注册用户，必须使用 `Login()` 或 `LoginForm()` `Cicode` 函数。

附加字段

可查看区域

允许用户查看的区域。输入一个不超过 16 个字符的值。但是，请记住您仍然必须为这些区域中的元素分配权限，如图形对象、报警、累积器和命令等。否则，用户将完全有权访问这些元素。例如，如果没有为某个区域中的命令分配权限，则用户可发出这个命令。

为了使某个元素（如表达式中的按钮）仅供特定用户查看，请为其分配表达式和权限。将区域添加到用户的可查看区域列表，但不要为用户分配该区域中所需的（或所需的全局）权限。

可以使用组来定义多个区域。

如果没有指定可查看区域，则用户仅有权访问缺省区域（区域 0）。

权限 1..到权限 8 的区域

分配给用户的权限（由区域）。输入一个不超过 16 个字符的值。使用区域与权限的此组合形式，可以为用户分配适用于不同区域的不同权限。例如，在区域 29 和 30 中具备权限类 6 的用户，仅有权访问那些区域中要求权限类 6 的命令。这不影响分配给用户的全局权限（请参见上文）。具备全局权限类 1 和 2 的用户仍可访问所有可查看区域中的具备权限类 1 和 2 的命令。

如果没有指定具有关联权限的区域，则访问仅由“可查看区域”和“全局权限”来定义。

注意：只有当相关区域被列入以上“可查看区域”字段中时，在这些字段中输入的权限才会适用。

Entry 命令

用户登录时执行的 Cicode 命令。用户可以使用任何 Cicode 命令或函数。输入一个不超过 254 个字符的值。

Exit 命令

用户注销时执行的 Cicode 命令。用户可以使用任何 Cicode 命令或函数。输入一个不超过 254 个字符的值。

定义用户权限

要限制对特定系统元素（命令、对象、报表、报警等）的访问，应向其赋予一个**权限**要求，然后将该权限赋予使用它的用户。Vijeo Citect 提供了 8 种权限，编号从 1 到 8。例如，可以向不同类型的操作分配不同的权限，如下表所示：

权限	命令
1	操作传送带
2	操作灶
3	操作制罐容器
4	确认报警
5	打印报表

要使用户可以操作传送带，向用户的注册记录分配权限 1，例如：

全局权限	1
------	---

要允许用户确认报警，向用户的注册记录分配权限 4，例如：

全局权限	4
------	---

要允许用户确认报警和操作传送带，向用户的注册记录分配权限 1 和权限 4：

全局权限	1, 4
------	------

权限分类必须用逗号 (,) 分隔。

要允许用户访问系统的所有命令，在用户注册记录中分配所有权限，例如：

全局权限	1, 2, 3, 4, 5
------	---------------

分配权限后，您可以定义系统元素（命令、报表、对象、报警等）的权限要求：

命令	CONVEYOR = 1;
权限	1
注释	具有权限分类 1 的操作员可以操作传送带
命令	Report("Shift");
权限	5
注释	具有权限分类 5 的操作员可以打印报表

并非所有系统元素都需要权限分类。至少有一个必须由所有用户发布的、用于登录系统的命令：

命令	LoginForm();
权限	
注释	空权限（或权限 0）表示此命令无分类，它适用于所有用户

另请参见：
[使用等级权限](#)

使用等级权限

缺省情况下，权限是非等级的（也就是说，具有权限 3 的用户只能访问分类为 3 的命令）。非等级权限可增加系统灵活性，特别是与区域工具一起使用时。

权限设置为等级权限时，权限 1 最低，权限 8 最高（也就是说，具有权限 3 的用户可以访问权限分类为 3、2、1 的命令）。要分配所有权限，只需指定权限 8。

全局权限	8
------	---

使用权限工具，可以很容易开发安全的 Vijeo Citect 系统。但应在设置系统前小心计划安全方法。需要确定可组合为一类的命令、每类命令的权限和分配给每个操作员的权限。

注意：如果工厂可以划分为几个不连续部分（或区域），则可以通过使用 Vijeo Citect 区域工具添加额外的系统安全性级别。

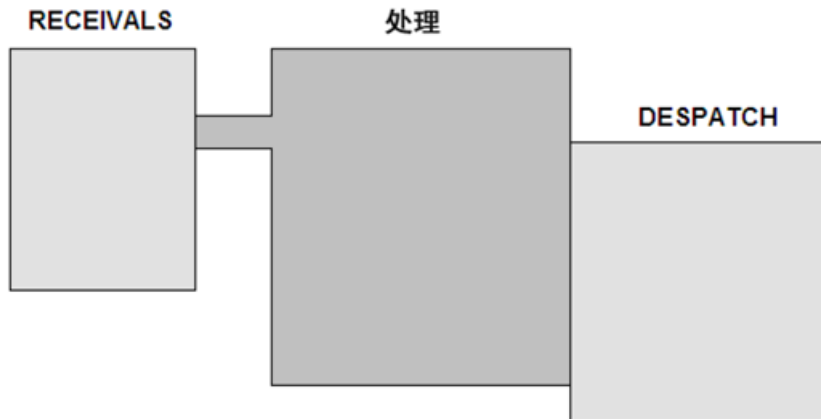
定义区域

当为大型应用程序执行 Vijeo Citect 时，您通常可能会将工厂想象成一系列不连续的部分或区域。您可以从地理位置角度（尤其是当工厂被遥远距离或物理障碍分开时）或从逻辑角度（作为不连续的过程或单个任务）定义这些区域。

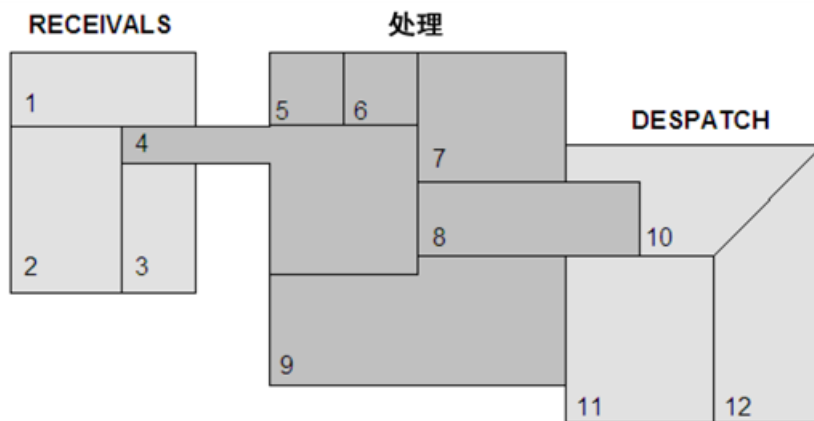
注意：区域工具由扩展属性执行。

通过从区域的角度来考虑自己的工厂，您可以向系统安全加入灵活性。如果没有区域，您只能向用户分配全局权限。拥有全局权限的用户可以通过匹配权限访问系统的任何部分。从另一方面来说，您可以通过区域添加额外的控制级别。您可以针对不同区域向用户赋予不同的权限，而不是赋予全局权限。可以向每一系统元素（对象、报警、报表、累积器等）赋予一个权限要求，并将其分配给一个特定区域。这意味着仅当拥有对所需区域的访问权并拥有该区域的所需权限时，用户才对该区域拥有完全控制权。

某些工厂可以只分为三个区域，原始产品到达收货区，传送到处理区域，然后传送到包装或分派区域。

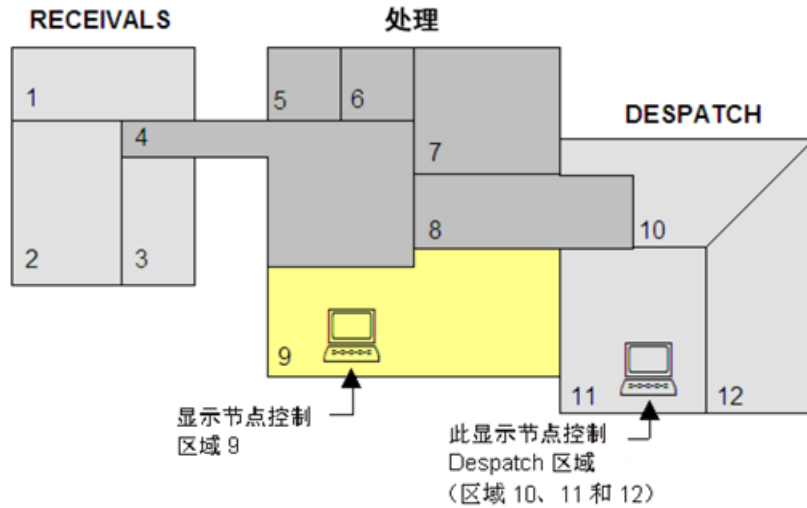


但是，对于规模较大或复杂程度较高的工厂，您可能需要定义数个区域，如：



定义区域时，通常应包括一个由一名操作员（或通过一个 Vijeo Citect 控制客户机）来控制的工厂部门。

还可以定义一些由一个操作员或控制客户端控制的较小区域。这种方法能增加灵活性，但可能会增加系统的复杂性。



您可定义多达 255 个不同区域。然后，您可以按编号（1 至 255）引用这些区域，或使用标签为区域赋予有意义的名称（例如，接收、pre_process、输送等）。

另请参见

- [用区域提高安全性](#)
- [用标签命名区域](#)
- [使用区域组](#)
- [使用带权限的区域](#)
- [指定安全要求](#)
- [查看工厂区域](#)

将区域用于安全性

定义区域后，可以配置命令、对象、报警、报告等，然后操作员可在区域中使用。

例如：

命令	CONVEYOR = 1;
区域	10
注释	此命令属于“区域 10”

在此例中，对“区域 10”没有访问权限的操作员不能发出命令。

另请参见

- [使用标签命名区域](#)

使用标签命名区域

形象的标签（名称）比数字更容易记住。例如：

标签名称	DespatchAccum
表达式	10
注释	标签区域 10 作为“DespatchAccum”

在这种情况下，引用区域 10 时会使用“DespatchAccum”，例如：

命令	CONVEYOR = 1;
区域	DespatchAccum
注释	此命令属于“区域 10”(DespatchAccum)

注意：如果在表单中将“区域”字段保留空白，则该命令不属于任何特定区域，而是分配给工厂的所有区域。

如何为区域添加标签：

1. 选择**系统 I 标记符**。
2. 输入标签的**名称**。
3. 输入要用来替代标签的表达式。
4. 单击**添加**附加新记录，或单击**替换**修改现有记录。

另请参见
[使用区域组](#)

使用区域组

可以将几个区域分组并定义组的名称。

组名称	Despatch
关联 1	DespatchAccum
关联 2	11

关联 3	12
注释	区域 10、11、12 =“Despatch”

在上例中，区域 10、11 和 12 与名称“Despatch”关联。分配给“Despatch”的任何命令都属于区域 10、11 和 12。

命令	CONVEYOR = 1;
区域	Despatch
注释	此命令属于区域 10、11 和 12

还可定义包括其它组的组。

组名称	Plantwide
关联 1	Receivals
关联 2	Process
关联 3	Despatch
注释	将所有区域与“Plantwide”关联

在此例中，名称“Plantwide”是指“Receivals”、“Process”和“Despatch”组中定义的所有区域。

如何定义区域组：

1. 选择**系统 | 组**。将显示“组”对话框。
2. 完成“组”对话框。
3. 单击**添加**附加新记录，或单击**替换**修改现有记录。

另请参见
[组属性](#)

组属性

使用组对话框配置组的属性：

组具有以下属性：

组名称

组的名称。例如，您可以使用此工具定义多个区域或设备。输入一个不超过 16 个字符的值。

您可以在定义组后将其用在能使用单独实体的任何地方。还可通过定义包含组的组来指定复杂的组。

关联 1... 关联 10

一系列与组名称相关联的实体。输入一个不超过 16 个字符的值。关联可以是数字、名称或其它组。还能以 <n1..n2> 格式指定一系列数字，例如：

关联 1	4..10
------	-------

指定数字 4、5、6、7、8、9、10。

还可定义可通过一个名称访问的一组设备，例如：

关联 1	AlarmPrint
关联 2	AlarmLog
关联 3	AlarmDBF

在本例中，当组名称被用作设备时，信息会被发送到所有三个设备 - AlarmPrint、AlarmLog 和 AlarmDBF。

注释

任何有用的注释。输入一个不超过 48 个字符的值。

使用带权限的区域

通过结合区域和权限限制，可以选择操作员在特定区域中的控制权限。即使不使用区域仍可将权限分配给每个操作员 — 允许访问整个工厂（全局权限），但结合区域和权限则可以增加额外的灵活性级别。

用户名	J Smith
全局权限	2, 3

可查看区域	9, 10, 11, 12
权限 1 的区域	Despatch
权限 2 的区域	
权限 3 的区域	
权限 4 的区域	DespatchAccum
权限 5 的区域	DespatchAccum、11
权限 6 的区域	
权限 7 的区域	
权限 8 的区域	
注释	John 的登录

此处，John Smith 具有全局权限 2 和 3；他可以在工厂的可查看区域中使用权限分类为 2 和 3 的命令。他在“Despatch”区域（10、11 和 12）中具有权限 1，在“DespatchAccum”区域（10）中具有权限 4，在区域 10 和 11 中具有权限 5。这表示他可以控制系统元素（报警、报表、累积器、对象等）：

- 位于区域 9，权限要求为 2 或 3。
- 位于区域 10，权限要求为 1、2、3、4 或 5。
- 位于区域 11，权限要求为 1、2、3 或 5。
- 位于区域 12，权限要求为 1、2 或 3。

此外，在此例中，组和标签用于使安全配置更加直观。

另请参见
[指定安全要求](#)

指定安全要求

可以向每一系统元素（对象、报警、报表、累积器等等）赋予一个权限要求，并将其分配给特定区域。以可以确认报警的用户为例，要访问正确区域，他或她需要具有访问该区域所需的权限。

例如：

```
命令      CONVEYOR = 1;
权限      1
```


区域	10
注释	此命令属于区域 10，需要权限 1

这个例子中，操作员没有区域 10 的权限 1 将不能发出命令。

权限 - 区域结合

了解各种区域和权限的结合将以怎样的方式影响您的安全非常重要。

指定权限?	指定区域?	安全结果
是	是	操作员需拥有指定区域所要求的权限。
是	否	安全性由用户的全局权限单独确定。
否	是	对于指定区域，操作员只需具有浏览访问权。
否	否	所有操作员拥有全部控制权。

另请参见

[查看工厂区域](#)

查看工厂区域

操作人员可能需要访问工厂其他区域的信息，而不提供这些区域中的过程控制。例如，一个区域的处理可能直接影响其他区域。

以下例子中，John Smith 控制：

- 带权限要求 2 或 3 的任何系统元素；
- 带权限要求 1 的 **Despatch** 内的系统元素；和
- 带权限要求 4 的 **DespatchAccum** 内的系统元素。

工厂中的其他元素只可查看。

用户名	J Smith
全局权限	2, 3
可查看区域	Plantwide
权限 1 的区域	分派
权限 2 的区域	

权限 3 的区域	
权限 4 的区域	DespatchAccum
权限 5 的区域	
权限 6 的区域	
权限 7 的区域	
权限 8 的区域	
注释	John 的登录名

或者，可将一个操作人员限定到一组区域（例如，"Receivals"）或一个区域（例如，12）。

使用 Vijeo Citect 与 Windows 安全性集成

在 Vijeo Citect 中，现已能够将 Vijeo Citect 用户和安全性选项结合到标准的 Windows 安全系统中。如果更愿意 [在工程中定义用户](#) 并登录到 Vijeo Citect 运行期，当然仍然能够使用现有的 Vijeo Citect 安全性。

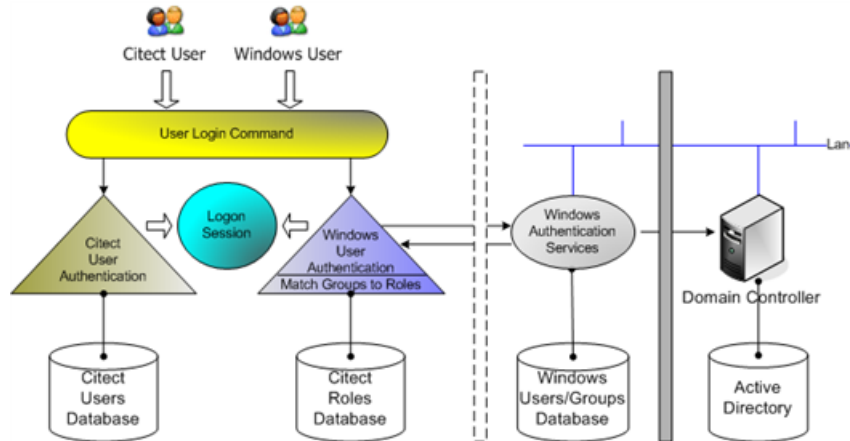
使用集成的 Windows 安全功能，Windows 用户可以使用在工程中配置的运行权限和区域登录到 Vijeo Citect 运行期。对于能够登录运行时的 Windows 用户，必须链接到 Vijeo Citect“角色”，该角色在工程中通过关联的权限定义。

为了将 Windows 用户关联到 Vijeo Citect 角色，应将“角色”作为 Windows 组（该用户为其成员）添加。

如果是与 Citect 角色关联的 Windows 用户，预先存在的 AutoLogin 功能也已经扩展到包括客户端，如果是与 Citect 角色关联的 Windows 用户。为了为 Windows 用户调用此功能，您需要在 Citect.ini 文件中设置 [\[Client\]AutoLoginMode](#)。

如果 Vijeo Citect 用户也是 Windows 用户，则在运行期登录时，该用户始终会继承 Windows 用户的权限。但是，如果有效的 Vijeo Citect 用户因某些原因而登录失败，则不会检查 Windows 用户凭据，并会生成提醒信息以提示登录无效。

Vijeo Citect 安全性和 Windows 安全性的概念图



多签名支持

当 Windows 用户使用其所属角色的关联权限和区域登录运行系统时，如果该用户要执行某些特定操作，有时需要更高级别的身份验证。发生这种情况时，系统会通过 Cicode 显示 [MultiSignatureForm](#)，以允许拥有所需权限级别的另一用户进行身份验证。

另请参见

[添加角色](#)

[使用安全性](#)

添加组和用户

Windows 组和用户由 Windows 管理员和获得授权的超级用户通过 Windows 管理功能进行定义，与 Vijeo Citect 管理不同。

域组和用户是由域管理员在域服务器上定义或创建的。本地组和用户是由本地管理员在本地计算机上定义或创建的。要将 Windows 用户与 Vijeo Citect 角色关联，用户必须是该 Windows 组的成员且组名称必须与 Vijeo Citect 角色中指定的 Windows 组名称一致。

有关如何将组和用户添加到 Windows 安全中的信息，请参考适合于您的操作系统的 Windows 文档。

应注意 Windows 用户不需要成为 Vijeo Citect 用户，这一点非常重要。但是，如果将 Windows 用户添加到与 Vijeo Citect 角色关联的 Windows 组中，则 Windows 用户将拥有指定给该角色的权限。

Windows 管理员通过选择是否要将用户添加到链接的 Windows 组，可以控制哪些 Windows 用户可以登录运行时，哪些 Windows 用户不能登录运行时。

Windows 用户可以与多个角色关联，最多为 256 个角色。如果已与多个角色相关联的 Windows 用户登录 Vijeo Citect 运行期，则将为该用户分配的 Vijeo Citect 权限和区域都是该关联角色的组合权限。

例如：

Citect 用户 "Engineer" 是角色 "Fitter" 的成员，也是角色 "Technician" 的成员。

角色 "Fitter" 拥有全局权限 P1 和 P3，以及查看访问权 A1 和 A4。

角色 "Technician" 拥有全局权限 P2 和 P5，以及查看访问权 A2 和 A3。

则用户 "Engineer" 拥有全局权限 P1、P2、P3 和 P5，以及查看访问权 A1、A2、A3 和 A4。

有关使用权限和区域的进一步信息，请参阅 [指定安全要求](#)

添加角色

必须为这些用户添加，才能将系统用作 Windows 安全组的成员。虽然不是必须的操作，但在向 Vijeo Citect 的组中添加角色之前，一个良好的习惯是添加一个 [Windows 安全组](#)，或确定一个现有的组。否则，您可能向不应具有该角色权限的组中添加角色。

添加角色记录：

1. 选择 **系统 | 角色** 以显示“角色”对话框。
2. 完成“角色”对话框的内容。
3. 点击 **添加** 以附加新记录，或者点击 **替换** 以修改现有记录。

使用“角色”对话框为角色定义属性。

角色名称

输入一个不超过 16 个字符的值，例如“Operator”角色名称限制使用与标签名称相同的语法。参见 [标签名称语法](#)。

Windows 组名称

输入您希望与之关联的 Windows 安全组的组名称。验证该名称是否与您希望将此角色连接到的 Windows 中的组名称相同。它最多可以包含 254 个大写或小写字符。验证是否仅使用了 Windows 中 Windows 组帐户所允许的字符。

Windows 组名称可以包括域名或本地计算机名称，格式分别为 "domainname\operator" 或 "localcomputername\operator"。如果组名称中指定了两者，则 Vijeo Citect 运行时将仅验证名称中指定的服务器上的组或本地计算机上的组。

注释

任何有用的注释。输入一个不超过 48 个字符的值。

全局权限

以全局方式分配给角色的权限类。输入一个不超过 16 个字符的值。

配置系统时，可以将权限分配给各种元素，如图形对象、报警、累积器、命令等。例如，具备全局权限 3 的角色可以发出任何已分配了权限 3 的命令，可以处理任何具备权限 3 的报警，也可以点击任何已分配了权限 3 的按钮等。除非您正在使用区域，否则角色将无法访问任何分配了权限的命令（如果未指定全局权限）。

注意： 全局权限会覆盖应用于某个角色的“可查看区域”设置。

通过将系统分为多个区域，分配角色权限或对于特定区域分配只查看权限，可以更灵活地进行安全保护。

[Ref1411470390](#)

在完成了此对话框中的字段后，如果在添加角色之前尚未完成，则需要将用户添加到您希望具有此角色权限的 Windows 安全组中。

另请参见

在 [Windows 安全中添加组和用户](#)。

此对话框中的 [附加字段](#) 使用的是扩展形式（按 **F2**）。

附加字段

可查看区域

允许用户查看的区域。输入一个不超过 16 个字符的值。但是，请记住您仍然必须为这些区域中的元素分配权限，如图形对象、报警、累积器和命令等。否则，用户将完全有权访问这些元素。例如，如果没有为某个区域中的命令分配权限，则用户可发出这个命令。

为了使某个元素（如表达式中的按钮）仅供特定用户查看，请为其分配表达式和权限。将区域添加到用户的可查看区域列表，但不要为用户分配该区域中所需的（或所需的全局）权限。

可以使用组来定义多个区域。

如果没有指定可查看区域，则用户仅有权访问缺省区域（区域 0）。

权限 1.. 到权限 8 的区域

分配给用户的权限（由区域）。输入一个不超过 16 个字符的值。使用区域与权限的此组合形式，可以为用户分配适用于不同区域的不同权限。例如，在区域 29 和 30 中具备权限类 6 的用户，仅有权访问那些区域中要求权限类 6 的命令。这不影响分配给用户的全局权限（请参见上文）。具备全局权限类 1 和 2 的用户仍可访问所有可查看区域中的具备权限类 1 和 2 的命令。

如果没有指定具有关联权限的区域，则访问仅由“可查看区域”和“全局权限”来定义。

注意：只有当相关区域被列入以上“可查看区域”字段中时，在这些字段中输入的权限才会适用。

Entry 命令

用户登录时执行的 Cicode 命令。用户可以使用任何 Cicode 命令或函数。输入一个不超过 254 个字符的值。

Exit 命令

用户注销时执行的 Cicode 命令。用户可以使用任何 Cicode 命令或函数。输入一个不超过 254 个字符的值。

方案和使用情况

下列示例显示了当用户使用 Windows 组和 Vijeo Citect 角色时，如何在各种情况下允许用户与 Vijeo Citect 运行时系统进行协商。

本地用户登录

当验证 Windows 用户身份而且“角色|组名称”不包含域路径时，或者当本地计算机名称被显式指定时，Windows 将仅验证本地用户的身份。

域登录

当验证 Windows 用户身份而且“角色|组名称”包含域名时，Windows 将尝试验证 Windows 域用户和用户组的身份。如果域控制器不可用，则将使用缓存的凭据和 Citect 组名称（如果可用）。

注意：

Web 客户端不支持缓存的凭据。

如果用户使用 SE_TCB_NAME 权限登录，Windows 2000 将仅利用缓存的凭据。

本地客户机身份验证

当在作为域的一部分的控制客户机或仅查看客户机上执行 Vijeo Citect Windows 登录时，客户机本身将负责向域证明身份。Vijeo Citect 服务器仅验证帐户是否存在，它不执行身份验证。

远程客户机身份验证

当在作为域或受信域的一部分的远程客户机上执行 Vijeo Citect Windows 登录时，客户机本身将负责向域证明身份。Vijeo Citect 服务器仅验证帐户是否存在，它不执行身份验证。从本质上讲，此机制与本地客户机身份验证相同。

Web 客户机身份验证

当在不是配置域的成员的 Web 客户机上执行 Vijeo Citect Windows 登录时，服务器将负责对域中的用户进行身份验证。不在 Web 客户机计算机上进行本地 Windows 身份验证。在这种情况下不能发生自动登录的情况。

多域身份验证

当在作为域的一部分的控制客户机或仅查看客户机上执行 Vijeo Citect Windows 登录时，客户机本身将负责向域证明身份。当该客户机能够访问多个域中的 Vijeo Citect 服务器时，则该客户机可能只在一个域中证明了自己的身份。

CtAPI 身份验证

在此 CtAPI 发行版中不支持使用 Windows 安全性。

Chapter: 12 配置您的系统

在运行工程之前，您需要配置 Vijeo Citect 系统中的每台计算机。配置信息存储在每台计算机上的 `citect.ini` 文件中，包括安装、数据库配置和编译的工程等信息。配置由“计算机设置精灵”完成。

“计算机设置精灵”包含一系列画面，允许配置一些最重要的计算机特定设置，包括：

- 计算机在 Citect 网络中具有的角色
- 被运行的工程
- CPU 配置
- 为每个组件启用的 Citect 事件
- 启动时为每个组件运行的 Cicode
- 集群配置
- 应用的安全设置

精灵使用工程数据库中存储的配置向您提供信息。选定的选项将被写入到 `citect.ini` 文件中。

必须在您的系统中运行 Vijeo Citect 的每台计算机上运行精灵，以便为每台特定的计算机配置 Vijeo Citect。应该在编译工程之后、运行系统之前作为最后一步运行精灵。

另请参见

[运行计算机设置向导](#)

运行计算机设置向导

如何启动 Citect 计算机设置向导：

1. 打开 Citect 浏览器。
2. 在工程列表区域中，选择**我的工程**——由一个计算机图标表示。
3. 双击**计算机设置向导**图标，或选择**工具 | 计算机设置向导**。这时将显示 Citect 计算机设置向导。
4. 选择**快速设置**或**自定义设置**。

显示的页面取决于计算机配置和以下配置：

- [工程配置](#)
- [计算机角色配置](#)
- [网络模式](#)
- [Internet 服务器配置](#)
- [报警配置](#)
- [报表配置](#)
- [趋势配置](#)
- [CPU 配置](#)

- [事件配置](#)
- [启动函数配置](#)
- [集群连接配置](#) *
- [控制菜单安全配置](#) *
- [键盘安全配置](#) *
- [其它安全配置](#) *
- [常规选项设置](#) *

* 仅在自定义设置模式下可用。

每屏向导都在屏幕后面的部分进行了介绍。

另请参见
[工程配置](#)

工程配置

选择要在此 Vijeo Citect 计算机上运行的工程。计算机设置向导将向您显示除包含工程之外，所有在工程列表中定义的已编译工程。

如果只有一个已编译工程，它将被自动选中。如果没有已编译工程，将显示提醒消息，并且向导将终止。如果出现这种情况，返回 Citect 浏览器并确认所需工程已被保存在本地并且已无错误地编译。

另请参见
[计算机角色配置](#)

计算机角色配置

使用“计算机角色设置”画面可以指定运行 Vijeo Citect 的计算机的角色。从下述选项中选择一个。

注意：要使用 Vijeo Citect 的多进程功能，必须启用联网。

选项	描述
服务器和控制客户端	<p>此计算机将是单机或联网 I/O 服务器和控制客户端。如果未向此计算机赋予要运行的服务器组件，此选项将被禁止。选择此选项将启用多进程复选框。</p> <p>选择多进程复选框可将客户端和服务组件分为单独的进程。可以使用此选项跨多个 CPU 分布组件。</p> <p>如果将多进程复选框留为未选中状态，Vijeo Citect 将在一个进程中运行客户端和所有服务器组件。</p> <p>如果已选中多进程复选框，Citect.ini 文件中的 [General]MultiProcess 参数将与值 1 一起保存。如果未选中，此参数将与值 0 一起保存。</p>
控制客户端	<p>此计算机将只作为控制客户端。如果向此计算机赋予了要运行的服务器组件，此选项将被禁止。选择此选项将启用全项许可证复选框。</p>

选项	描述
	如果希望此控制客户端使用全项许可证，选择 全项许可证 复选框。这可以将 Citect.ini 文件中的 [Client]FullLicense 参数设置为 1（默认值）。
仅查看客户端	此计算机将只作为仅查看客户端。如果向此计算机赋予了要运行的服务器组件，这一只读选项将被禁止。

根据哪些服务器被配置为在此计算机上运行，上述某些选项可能会被禁止。计算机设置向导通过在工程配置中为每个服务器配置的网络地址交叉参考计算机的网络标识。

另请参见

[网络模式](#)

[CPU 配置](#)

网络模式

选择要应用到此 Vijeo Citect 计算机的网络模式。选项包括：

- 无网络
- TCP/IP

“无网络”选项在选择了多过程选项时不可用。从版本 7.0 开始，Vijeo Citect 使用 TCP/IP 简化网络之间的通讯。

注意： Citect 服务器的 TCP/IP 地址信息在 Citect 工程本身中进行配置。有关详细信息，请参见 [网络地址定义](#)。

完成“计算机设置向导”后，所选网络模式会写入 citect.ini 文件中的 [LAN] 部分；例如：

```
...
[LAN]
TCPIP=1
...
```

另请参见

[Internet 服务器配置](#)

互连网服务器配置

选中**此计算机作为互联网服务器**选项可将该计算机作为互联网服务器。为了能够与远程互联网显示客户机进行通信，互联网服务器需要永久互联网连接和静态 IP 地址（或主机名）。

1. 在**客户机连接信息区域**中，键入**互联网服务器IP地址或主机名**。如果没有预先存在的输入，此键入将添加 **Primary=<Internet Server IP address>** 输入到 **citect.ini** 文件的 **[DNS]** 部分。有关详细信息，请参见 [\[DNS\] 首选](#)。

注意：要确定互联网服务器计算机的 TCP/IP 地址，请选择**开始 | 运行**。键入 **CMD** 并按 **Enter**。然后在 DOS 提示符处键入 **IPCONFIG** 并按 **Enter**。

2. 键入**备用互联网服务器 IP 地址或主机名**。如果没有预先存在的输入，此键入操作会将 **Standby=<Alternate Internet Server IP address>** 输入添加到 **citect.ini** 文件的 **[DNS]** 部分。有关详细信息，请参见参数中的“**[DNS] 备用**”联机帮助。

如果互联网显示客户机与首选服务器的连接丢失，它将自动连接到此备用服务器。

注意：这种处理机制只有在已经与首选互联网服务器建立了初始连接的情况下才能自动发生。

另请参见
[报警配置](#)
[DNS参数](#)

报警配置

只有在将此计算机配置为工程编辑器中的警报服务器时才会显示“报警配置”页。

Vijeo Citect 具有多个可用于处理报警的选项：

选项	描述
报警扫描时间	<p>确定对报警进行扫描以及处理的速率。值 250（缺省值）表示 Vijeo Citect 每 500 毫秒尝试处理一次报警。但是，如果 Vijeo Citect 无法在 500 毫秒内从 I/O 设备中读取全部的报警数据，报警的处理速度将会变慢。例如，如果从 I/O 设备读取所有报警数据需要 800 毫秒，那么 Vijeo Citect 将会每 800 毫秒对报警处理一次。</p> <p>如果选择更大的值作为报警扫描时间，警报服务器对 CPU 的占用也将减小（因为无需经常对报警记录进行处理）。从 I/O 设备读取的数据量也将减少，因此其它的过程（趋势、报表、以及当前页面）将会更快地从 I/O 设备获取数据。可以输入从 0 到 60000 之间的值（毫秒）。</p>
报警保存周期	<p>保存报警和事件数据（到磁盘）的周期。用户可以定期地将报警以及时间数据进行保存，以便在系统按计划或未按计划关闭后仍然能够将数据重新恢复。请注意该周期越短，系统的负担越大。</p>
摘要长度	<p>内存中可以保存的报警摘要条目的最大数目。可以在报警描述页面上来查看这些报警摘要条目。请注意每一个事件及其注释内容都需要 62 个字节的内存空间。32,000 个事件将需要至少 1.9 MB 的内存空间。如果使用了很多事件，RAM 中应当有足够的内存来进行存放。</p>
摘要超时	<p>报警摘要条目在报警摘要队列中保留的时间长度。</p>

选项	描述
主警报服务器保存路径	主保存文件的路径。Vijeo Citect 对每个警报服务器使用两个保存文件，ALMSAV.DAT 和 ALMINDEXSAVE.DAT。保存主路径是主警报服务器在其中创建其保存文件的目录。在对文件进行恢复时，将使用最新的（主警报服务器和辅警报服务器中）保存文件。
辅警报服务器保存路径	备用保存文件的路径。

为将同一计算机上运行的不同集群中的多个警报服务器所使用的报警文件之间的冲突机会最小化，报警文件需具备基于以下格式的动态命名约定：

```
<ProjectName>_<ClusterName>_<filename>.DAT
```

另请参见
[报表配置](#)

报表配置

只有在将此计算机配置为工程编辑器中的报表服务器时才会显示“报表配置”画面。

注意： 一台在网络中的计算机要成为报表服务器，它必须同时还是 I/O 服务器或者能够在网络中同 I/O 服务器进行通讯。

Vijeo Citect 具有多个可用于处理报表的选项：

选项	描述
启动报表	定义 Vijeo Citect 启动时要运行的报表的名称。
启动时禁止触发报表	例如，在启动时用户可能有一个报表是由某一个数字位的上升沿进行触发的。报表服务器在检测到该比特位时运行报表。当本选项被选中时，报表服务器将不会运行该报表，直到它第二次读取到该 I/O 设备。
同时运行报表和主报表服务器	启用或者禁用报表的前后处理机制。当服务器为备用报表服务器时，它可以同主报表服务器一起一前一后地对所有报表进行处理，或者保持闲置状态直到被调用。

另请参见
[趋势配置](#)

趋势配置

只有在将此计算机配置为工程编辑器中的趋势服务器时才会显示“趋势配置”画面。

注意： 一台在网络中的计算机要成为趋势服务器，它必须同时还是 I/O 服务器或者能够在网络中同 I/O 服务器进行通讯。

Vijeo Citect 具有一个可用于处理趋势的选项：

选项	描述
启动时禁止触发趋势	在启动时，用户可能有一个在某一个比特位的上升沿被触发的趋势。当本选项被选中时，趋势服务器将不显示此趋势，直到其第二次读取到 I/O 设备。

另请参见

[CPU 配置](#)

CPU 配置

“CPU 设置”画面用于在多处理器计算机上将客户机和服务器组件指定给特定处理器。

此画面列出每个组件的完整名称，包括所属集群、优先级和 CPU 分配情况。如果未在 [计算机角色配置](#) 画面上选择“多处理”选项，将只列出“客户机”或“客户机和服务器”中的一个条目。如果选中了“多处理”选项，则可以为客户机、I/O 服务器、报警服务器、趋势服务器和报表服务器选择特定的 CPU。

为组件指定 CPU：

1. 从列表选择一个或多个组件（按住 **Ctrl** 键可选择多个组件）。
2. 单击**修改**。
3. 键入 CPU 的编号，单击**确定**。

完成“计算机设置向导”后，CPU 分配情况就会写入到 `citect.ini` 文件中的每个组件部分；例如：

```
...
[Alarm.Cluster1.AlarmServer1]
CPU=1
Clusters=Cluster1
...
[Trend.Cluster1.TrendServer1]
CPU=2
Clusters=Cluster1
...
```

另请参见
[事件配置](#)

事件配置

事件可用于触发动作，例如一个命令或一组命令。例如，当某个生产过程完成之后通知操作人员，或者当生产过程达到某个特定的步骤时执行一系列的指令。如果要在此 Vijeo Citect 计算机上启用事件，请选中**在此计算机上启用事件**复选框。

“事件设置”画面列出了每个组件的完整名称（包括它所属的集群），以及可以为每个组件启用的事件列表。如果未在 [计算机角色配置](#) 画面上选择“多处理”选项，将只列出“客户端”或“客户端和服务端”中的一个条目。如果选择了“多处理”选项，则可以选择为此计算机上的每个组件启用事件。

注意：“计算机设置向导”仅显示所选工程的已命名事件。如果要在包括的工程中使用事件，您需要编辑 `citect.ini` 文件，以便将这些事件添加到 [Events] 章节标题之下。

注意：名称为“全局”或者没有标题的事件将不会出现，因为这些是全局事件。这些事件将在所有启用了事件的计算机上运行。这些事件将在客户机处理中运行。

如何启用组件的事件：

1. 从列表中选择组件。
2. 选择希望为该组件启用的事件，或点击**全部启用**或**全部禁用**。
3. 完成时点击**下一步**。

完成“计算机设置向导”后，事件就会写入到 `citect.ini` 文件中的每个组件部分；例如：

```
...
[Alarm.Cluster1.AlarmServer1]
CPU=1
Clusters=Cluster1
Events=CSV_AlarmClient...
[Trend.Cluster1.TrendServer1]
CPU=2
Clusters=Cluster1
Events=CSV_TrendXClient,CSV_TrendXServer...
```

另请参见
[启动函数配置](#)

启动函数配置

“启动函数设置”页面用于定义由每一 Vijeo Citect 进程执行的启动 Cicode。

“启动函数设置”页面将列出每个组件的完整名称（包括它所属的集群）、组件的属性以及指定给每个组件的启动函数。如果未在 [计算机角色配置](#) 画面上选择“多进程”选项，将只列出“客户端”或“客户端和服务端”中的一个条目。如果选择了“多进程”选项，则可以选择为此计算机上的每个组件指定启动函数。

如果某个进程的 StartupCode 参数值无效，Vijeo Citect Runtime 管理器会在启动时直接将其忽略。

为组件指定启动函数：

1. 从列表中选择组件。要选择多个组件，请在选取每个项时按住 Ctrl 键。
2. 点击**修改**。
3. 键入要在启动时为该组件调用的 Cicode 函数的名称。
4. 点击**确定**。

完成“计算机设置向导”后，事件就会写入到 citect.ini 文件中的每个组件部分；例如：

```
...
[Alarm.Cluster1.AlarmServer1]
CPU=1
StartupCode=alarmServerStartup
...
[Trend.Cluster1.TrendServer1]
CPU=2
StartupCode=trendServerStartup
...
```

另请参见

[CPU 配置](#)

[集群连接配置](#)

集群连接配置

“集群连接设置”画面用来指定每个组件启动时连接到的集群。这控制组件可以在系统中看见的数据流。

“集群连接设置”画面将列出每个组件的完整名称（包括它所属的集群）、组件的属性以及指定给每个组件的集群。如果未在 [计算机角色配置](#) 画面上选择“多处理”选项，将只列出“客户端”或“客户端和服务端”中的一个条目。如果选择了“多处理”选项，则可以选择为此计算机上的每个组件指定集群。

缺省情况下，所有组件将连接到所有集群，除非进行了修改。

如果某个进程的“集群”参数值无效，则 Vijeo Citect 运行期间在启动时就会将其忽略。

为组件指定集群：

1. 从列表中选择组件。要选取多个组件，请在选取每个项时按住 **Ctrl** 键。
2. 单击**修改**。
3. 选择要让组件在启动时连接到的集群。
4. 单击**确定**。

完成“计算机设置向导”后，集群就会写入到 `citect.ini` 文件中的每个组件部分；例如：

```
...
[Alarm.Cluster1.AlarmServer1]
CPU=1
Clusters=Sydney
...
[Trend.Cluster1.TrendServer1]
CPU=2
Clusters=Sydney,Tokyo
...
```

另请参见
[实施集群](#)

控制菜单安全配置

Vijeo Citect 窗口属性选项允许您控制操作员的系统访问权限。

这样使得在运行期间系统将具有较灵活的安全性。

选项	描述
菜单配置环境	允许操作员在运行期间使用控制菜单（左上方的图标）从 Vijeo Citect 中访问 Citect 编辑器、工程编辑器、图形编辑器和 Cicode 编辑器。禁用该选项将提供更好的安全性。
显示标题栏	<p>允许标准的 Windows 标题栏在运行期间显示在用户的 Citect 窗口上方。禁用该选项将提供更好的安全性：画面将呈现为全屏状态。用户将无法使用窗体的标题、最大化和最小化按钮（在标题栏的最右端）、以及控制菜单按钮（在标题栏的最左端）。</p> <p>以全屏模式进行显示的页面将占用整个的屏幕显示区域（假定不影响其长宽比例），而且不能够对页面的大小进行调整。对于单个页面可以通过在页面创建时选择标题栏（Title Bar）选项，或者在页面创建后在页面属性中来对该选项进行修改。</p> <p>改变页面的大小可能降低图像的质量。如果无法接受这种情况，可以使用需要的分辨率重新设计画面。</p>
在菜单上显示“关闭”菜单项	允许操作员在运行期间通过控制菜单（左上图标）来关闭 Vijeo Citect 。该关闭操作不受密码以及权限的保护。禁用该选项将提供更好的安全性。
在菜单上显示内核 (Kernel) 菜单项	允许操作员在运行期间通过控制菜单（左上图标）显示 Vijeo Citect 内核。禁用该选项将提供更好的安全性。

另请参见
[键盘安全配置](#)

键盘安全配置

Vijeo Citect 在运行期间支持（可选）一系列的 Windows 标准任务切换快捷命令。该选项允许在运行期间启用或禁用 Windows 的 Alt- 空格命令。用户可通过 Alt- 空格命令访问 Windows 控制菜单（即便标题栏已禁用也无妨）。

注意：目前尚不能禁用 Alt-Escape、Ctrl-Escape 和 Alt-Tab。

另请参见
[其他安全配置](#)

其它安全配置

某些标准 Windows 特性可能会干扰系统的安全操作。使用“其它安全”页面可以禁止这些特性。

选项	描述
Vijeo Citect 运行时禁用屏幕保护程序	避免屏幕保护程序影响应始终可见的重要屏幕。另外，屏幕保护程序密码可以提高安全性。
启动时显示“取消”按钮	提供防止 Vijeo Citect 自动启动的功能。自动启动是潜在的安全性问题

另请参见
[常规选项设置](#)

常规选项设置

可使用“常规选项设置”画面指定常规选项。

选项	描述
数据目录	Vijeo Citect 日志文件所在的目录。数据文件是在运行时生成的文件：趋势文件、磁盘 PLC 等
备份工程路径	在无法找到某运行数据库（由于硬件不正常或文件已被移动、破坏或删除）时使用的备

选项	描述
	份目录。
启动画面	当 Vijeo Citect 启动时需要进行显示的图形画面的画面名称。
画面扫描时间	<p>在对图形画面进行更新和开始下一个通信循环之间的延迟（以毫秒计）。画面扫描时间设置图形画面的缺省更新频率。更新画面时，所有相关的数据（图形画面上显示的变量标签等）将被扫描以确定现场条件是否已发生了变化。此设置将由在画面属性中指定的扫描时间覆盖（在改变生效后）。</p> <p>值 250（缺省值）表示 Vijeo Citect 每 250 毫秒尝试更新一次画面。但是，如果 Vijeo Citect 无法在 250 毫秒内从 I/O 设备中读取全部的数据，画面的处理速度将会变慢。例如，如果从 I/O 设备读取所有数据需要 800 毫秒，那么 Vijeo Citect 将会每 800 毫秒对画面处理一次。</p> <p>在某些条件下，您可能需要减慢画面的更新速度以减轻 I/O 服务器的负荷。通过缩短画面扫描时间，可以增加与其它 Vijeo Citect 任务或客户机之间的通信带宽。例如，您可能希望加快主操作员计算机的响应速度，同时减慢管理员计算机的响应速度。可以输入从 0 到 60000（毫秒）的任何值。</p>

另请参见
[完成](#)

完成

点击**完成**可将设置保存到 citect.ini 文件，点击**取消**可不保存退出向导，也可以点击**返回**导航到需要调整的画面。

Chapter: 13 实施集群

为系统设置集群（包括您需要的服务器配置）后，您可以继续实施该设计。您需要配置：

- [集群定义](#)

每个集群都必须通过在工程中为其提供唯一名称加以定义。

- [网络地址定义](#)

系统中的每个物理服务器都必须用唯一的名称和 IP 地址加以标识。

- [报警服务器定义](#)

每个报警服务器都必须命名并分配到一个集群和物理服务器。每个服务器都应标识为“主要”或“备用”。

- [报表服务器定义](#)

每个报表服务器都必须命名并分配到一个集群和物理服务器。每个服务器都应标识为“主要”或“备用”。

- [趋势服务器定义](#)

每个趋势服务器都必须命名并分配到一个集群和物理服务器。每个服务器都应标识为“主要”或“备用”。

- [I/O服务器定义](#)

每个 I/O 服务器都必须命名并分配到一个集群和物理服务器。每个服务器都应标识为“主要”或“备用”。

另请参见

[集群的规则](#)

[在运行时为集群分配标签](#)

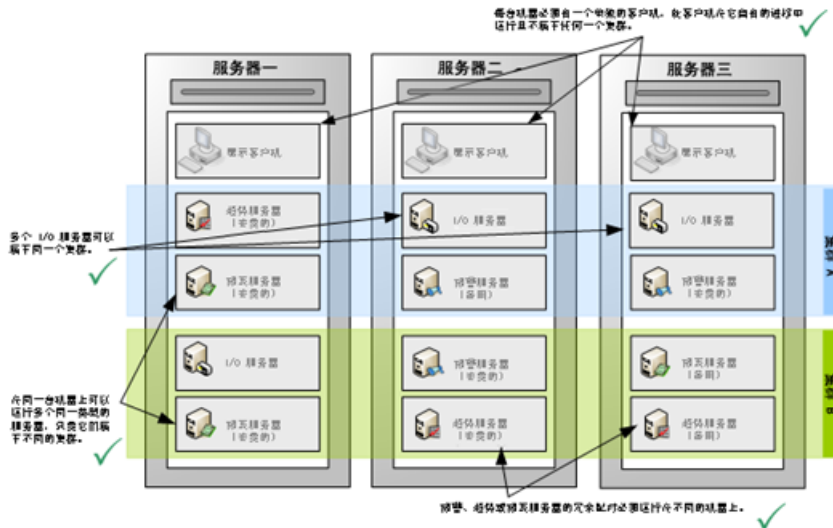
集群的规则

配置 Vijeo Citect 时，将应用以下集群规则：

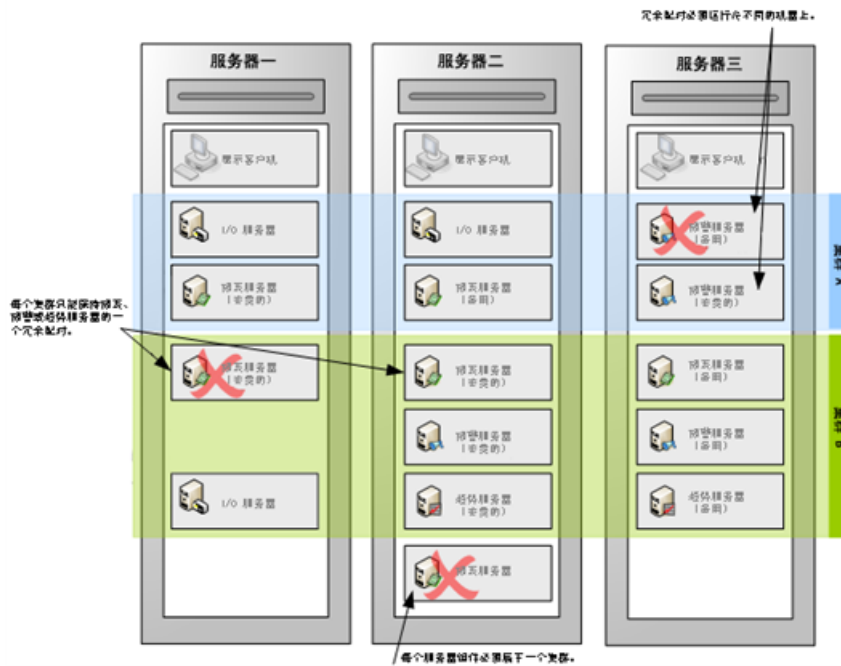
- 每个集群的名称必须唯一。
- 每个服务器组件的名称必须唯一。
- 每个服务器组件必须属于一个集群。
- 每个集群只能包含一对冗余报警服务器。这些服务器必须驻留在不同计算机。
- 每个集群只能包含一对冗余报表服务器。这些服务器必须驻留在不同计算机。
- 每个集群只能包含一对冗余趋势服务器。这些服务器必须驻留在不同计算机。
- 每个集群可以包含无限数量的 I/O 服务器。

集群式系统的配置方式可采用无数变体。最适当的配置将取决于要部署的解决方案的**要求**，以及对其进行部署的**环境**。有关详细信息，请参阅 [典型系统方案](#)。

下图举例说明了在三台计算机中运行两个集群的系统。所有服务器和客户机部件都已根据集群规则进行了部署。



下一个图表说明了没有正确遵循集群规则的状况。



Vijeo Citect 编译器或 Vijeo Citect 运行时管理器检测用户何时未遵守集群规则，并对用户提出相应的建议。

另请参见
[关于集群环境](#)

集群定义

要获取更多信息，请参见 [集群的规则](#)。

定义集群：

1. 在工程编辑器中，选择**服务器 | 集群**。
2. 在“集群”对话框中，完成集群属性：

选项	描述
集群名称	集群的名称（最多 16 个字符）。名称对工程必须唯一，而且不能包含空格。
注释	任何有用的注释（最多为 48 个字符）。此属性为可选属性，在运行期间不使用。

3. 单击**添加**按钮附加一条新记录，如果已经修改了一条记录，请单击**替换**。

另请参见

[网络地址定义](#)

网络地址定义

如何配置网络地址：

1. 在工程编辑器中，选择**服务器 | 网络地址**。
2. 在“网络地址”对话框中，完成以下属性：

选项	描述
名称	在要配置的网络地址处的计算机名称（最多 16 个字符）。名称对于工程必须唯一，且一定不能包含空格。
地址	IP 地址、计算机名或要配置的网卡（最多 32 个字符）。对于具有双重网卡的计算机，需对要通讯的每台计算机中的每个网卡添加网络地址。有关信息，请参见 网络冗余 。
注释	任何有用的注释（最多为 48 个字符）。此属性是可选属性，不会在运行时使用。

3. 单击**添加**按钮附加新记录或单击**替换**（如果已对记录进行了修改）。

另请参见

[警报服务器定义](#)

警报服务器定义

请注意 [配置服务器](#) 下方的缺省服务器端口号。

配置警报服务器：

1. 在“工程编辑器”中，选择**服务器 | 警报服务器**。
2. 在“警报服务器”对话框中，完成以下属性：

选项	描述
集群名称	此警报服务器所属的集群的名称（最多 16 个字符）。如果只在工程中定义了一个集群，则可以将此字段保留为空。警报服务器将缺省作为已定义的集群
服务器名	服务器的名称（最多为 16 个字符）。名称对工程 and 任何包括的工程而言必须是唯一的，而且不得包含空格。
模式	此服务器的模式，“主要”或“备用”（最多 16 个字符）。如果将此字段保留为空，则缺省值将为“主”。主和辅服务器必须运行于不同的计算机上，且在每个集群中只能定义一个主服务器和一个辅服务器。
网络地址	要配置的服务器的网络地址（最多 70 个字符）。要指定该服务器的双向网络连接，请使用以逗号分隔的列表。请参见 网络冗余 。
端口	此服务器将侦听的端口（最多 16 个字符）。如果您在计算机上运行一个警报服务器（在这种情况下将会使用缺省端口），则可以将此字段保留为空。
注释	任何有用的注释（最多为 48 个字符）。此属性是可选的，在运行时间不使用。

扩展形式的字段

以下字段通过扩展形式实现（按 **F2**）。

选项	描述
发行警报属性	TRUE 或 FALSE。缺省情况下此值为 FALSE。转换为 TRUE 时，将发行警报属性，并将报警属性视为正常的变量标签，并且警报服务器将侦听，就好像它是 I/O 连接器一样。
端口	警报服务器将在其上进行的侦听的端口（如果将“发布报警”属性设置为 TRUE）。（最多 16 个字符）

3. 点击**添加**按钮以附加新记录，或者点击**替换**（如果已修改记录）。

另请参见
[报表服务器定义](#)

报表服务器定义

请注意 [配置服务器](#) 下方的默认服务器端口号。

如何配置报表服务器：

1. 在“工程编辑器”中，选择**服务器 | 报表服务器**。
2. 在“报表服务器”对话框中，完成以下属性：

选项	描述
集群名称	此报表服务器所属的集群的名称（最多 16 个字符）。如果只在工程中定义了一个集群，则可以将此字段保留为空。报表服务器将缺省作为已定义的集群
服务器名	服务器的名称（最多 16 个字符）。名称对于工程必须唯一，且一定不能包含空格。
模式	此服务器的模式，“主要”或“备用”（最多 16 个字符）。如果将此属性保留为空，则默认参数值将为“主要”。主和备用服务器必须运行于不同的计算机上，且在每个集群中只能定义一个主服务器和一个备用服务器。
网络地址	要配置的服务器的网络地址（最多 70 个字符）。要指定到服务器的双向网络连接，请使用逗号分隔列表。请参见 网络冗余 。
端口	此服务器将侦听的端口（最多 16 个字符）。如果您在计算机上仅运行一个报表服务器（在这种情况下将会使用缺省端口号），则可以将此字段保留为空。
注释	任何有用的注释（最多为 48 个字符）。此属性是可选属性，不会在运行时使用。

3. 单击**添加**按钮附加新记录或单击**替换**（如果已对记录进行了修改）。

另请参见
[趋势服务器定义](#)

趋势服务器定义

请注意 [配置服务器](#) 下方的默认服务器端口号。

如何配置趋势服务器：

1. 在“工程编辑器”中，选择**服务器 | 趋势服务器**。
2. 在“趋势服务器”对话框中，完成以下属性：

选项	描述
集群名称	此趋势服务器所属的集群的名称（最多 16 个字符）。如果只在工程中定义了一个集群，则可以将此字段保留为空。趋势服务器将缺省作为已定义的集群。
服务器名	服务器的名称（最多 16 个字符）。名称对于工程必须唯一，且一定不能包含空格。
模式	此服务器的模式，“主要”或“备用”（最多 16 个字符）。如果将此属性保留为空，则默认参数值将为“主要”。主和备用服务器必须运行于不同的计算机上，且在每个集群中只能定义一个主服务器和一个备用服务器。

选项	描述
网络地址	要配置的服务器的网络地址（最多 70 个字符）。要指定到服务器的双向网络连接，请使用逗号分隔列表。请参见 网络冗余 。
端口	此服务器将侦听的端口（最多 16 个字符）。如果您在计算机上仅运行一个趋势服务器（在这种情况下将会使用缺省端口号），则可以将此字段保留为空。
注释	任何有用的注释（最多为 48 个字符）。此属性是可选属性，不会在运行时使用。

- 单击**添加**按钮附加新记录或单击**替换**（如果已对记录进行了修改）。

另请参见
[I/O 服务器定义](#)

I/O 服务器定义

请注意 [配置服务器](#) 下方的缺省服务器端口号。

如何配置 an I/O 服务器

- 在工程编辑器中，选择**服务器 | I/O 服务器**。
- 在“I/O 服务器”对话框中，完成以下属性。

选项	描述
集群名称	此 I/O 服务器所属的集群的名称。如果只在工程中定义了一个集群，则可以将此字段保留为空。I/O 服务器将缺省为已定义的集群。最多 16 个字符。
服务器名	服务器的名称。名称对工程 and 任何包括的工程而言必须是唯一的，而且不得包含空格。最多 16 个字符。
网络地址	要配置的服务器的网络地址。要指定到服务器的双向网络连接，请使用逗号分隔列表。请参见 网络冗余 。最多 70 个字符。
端口	此服务器将侦听的端口。可以将其保留为空，在这种情况下将使用缺省端口号。最多 16 个字符。
对等端口	对等端口用于 I/O 服务器之间的通讯，为辅写提供 I/O 服务器信息更新。它还用作传统端口，允许 v7 之前的 Vijeo Citect 客户机仍从 v7 I/O 服务器检索 I/O 数据。缺省参数值为 2078。如果为此端口指定其它参数值，则 v7 I/O 服务器仍正确通讯，但传统 Vijeo Citect 客户机将不能再检索数据。最多 16 个字符。
注释	任何有用的注释。此属性是可选的，不会在运行时使用。最多 48 个字符。

- 单击**添加**按钮附加新记录或单击**替换**（如果已对记录进行了修改）。

在运行时为集群分配标签

Vijeo Citect 包括用于在运行时为特定集群分配页面上的标签的功能。

如果页面上的标签存在于许多不同的集群中，可以使用 **Cicode** 将群集名称传递到页面（如果该页面打开）。然后，在启动页面时会将没有显式定义集群的所有标签分配给指定的集群。

这将为复制的系统提供一个实际解决方案，在该解决方案中您的工程控制大量相同的地点或生产线。由于各个地点包含相同的设备，因此表示硬件架构的标签对于每个地点可能是相同的。在运行时将集群名称传递到页面的功能意味着，需要一个模拟页面来表示多个地点，从而缩短配置时间并简化工程部署。

例如，可以将一个菜单页面，以针对这三条相同的生产线启动模拟页面，每个条生产线基于称为“**ProductionLine**”的同一页面。

要实现此目标，可以使用 **PageDisplay** 函数在该菜单页上配置以下按钮：

按钮 1	文本	生产线 A
	命令	<code>PageDisplay("ProductionLine","Cluster_A")</code>
	注释	在生产线 A 的上下文中显示模拟页面
按钮 2	文本	生产线 B
	命令	<code>PageDisplay("ProductionLine","Cluster_B")</code>
	注释	在生产线 B 的上下文中显示模拟页面
按钮 3	文本	生产线 C
	命令	<code>PageDisplay("ProductionLine","Cluster_C")</code>
	注释	在生产线 C 的上下文中显示模拟页面

在每种情况下，**PageDisplay** 都会根据所选的按钮将主机集群的名称传递到该页面。由于每条生产线共享同一架构，因此打开页面时会将没有显式定义集群的所有标签分配给指定的集群。

此功能仅受以下 **Cicode** 函数支持：

- **PageDisplay**
- **PageGoto**
- **WinNew**
- **WinNewAt**

还可以使用 **PageInfo** 函数来确定已为页面设置的集群上下文。

此功能将扩展到可从图形页面调用的任何 Cicode。可以编写仅指定变量标签的 Cicode，以允许通过从中启动群集的页面的当前上下文来定义该集群。

如果 Cicode 由函数 TaskNew 执行，可以选择通过设置 ClusterName 参数来指定集群。

注意：代码运行后，不能更改 Cicode 的集群上下文。

Chapter: 14 在系统中建立冗余

可以在多个不同级别定义 Vijeo Citect 中的冗余。为系统建立冗余时，确定适合需要的保护程度非常重要，可从以下几方面考虑：

- 定义过程的重要程度；
- 确定系统停机的总体可能性，以及将花费最长时间恢复服务的重要过程和设备；
- 确定要为哪些部分实施冗余；以及
- 考虑冗余的设计和维持后果。

该部分包含以下冗余概念：

- [I/O 服务器冗余](#)
- [I/O 设备提升](#)
- [冗余和持续](#)
- [数据路径冗余](#)
- [网络冗余](#)
- [报警、报表和趋势服务器冗余](#)

注意：可使用计算机设置向导通过定义每台计算机的功能来定义所需的冗余级别（参见 [运行计算机设置向导](#)）。

另请参见

[报警、报表和趋势服务器冗余](#)

[如何Vijeo Citect处理文件服务器冗余](#)

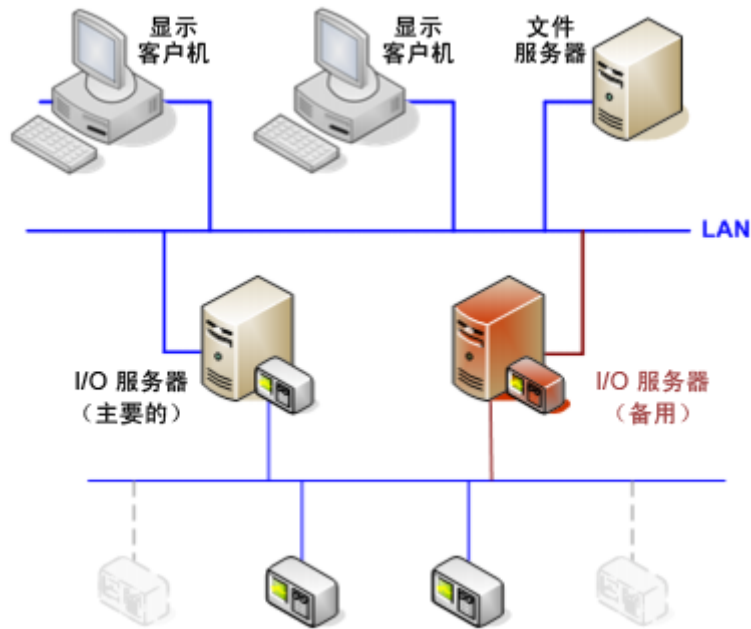
[如何Vijeo Citect处理 FTP 服务器冗余](#)

[单机系统的冗余](#)

I/O 服务器冗余

仅有一台 I/O 服务器的系统可能会因为唯一性设备或过程的故障而中断。如果主要服务器工作不正常，将失去对系统的控制和监控。通过引入辅助 I/O 服务器并将它专用于与相同的 I/O 设备进行通讯，则唯一性设备对整个系统的影响将被最小化。系统中现已包含 *主要* 和 *备用* I/O 服务器，其中备用 I/O 服务器会在主要 I/O 服务器不正常时工作。

下图展示了将备用 I/O 服务器引入到现有系统。当系统处于运行状态时，两个 I/O 服务器需要同等维护。



注意：尽管两个 I/O 服务器是相同的，但还是要注意备用服务器与主要服务器的功能不是重复的。如果功能重复，则对网络 PLC 部分的负荷会加倍，并且性能也会大大下降。因此，在给定时间内，只有主要服务器与 PLC 通讯。

冗余按以下方式提供：

- 当系统处于运行状态且主要 I/O 服务器不正常时，或者需要离线执行某些维护，所有客户机将转向备用服务器，从而不用中断系统，或使中断时间最小化。
- 主要服务器恢复在线状态时，系统将 I/O 设备的控制归还给主要服务器。控制的归还通过将磁盘镜像从备用服务器复制到主要服务器来完成，这样客户机就可以重新连接到主要服务器并继续对系统进行控制。

当系统运行时，用户还可以使用冗余 I/O 服务器分摊处理负荷。这样性能会更高，因为所有 I/O 服务器会在为 I/O 设备服务时并行运行。

另请参见
[I/O 设备提升](#)

I/O 设备提升

I/O 设备提升是指当系统事件强制备用 I/O 设备在运行期认定主要角色。

如果集群内有多个备用设备可用，则设备的提升顺序取决于工程的配置方法。

如果需要，可以通过扩展的 [I/O 设备属性](#) 表中的 **优先级** 字段手动设置备用设备的提升顺序。如果未设置优先级，则编译器会根据以下规则为每个备用设备自动分配优先级参数值：

- 用户配置的优先级设置享有优先权
- 没有进行优先级设置的任何备用设备将按配置顺序分配优先级
- 如果备用设备都没有进行优先级设置，则根据配置顺序为其分配优先级。

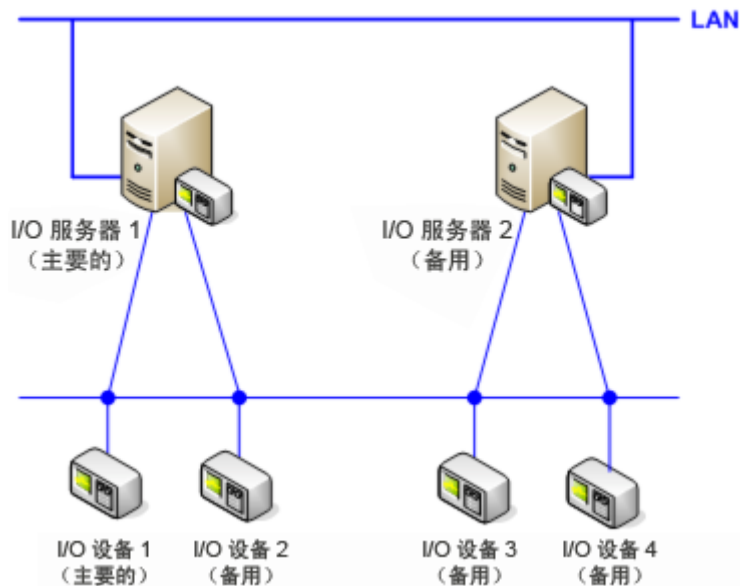
不论 I/O 设备与 I/O 服务器之间的连接在集群中的配置如何，这些规则都会应用。

备用设备的优先级在工程编译期间进行确认和/或分配。系统会为编译器发出的每个分配提供通知。在以下情况下会出现编译错误：

- 如果主要设备的优先级设置不是缺省参数值 1
- 如果备用设备的优先级参数值重复

范例

下图显示连接到两个 I/O 服务器的四个 I/O 设备，其中只配置了一个主要设备。



如果没有为备用设备设置优先级，编译器会进行以下分配：

- 缺省情况下为 I/O Device1 分配**优先级 1**（无编译器提醒）
- 为 I/O Device2 分配**优先级 2**（编译器警告提醒）
- 为 I/O Device3 分配**优先级 3**（编译器警告提醒）
- 为 I/O Device4 分配**优先级 4**（编译器警告提醒）

这样就假定设备按数字顺序配置。

如果已将 I/O Device3 手动设置为优先级 2，则编译器会进行以下分配：

- 缺省情况下为 I/O Device1 分配**优先级 1**（无编译器提醒）
- 为 I/O Device2 分配**优先级 3**（生成编译器提醒）
- 为 I/O Device3 分配**优先级 2**（不生成提醒）
- 为 I/O Device4 分配**优先级 4**（生成编译器提醒）

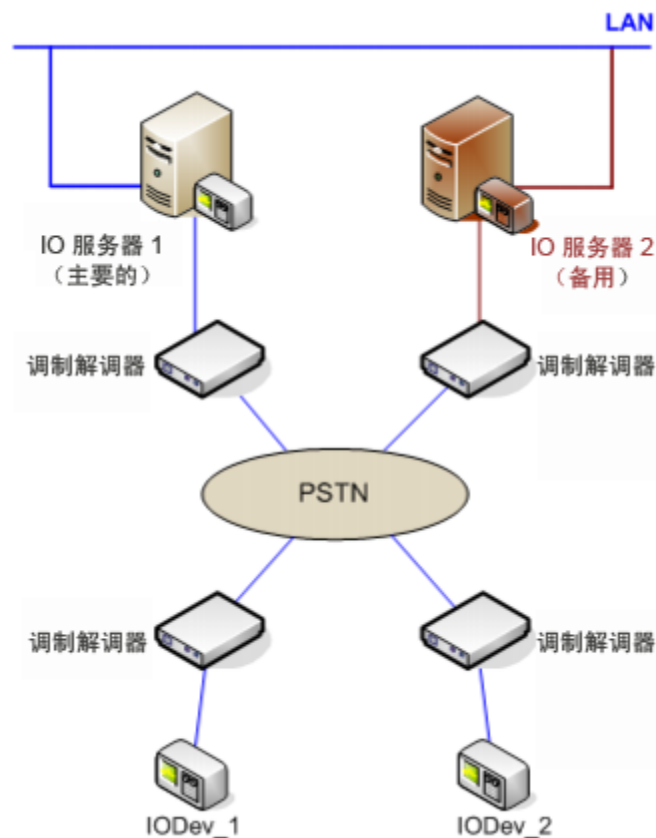
在这种情况下，I/O Device3 的设置享有优先权。其余备用设备根据配置顺序进行提升设置。

注意：自动设置的优先权使用 $n+1$ 公式，如果已将 I/O Device3 设置为优先级 5，则为其余设备分配优先级 6 和 7。

另请参见
[冗余和持续](#)
[数据路径冗余](#)

冗余和持续

如果您使用的是服务器冗余，则持续缓冲区（I/O 服务器缓存）将使用最新读取到的设备数据对备用服务器进行不断地更新。持续缓存或 I/O 服务器缓存创建用于每个缓存 I/O 设备。下图介绍了持续缓存的概念。



图中显示了两个 I/O 服务器，即 IO Server1（主）和 IO Server2（备用）。每个服务器都通过调制解调器连接到公共交换电话网 (PSTN)，而 PSTN 也通过调制解调器连接到 I/O 设备。持续缓存的工作方式如下：

1. 对于每个 IODevices->Cache Time period, 所有来自 I/O 设备的数据都临时存储在 I/O 服务器的内存中 (I/O 服务器缓存)。
2. 对于每个 [IOServer]SavePeriod, IOserver1 将其缓存中的内容保存到磁盘。
3. 缓存保存在持续缓存中, 每一个缓存设备都分配有一个这样的缓存。
4. IOserver1 将持续缓存的 UNC 路径 (由 [IOServer]SaveNetwork 设定) 告知其他所有 I/O 服务器。
5. 通过这些持续缓存, IOserver2 会不断更新其内存缓冲中的 I/O 设备信息。
6. 根据 "[IOServer]SavePeriod" (确定每隔多少秒将持续缓存保存在硬盘中) 的 I/O 服务器参数值, IOserver1 每隔 x 秒就将其内存缓冲保存到硬盘中。

注意: 您可以使用快速通讯向导, 或者通过在 Vijeo Citect 工程编辑器的 I/O 设备窗体中添加一个新设备来定义某个 I/O 服务器上的 I/O 设备。

由于向所有 I/O 服务器告知了在 [IOServer]SaveNetwork 中设置的 UNC 路径名, 因此, 您可以使用多台备用服务器。通过持续缓存, 每个 I/O 服务器将只为在该服务器上定义的 I/O 设备更新其缓存。因此, 用户可以设定多台 I/O 服务器, 并始终用读取的最新数据来更新它们的缓存。

例如, 我们按照如下方式设置了 [IOServer]SaveFile 和 [IOServer]SaveNetwork 参数:

在 IOserver1 上	在 IOserver2 上
[IOServer]	[IOServer]
SaveFile=C:\Data\IOserver1.dat	SaveFile=C:\Data\IOserver2.dat
SaveNetwork=\\IOserver1\Data\IOserver1.dat	SaveNetwork=\\IOserver2\Data\IOserver2.dat

IOserver1 将向其他所有 I/O 服务器告知持续缓存的以下 UNC 路径:

'\\IOserver1\Data\IOserver1.dat'。IOserver2 将使用持续缓存, 用 IOserver1 读得的最新设备数据来更新其内部缓存。

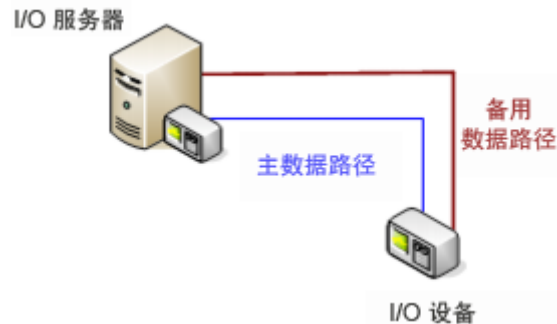
另请参见

[数据路径冗余](#)

数据路径冗余

数据路径冗余是涉及在 I/O 服务器和连接的 I/O 设备之间定义数据路径的另一种冗余形式。通过提供一个辅助 (并行的) 数据路径, 可以有效确保如果一个指向 I/O 设备的数据路径出现断开连接, 可以使用另一个路径。

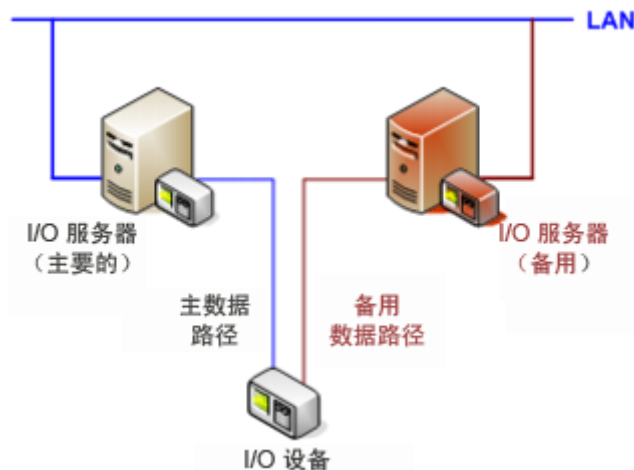
大多数 PLC 品牌都具有可用于在 I/O 服务器和 I/O 设备之间安装平行数据路径的工具。



上图显示已定义了一个额外的数据路径（并行运行）。冗余按以下方式提供：

- 当启动运行系统时，Vijeo Citect 使用**主数据路径**连接到 I/O 设备。
- 一旦与 I/O 设备之间的通讯丢失（例如通讯电缆被切断），Vijeo Citect 将在不中断系统运行的情况下**切换到备用数据路径**。
- 当主数据路径恢复工作时，Vijeo Citect 再次通过该路径连接。

对于大型系统（如联网运行的系统），还可以使用数据路径冗余保持设备与多重 I/O 服务器冗余通讯，如下图所示。



冗余按以下方式提供：

- 通过使用来自 I/O 设备的冗余数据路径（一个路径对应一个 I/O 服务器），可以保持 I/O 设备的通讯。
- 即使与主 I/O 服务器或备用 I/O 服务器间的通讯断开连接，仍可以访问 I/O 设备。

另请参见

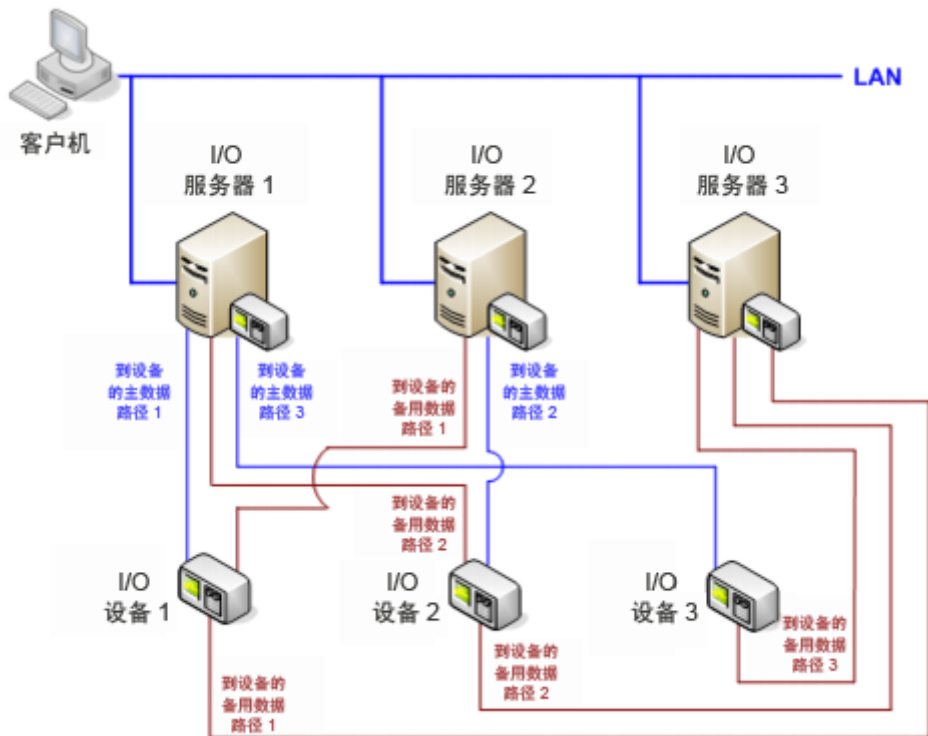
[多重设备冗余（备用数据路径）](#)

多重设备冗余（备用数据路径）

如果 I/O 设备支持对等通讯，则可以通过复制 I/O 设备来添加其它冗余级别。

注意： 尽管 I/O 服务器未被指定主要或备用角色（基于它们所连接到的 I/O 设备），在冗余 I/O 系统中的共同做法是将主要 I/O 设备连接至主要 I/O 服务器，而将备用 I/O 设备连接至备用 I/O 服务器。一个 I/O 服务器可以与主要 I/O 设备连接，也可以与备用 I/O 设备连接。I/O 服务器可以支持任何数量的备用数据路径。

下图说明了多重设备冗余和备用数据路径：



在此方案中，有三个 I/O 服务器通过以下方式与三个 I/O 设备连接：

I/O 服务器	连接的 I/O 设备
IOServer1	I/O Device1（主要） I/O Device2（备用） I/O Device3（主要）
IOServer2	I/O Device1（备用） I/O Device2（主要）
IOServer3	I/O Device1（备用） I/O Device2（备用） I/O Device3（备用）

以下内容已知:

- Vijeo Citect 客户机同时与所有已配置的 I/O 服务器通讯 (启动时, 客户机尝试连接到每个已配置的 I/O 服务器。如果无法建立与 I/O 服务器的通讯, 将产生硬件错误)。
- 当所有设备都运行时, Vijeo Citect 会处理主要 I/O 设备中的 I/O, 这样会减少 I/O 设备和 PLC 网络的 I/O 负荷 (这对获取改进性能很重要)。
- 客户机会对三个 I/O 服务器创建网络进程。
- 然后, 客户机会将 I/O Device1 和 I/O Device3 的请求发送给 I/O Server1, 将 I/O Device2 的请求发送给 I/O Server2。

冗余按以下方式提供:

- 如果 I/O Device1 在 I/O Server1 中出现故障, 客户机会通过备用数据路径将 I/O Device1 的请求发送给 I/O Server2。然后继续将 I/O Device3 的请求发送给 I/O Server1。
- 如果 I/O Device2 也在 I/O Server2 中出现故障, 客户机会将请求发送给 I/O Server3。
- 如果重新建立 I/O Device1 与 I/O Server1 之间的连接, 客户机会将 I/O Device1 的请求继续发送给 I/O Server1。

由于用户可以在不同的 I/O 服务器上存放主要 I/O 设备和备用 I/O 设备, 因此应在两个 I/O 服务器之间共享主要 I/O 设备, 从而平衡所有 I/O 服务器的负荷。但这种方式可能并非适用于所有协议, 因为负荷可能取决于 PLC 网络, 而不是 I/O 服务器 CPU。在这种情况下, 同一个 PLC 网络中的多个激活 I/O 服务器可能会使 PLC 网络运行速度降低, 因此会使整体响应速度减慢。

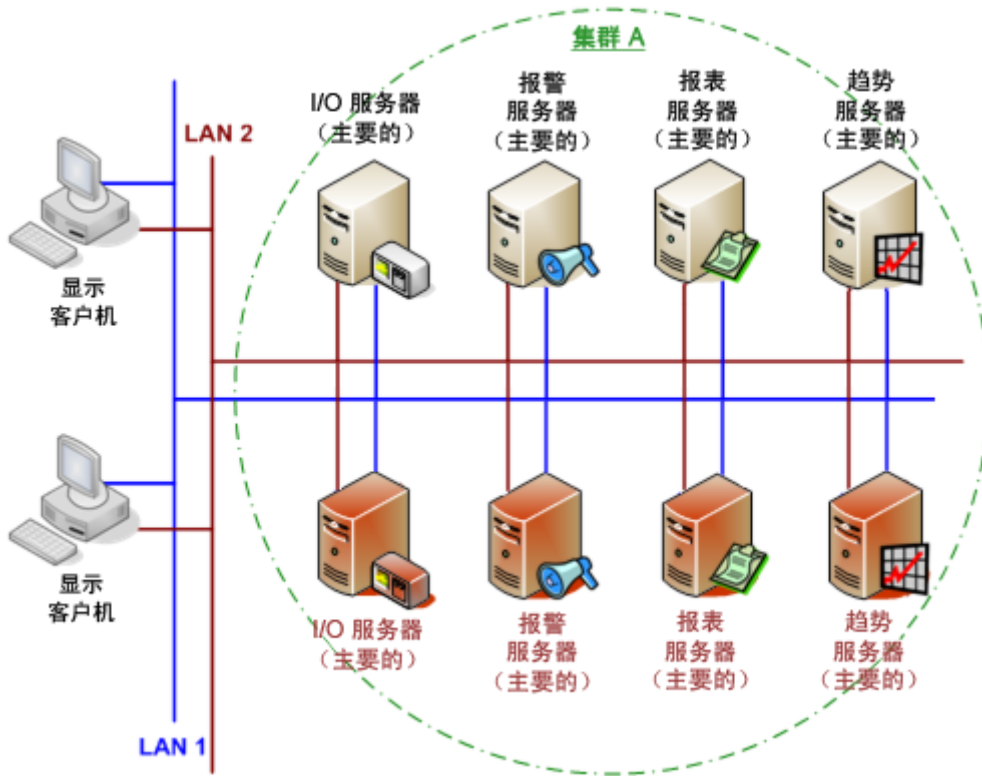
另请参见

[报警、报表和趋势服务器冗余](#)

网络冗余

通过使用每台客户机或服务器的双重 NIC (或多重网络接口) 功能, 用户可以指定从客户机到服务器完整且唯一的网络连接。

参看下图:



上例显示两个级别的冗余：

- 网络冗余
- 服务器冗余

网络冗余

集群中的每个系统组件都连接到第二个网络 (LAN 2)。连接操作通过使用每台服务器的双端点实现。此操作为连接提供两个单独的 LAN，继而提供以下 LAN 冗余：

- 如果 LAN 1 突然出现故障，则集群中的每个组件可以通过使用 LAN 2 轻松保持连接。
- 反之，如果 LAN 2 出现故障，则 LAN 1 保持运行状态。

服务器冗余

有关服务器冗余的信息，请参见 [报警、报表和趋势服务器冗余](#)。

另请参见
[配置网络冗余](#)

配置网络冗余

要使用双重冗余网络连接来连接到计算机，您需要首先定义每个网络接口的网络地址，然后指定每个服务器要使用的网络地址。

如何配置网络冗余：

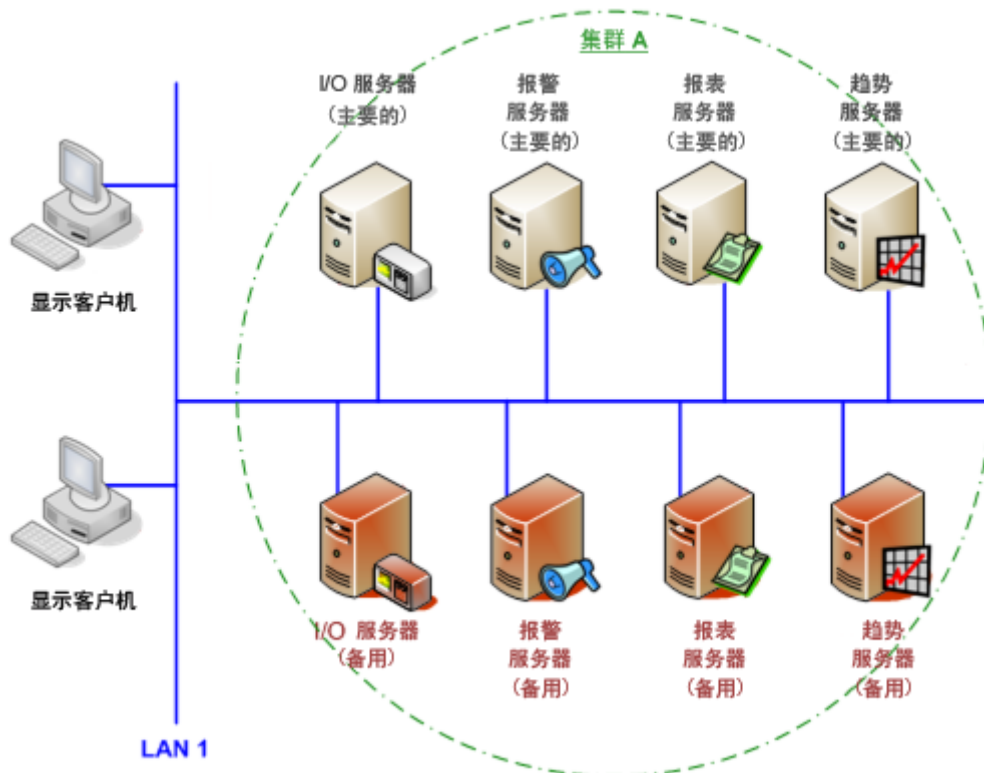
1. 在“工程编辑器”中，选择**服务器 | 网络地址**。
2. 在“网络地址”对话框中，定义每个网络接口卡的网络地址。请参见 [网络地址定义](#)。
3. 在“工程编辑器”中，选择**服务器**，然后选择要连接到网络的服务器的类型。
4. 在服务器配置对话框的“网络地址”域中，键入用逗号分隔的服务器的双重地址。例如，“AlarmPrimaryLAN1,AlarmPrimaryLAN2”。

另请参见
[网络冗余](#)

报警、报表和趋势服务器冗余

通过在已定义的集群中添加备用服务器（提供冗余的系统组件）来实现报警、报表、趋势和 I/O 服务器的冗余。此外，还可以利用每个组件的双端点（或多个网络接口）功能，以有效指定客户机到服务器的完整且唯一的网络连接。请参见 [网络冗余](#)。

考虑下图：



通过在同一个网络 (LAN 1) 上提供对应的备用服务器来实现图中每个组件的服务器冗余。冗余按以下方式提供：

- 如果任何服务器出现故障或通讯失败，则其备用服务器将立即执行操作。

- 对于 I/O 服务器冗余，当有故障的 I/O 服务器返回在线状态时，将恢复控制。
- 对于报警、报表和趋势服务器冗余，当有故障的主服务器返回在线状态时，客户机始终停留在备用服务器上（除非备用报表服务器在恢复到原始主服务器时出现故障）。

注意：由于 Vijeo Citect 在备用服务器和主服务器上保存的数据是完全相同的，所以客户机是从主服务器还是从备用服务器获取数据并不重要。而且经常会出现有些客户机与主服务器进行通讯，有些客户端与备用服务器进行通讯的情况。

另请参见

[报警服务器冗余](#)

[报表服务器冗余](#)

[趋势服务器冗余](#)

[文件服务器冗余](#)

[FTP 服务器冗余](#)

报警服务器冗余

可以在一个工程中配置两个报警服务器。即一个主报警服务器和一个备用报警服务器。通过两个报警服务器，您可以在自己的系统中实现改进（镜像）冗余。

当两个报警服务器同时运行时，报警在两个服务器中并行处理，并由主报警服务器记录。如果主报警服务器发生故障，备用报警服务器开始将报警记录到设备上。

当一个报警服务器首先启动时，它会尝试与另一个报警服务器进行连接。如果无法连接，它会将动态报警数据从正在运行的报警服务器传送到另一个报警服务器（此数据包含汇总数据和当前报警状态）。如果无法与另一报警服务器建立连接，报警服务器将打开保存文件（用 [Alarm]SavePrimary 参数定义的文件）并且从该文件中恢复数据。如果存在两个保存文件，一个来自主报警服务器，另一个来自备用报警服务器，Vijeo Citect 会使用日期靠后的保存文件（换言之，最新的保存文件）。如果未配置保存文件，报警服务器将无法获得报警的初始状态，并且没有可用的汇总信息。如果是这种情况，报警服务器将开始处理报警，然后确认所有新报警。

当两个报警服务器都处于活动状态时，它们都会从 I/O 服务器中读取数据并对报警进行处理。每一报警的开/关状态并没有在这两台服务器之间传递。当操作员对报警执行函数时（例如确认、禁止、启用、添加注释等），此信息将在两台报警服务器之间传递（如果操作员在一台服务器上确认了某报警，该服务器会告知另一台服务器确认同一报警）。

Vijeo Citect 客户端既可以连接到主报警服务器，又可以连接到备用报警服务器。启动时，所有客户机都将尝试同主报警服务器进行连接。如果无法与主报警服务器建立连接，它们会尝试与备用报警服务器建立连接。如果主报警服务器恢复活动并变为可用，连接到备用报警服务器的所有客户机都将保持与备用报警服务器的连接（这对于客户机可以与其报警服务器通信的 Vijeo Citect 并不重要，因为二者都包含相同的（镜像）数据）。

另请参见：

[报表服务器冗余](#)

报表服务器冗余

可以在一个工程中配置两台报表服务器。即一台主报表服务器和一台备用报表服务器。

当两个报表服务器同时运行时，预定的报表只在主报表服务器上运行。如果主报表服务器发生故障，预定的报表则会运行在备用报表服务器上（也可以将备用报表服务器配置为预定报表也将在主报表服务器上并行运行）。注意，在主报表服务器和备用报表服务器之间没有报表数据传输（Vijeo Citect 不会同步报表数据，因为报表可以将数据写入任意类型的设备）。

客户机既可以连接到主报表服务器，也可以连接到备用报表服务器。在启动时，所有客户机都会尝试连接主报表服务器。如果未能建立与主报表服务器的连接，然后会尝试建立与备用报表服务器的连接。若主报表服务器恢复运行，所有连接到备用报表服务器的客户机仍将保持与备用报表服务器的连接。

另请参见

[趋势服务器冗余](#)

趋势服务器冗余

可以在一个工程中配置两台趋势服务器。即一台主趋势服务器和一台备用趋势服务器。

当两台趋势服务器都在运行时，会并行处理两台服务器上的趋势，并写入磁盘（每台服务器必须写入它自己的磁盘或它在文件服务器上的私有区域）。

当一台趋势服务器启动时，它将尝试建立与另一台趋势服务器的连接。如果能建立连接，它将传输从它上次关机到当前时间段内的所有趋势数据（这样可以使趋势数据丢失的可能性最小）。

客户机既可以连接到主趋势服务器，也可以连接到备用趋势服务器。在启动时，所有客户机都会尝试建立与主趋势服务器的连接。如果未能建立与主趋势服务器的连接，客户机会尝试连接到备用趋势服务器。如果主趋势服务器恢复活动并变为可用，当前连接到备用趋势服务器的所有客户机都将保持与备用趋势服务器的连接（这对于客户机可以与其报警服务器通信的 Vijeo Citect 并不重要，因为二者都包含相同的（镜像）数据）。

另请参见

[文件服务器冗余](#)

文件服务器冗余

允许文件服务器冗余。[CtEdit]Backup 参数指定备份工程路径。如果 Vijeo Citect 不能在“运行”目录中找到文件（即，由 [CtEdit]Run 参数指定的），它将在备份路径中查找。如果在备份路径中找到文件，Vijeo Citect 将假定运行路径已经由于某些原因而不可用（例如，文件服务器不正常）。在进行转换之前，它将在备份中查找所有相关文件。当 Vijeo Citect 转换到备份路径时，将会调用事件号 11 并生成硬件错误：**文件服务器故障，转到备用。**

只有在转向器（或命令解释程序）能够在文件服务器不正常的情况下做出相应的响应时，文件服务器冗余才能正常运行。文件服务器不正常时，Novell Netware 的命令解释程序无法做到这一点，并且会使 Windows 本身变得不正常或将它报告为严重的网络错误。基于 Microsoft LAN 管理器的网络以及对等网络都能够在文件服务器不正常时正确地处理实例。因此，Vijeo Citect 文件服务器冗余在这些网络中将会正常运行。

注意：仅 Vijeo Citect 切换到备份路径。文件服务器发生不正常时，所有其他使用该服务器上文件的应用程序也会随之不正常。这可能会导致计算机长期处于等待文件服务器恢复的状态（或不正常）。Windows 本身也会发生这种情况，因此应把 Windows 安装在本地驱动器上。

为了启用文件服务器冗余，用户应将 [CTEDIT]Backup 参数设置为备份数据库路径。例如，如果主路径是 F:\CITECTUSER\DB，可将备份路径设置到另一台文件服务器或本地驱动器上，如 C:\CITECTUSER\DB。

应当确认备份路径中的工程与运行目录中的工程一致——在每次编译运行路径中的工程时用户应将其复制到备份路径中。



警告

不希望的设备操作

在将 Vijeo Citect 系统投入使用之前，确认备用文件服务器上已经具有与当前工程完全一样的副本，并且 [CTEDIT]Backup 参数已正确设置。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

另请参见

[FTP 服务器冗余](#)

FTP服务器冗余

支持 FTP 服务器冗余。如果主 FTP 服务器关闭，Vijeo Citect 会尝试连接到备用计算机上的 FTP 服务器。这种冗余和 I/O 服务器冗余是相互独立的，因此两台 FTP 服务器必须有相同的密码和相同的目录结构。

FTP 服务器冗余是通过设置主 FTP 服务器中 Citect.ini 文件的 [CLIENT] 和 [DNS] 部分的参数来配置的。如果主 FTP 服务器发生故障，在 [INTERNET]Redundancy 参数未设置为 0 的情况下，这些参数可以通过互联网显示客户机 (IDC) 下载到本地 Citect.ini 文件中。IDC 将使用下载的冗余信息连接备用 FTP 服务器。

注意：备用 FTP 服务器不必是互联网服务器。它可以是 IDC 能够连接的任意使用 TCP/IP 协议的服务器，只要网络中存在 IDC 许可证。

另请参见
[单机系统的冗余](#)

单机系统的冗余

如果您将 Vijeo Citect 作为单机系统使用，则在单个处理模式中，您仍可以通过在其他 CPU 上实现备用系统来达到冗余，但条件是您拥有多个 CPU 系统。您也可能希望将它用于负载平衡。

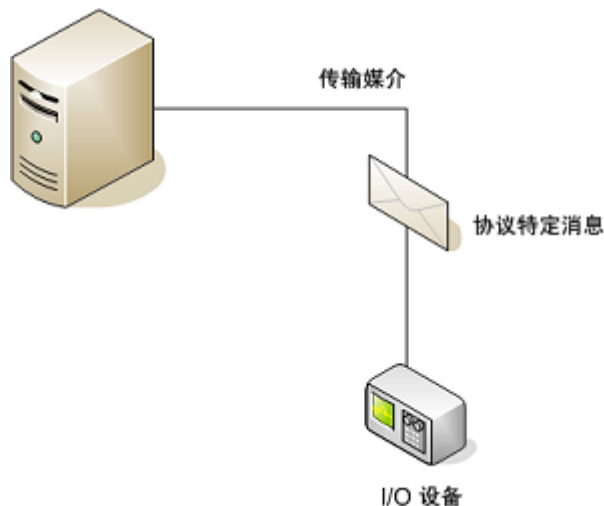
另请参见
[配置您的系统](#)

Chapter: 15 I/O 设备通讯

Vijeo Citect 能与具有通讯端口或数据交换接口的任何监控 I/O 设备通讯，包括 PLC（可编程逻辑控制器）、循环控制器、条形码读取器、科学分析仪、远程终端设备 (RTU) 和分布式控制系统 (DCS)。

典型的 Vijeo Citect 通讯由四个基本部分组成：

1. Vijeo Citect I/O 服务器；
2. 目标 I/O 设备；
3. 用于在二者之间传输消息的物理方法（传输）；
4. 在二者之间交换的消息（协议）。



这些组件共同将 I/O 设备的输入和输出展示给 Vijeo Citect 系统。

- I/O 设备的输入提供有关工厂的信息，如机器速度、传送带状态或炉子温度。
- I/O 设备的输出通常启动控制工厂操作的任务，如启动电机、改变速度或开关阀门和指示灯。

无需使用如图所示的直接电缆作为传输媒介；可使用任何消息传送方法 – 如高速无线链接或 FDDI 网络。同样，特定于协议的消息可以是简单的 ASCII 消息或基于对象的复杂消息（如 DNP 3）。需要特别注意的是，工程师可以自由组合 I/O 设备、传输和协议。

另请参见

[I/O 服务器的角色](#)

I/O 服务器的角色

直接连接到 I/O 设备的 Vijeo Citect 计算机即 I/O 服务器。I/O 服务器负责处理来自客户端的读取、写入和订阅请求。工程可以包含很多 I/O 服务器，每个集群也可以包含多个 I/O 服务器。

通过从每个 I/O 设备检索数据并将其存储在缓存（I/O 设备数据缓存）中，I/O 服务器可保留关于连接的所有 I/O 设备的最新信息。Vijeo Citect 客户端需要来自 I/O 设备的数据时，即会使用 I/O 服务器缓存中存储的信息。客户端不会从 I/O 设备直接检索数据。

I/O 服务器会从 I/O 设备读取所有所需的数据来完成请求的 Cicode 任务或进程。例如，调度报表时，Vijeo Citect 会在报表的第一行开始运行之前读取报表可能需要的所有 I/O 设备数据。

Vijeo Citect 异步执行对 I/O 设备的写入，这样在写入时仍可以继续其它操作。

另请参见

[I/O 设备的角色](#)

I/O 设备的角色

Vijeo Citect 是性能卓越的管理系统。它是直接监控和控制自动化设备的 I/O 设备。在大多数 I/O 设备（如 PLC）中，存储在其中的程序可控制输出。此存储程序的逻辑（控制策略）和输入的状态可确定每个输出的参数值。

每个输入和输出的参数值存储在 I/O 设备的单独内存寄存器中。每个内存寄存器通过其地址引用。

通过读写 I/O 设备中的内存寄存器，Vijeo Citect 会收集工厂的数据进行监控和分析，并提供设备和过程的高层（管理）控制。

通常不需要读取（或写入）I/O 设备的所有寄存器：用户可以通过 Vijeo Citect 指定要监视或控制的输入和输出。定义这些寄存器地址后，可以将其用于系统控制、操作显示、趋势分析、数据记录和报警指示。

I/O 设备的选择通常不对与系统通讯的个人工程开放。在多数情况下，I/O 设备硬件在集成之前购买，或者它本身就是传统设备。但是，如果用户对 I/O 设备的选择影响很大，则通讯能力应作为其中一个因素考虑。

注意：诸如可编程逻辑控制器 (PLC) 这样的 I/O 设备通常会提供控制工厂内低级进程的內部程序。PLC 程序不断扫描 PLC 的输入寄存器，并将输出寄存器设置为 PLC 程序逻辑确定的参数值。由于 Vijeo Citect 可以替换任何 PLC 程序，因此不推荐使用此操作。PLC 是为高速响应（通常为 1 到 100 毫秒）设计的，将此功能替换为 Vijeo Citect 会对控制系统的性能造成负面影响。只需使用 Vijeo Citect 即可辅助 PLC 程序，也就是说，可用于高级控制和系统监视。



不希望的设备操作

请勿将 Vijeo Citect 或其他 SCADA 软件用作基于 PLC 的控制程序的替代品。SCADA 软件并不是为直接、高速的系统控制而设计。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

另请参见

[传输媒介的角色](#)

传输媒介的角色

术语“传输”是指驱动它所需的物理通讯媒介和低级逻辑。至于 Vijeo Citect 所关注的，它只是简单用于定义如何将信息打包、如何发送信息以及将信息发送至何处。

可用传输选项通常由目标 I/O 设备可用端口的种类进行管理。但也受以下因素影响：自动化系统的地理位置、特定域要求（如安全、性能、冗余）以及所有者愿意投入的金额。

可用传输包括三个主要分类：

1. 简单串行 – 如 RS-232、RS-422、RS-485；
2. 以太网 – 基于支配框架的内网协议；或
3. 专用 – 使用中间计算机通讯硬件或软件（如 OPC 服务器）。

在这些分类中，以太网是迄今为止仍然选用的传输技术。

幸运的是，通过使用数据包封装和智能网关/路由设备的概念，可以混用不同的传输层。例如，可以使用从 Vijeo Citect 到调制解调器的串行电缆，从一个调制解调器到另一个调制解调器的 PSTN 线路，然后使用到以太网网关的串行电缆，以及到 I/O 设备的以太网。

注意：由于调制解调器被 Windows 识别为逻辑设备，因此被视为 Vijeo Citect 中的特殊情况。有关 [设置调制解调器](#) 的信息，请参阅特定联机帮助。

Vijeo Citect 使用“COMX”驱动程序实现简单串行传输。与之类似，“TCPIP”驱动程序可实现以太网传输。这些是最常用的传输驱动程序，两个驱动程序都包含大量可调整的选择和设置。每个专用板卡都有各自特定的传输驱动程序，此驱动程序通常与协议驱动程序集成。

另请参见

[协议的角色](#)

[使用设备驱动程序](#)

协议的角色

所有 I/O 设备至少支持一个协议，此协议用于管理命令的种类以及可以与设备交换的数据。协议的功能和复杂程度相差很大。但由于 Vijeo Citect 提供协议驱动程序，因此工程师无需了解协议的详细信息。

大多数现代设备支持两个或三个协议。这为工程师设计最适合的通讯架构提供很大的灵活性。有时设备中会有简单协议（提供对设备功能的有限访问）和较全面、较复杂的协议。偶尔协议会紧密链接到所用的传输层，尤其是专用协议和通讯硬件。

应选择的协议取决于：

- I/O 设备支持的协议；
- 需要访问的数据的种类；
- 需要发出的命令；以及
- 来自所选传输的任何限制。

在很多情况下，I/O 设备中需要安装特定卡件或模块来支持其它协议。

使用行业标准协议

自动化及相关行业已开发了大量标准化协议，用于与 I/O 设备通讯。由于这些协议是开放且协作开发的，因此他们承诺在连接性和寿命方面，这些协议比其它协议具有更高的地位。

以下列表显示某些性能最为卓越的通讯标准（这些标准在 Vijeo Citect 支持的自动化行业使用）：

- **ASCII** – 用于简单串行通讯；
- **Modbus** – 自动化中广泛使用的简单串行通讯；
- **DNP 3.0** – 分布式网络（如 RTU）的协议；
- **BACNet** – 专门用于建筑自动化控制行业；
- **OPC** – 在 PC 级别共享自动化数据的技术；
- **IEC870-5** – 用于发送基本遥控信息的通讯配置文件；
- **EIB** – 欧洲安装总线；
- **Profibus** – 自动化的地域总线通讯协议；
- **SNMP** – 网络设备广泛使用的协议。

由于 Vijeo Citect 允许用户选择地点所需的协议，因此可以在 I/O 设备支持的位置使用这些协议。

这在 I/O 设备品牌混杂但又希望简化和使用同一个协议时尤为有用。

另请参见

[通讯配置](#)

[使用设备驱动程序](#)

通讯配置

Vijeo Citect 使用逻辑性的结构化通讯配置，该配置将准确地映射到下图所示的元素。

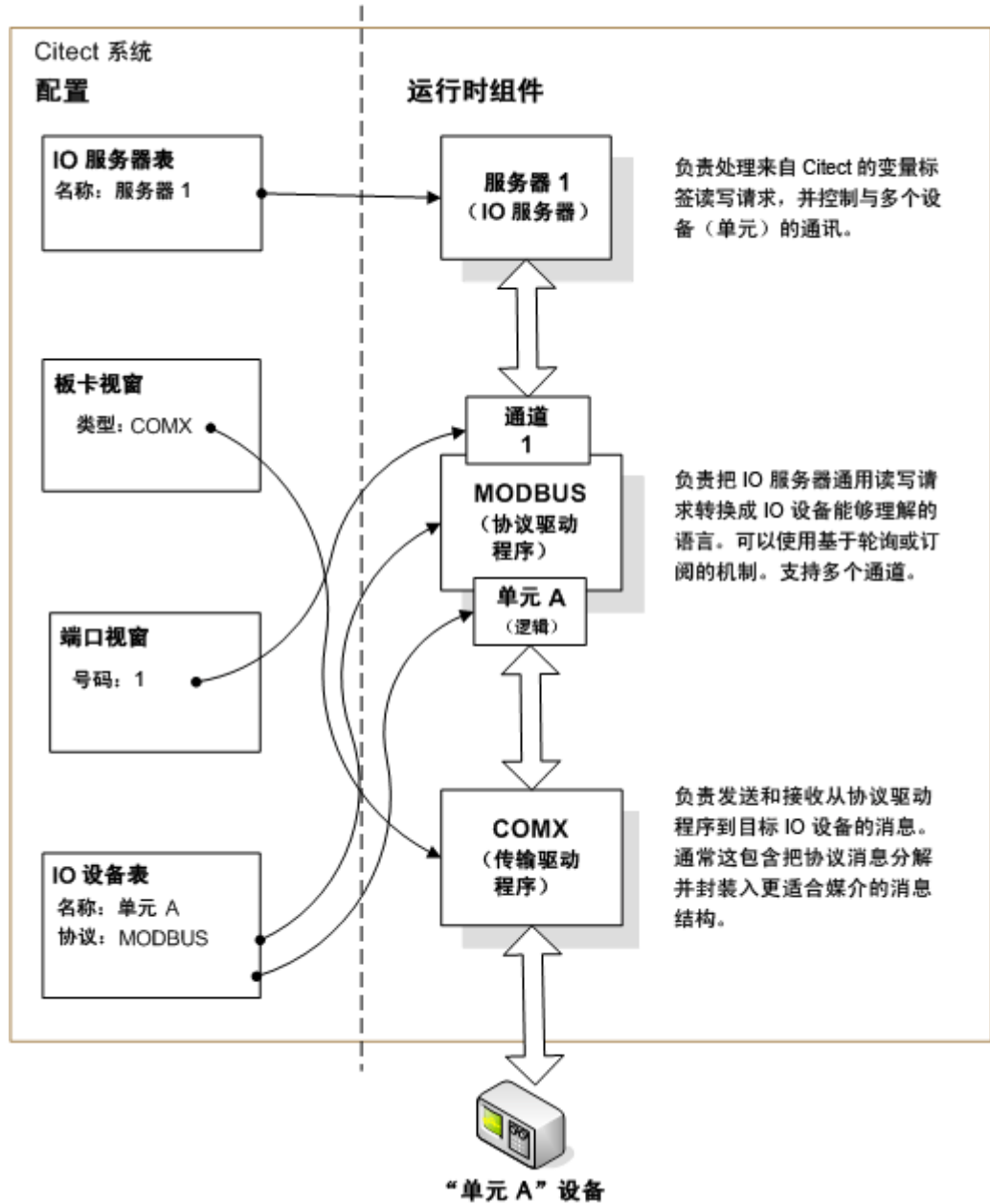
工程师需要执行以下操作：

1. 在 I/O 服务器表中定义一个 I/O 服务器；
2. 在板表中定义传输类型（COMx、TCPIP、PROFI...）；
3. 在端口表中定义要与外部通讯的端口。
4. 定义要使用的协议（MODBUS、DNP、ASCII...）。

注意：为说明从基本元素到基本通讯架构的映射，上述每一步骤都进行了简化。事实上，需要针对通讯设置执行一些其他任务。

下图显示配置环境中的元素如何映射到运行系统中的软件组件。

尽管这不需要重点掌握，但它可以帮助理解驱动程序如何与系统相适合。



注意: 为了清楚, 该范例忽略了许多逻辑运行组件 (如客户机)。此外, 用户在运行时看不到图示组件。但在运行时, 可以通过 Vijeo Citect Kernel 访问所有驱动程序。

设置简单通讯时, 以“快速 I/O 设备设置”向导作为起点是一个不错的选择。了解设置的工作方式后, 您可以手动配置更复杂的排列。可以使用驱动程序选项和参数修改协议和传输

的功能。在在线帮助中，每个驱动程序都有其自己的特定部分，可引导您完成所有配置选项。

设置通信

设置设备与 Vijeo Citect I/O 服务器之间的通信通常涉及包括下列步骤的过程：

1. 选择通信方法

通过检查 I/O 设备和控制系统网络的能力，您应该选择一个适当的通信机制。这种机制通常意味着必须在 Vijeo Citect 中配置的传输和协议驱动程序。

有关详细信息，请参见 [传输媒介的作用](#) 和 [协议的作用](#)。

2. 准备设备

确保您的设备满足与 Vijeo Citect 进行通信所需的硬件和软件要求。驱动程序参考帮助提供每个受支持设备特定的硬件设置信息。

3. 准备要通信的 I/O 服务器

这涉及准备要与一个设备进行通信的 I/O 服务器，包括在 I/O 服务器中设置 COM 端口、串行板卡或专用板卡。

4. 创建测试工程

配置 Vijeo Citect 工程之前，应确认 Vijeo Citect 与系统设备之间能够进行通信。通过创建测试工程，可以测试隔断的通信路径，确认 Vijeo Citect 能够让您的设备上线。

5. 使用通信快速向导配置通信

快速通信向导提供缺省值以及根据所选 I/O 设备的通信要求定制的设置。

或者

[手动配置通信](#)

可以使用 Vijeo Citect 工程编辑器中的板卡、端口和 I/O 设备表单配置到 I/O 设备的通信。手动配置通常灵活性更大，但需要具有更详尽的驱动程序设置和参数的知识。

在执行了这些步骤之后，便可以运行和验证通信。

另请参见

[使用 Citect.ini 参数自定义工程](#)

[使用磁盘 I/O 设备](#)

准备设备

您需要确认自己的系统符合在设备和 Vijeo Citect 之间建立通信所需的硬件和软件要求。

这可能包含多种可能性，包括：

- 专有通信卡的安装
- 设备服务器的建立
- 配置软件等软件的安装

驱动程序参考帮助提供了每种设备的硬件和软件要求，以及需要了解的有关设置特定设备的任何其它信息。要访问此帮助，请从 Citect 工程管理器和工程编辑器中的**帮助**菜单中选择**驱动程序帮助**。

另请参见

[准备要通信的 I/O 服务器](#)

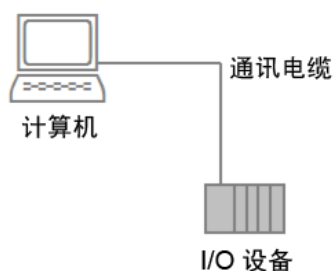
准备要通信的 I/O 服务器

为支持所需的通信方法，您需要确认 Vijeo Citect I/O 服务器已被启用。这包括安装和配置通信所需的任何通信硬件和/或软件。通用方案为：

- [使用 COM 端口](#)
- [使用串行板卡](#)
- [使用 Ethernet 卡](#)
- [使用专用板卡/中间软件](#)

使用 COM 端口

最简单的 Vijeo Citect 系统使用单台计算机连接到 I/O 设备。您可以将 I/O 设备直接连接到使用标准 RS-232 通讯电缆的通讯端口。



如何设置 Vijeo Citect 使用您的计算机 COM 端口：

1. 确认将**板卡**的**类型**配置为 **COMx**，将**地址**设置为 0。**I/O 端口**、**中断**和**特殊选项**可以保留为空。
2. 在**端口**配置中输入**端口号**。COM 端口号通常是 1 或 2，在控制面板的端口段中设置。使用**特殊选项**字段修改 COMx 驱动程序的行为。（有关详细信息，请参见 [COMx 驱动程序专用选项参考](#)）。

注意：您只需要定义 COMx 板卡一次。然后增加几个使用相同 Vijeo Citect 板卡的端口。例如，一个 COM 端口和两个串行板卡会在 Vijeo Citect 中定义为一个带有多个端口的 COMx 板卡。

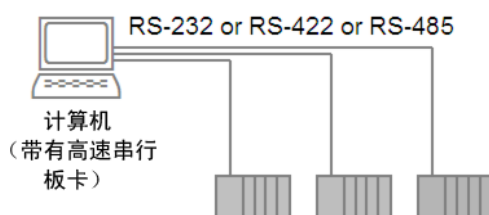
另请参见

[调试一个 COMx 驱动程序](#)

使用串行板卡

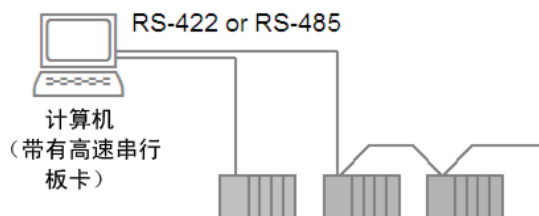
使用串行板卡

计算机中的通讯端口不是为高速通讯设计的，因此会降低系统性能。但计算机中安装了高速串行板卡（如 Digiboard）。通过高速串行板卡上的几个端口（通常为 4 个、8 个或 16 个），可将多个 I/O 设备连接到 Vijeo Citect 系统。

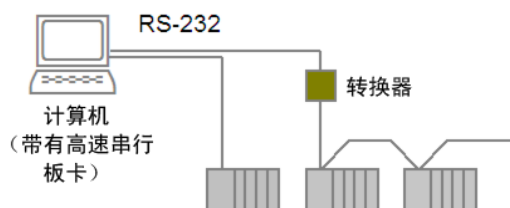


可以使用相同的 I/O 设备或不同制造商提供的 I/O 设备；Vijeo Citect 支持绝大部分常见 I/O 设备。可以连接任意数目的 I/O 设备；唯一的限制是计算机的大小。高速串行板卡可用于 RS-232、RS-422 或 RS-485 通讯。

如果几个 I/O 设备都来自同一制造商，并且这些 I/O 设备支持多点式通讯，则可以将其连接到计算机中安装的 RS-422 或 RS-485 高速串行板卡。（RS-232 标准不支持多点式通讯。）



并非所有的高速串行板卡都支持 RS-422。可以使用 RS-232/RS-422 或 RS-232/RS-485 转换器完成相同的目的。



注意： 使用转换器可能导致握手/定时注意事项。

如何设置 Vijeo Citect 以使用串行板卡：

1. 在计算机中安装板卡，然后按照附带的说明在 Windows 下对其进行设置。使用板卡制造商的最新驱动程序。
2. 确保板卡配置的类型为 COMx，并且地址设置为 0。I/O 端口、中断和特殊选项可以保留空白。
3. 在端口配置中输入端口号。COM 端口号通常大于 2，并在控制面板的端口部分中进行设置。使用特殊选项字段可修改 COMx 驱动程序的行为。（有关详细信息，请参见 [COMx 驱动程序专用选项参考](#)）。

注意

- 如果使用的是计算机的 COM 端口，则无需安装附加软件。
- 只需定义 COMx 板卡一次，然后添加几个使用相同 Vijeo Citect 板卡的端口。例如，一个 COM 端口和两个串行板卡会在 Vijeo Citect 中定义为一个带有多个端口的 COMx 板卡。

另请参见

[使用专用板卡](#)

COMX 驱动程序特殊选项参考

特殊选项（在端口表格中）是用空格分隔的选项，以破折号 (-) 开始，后面紧跟着选项字符。将只有一小部分用户需要使用以下选项：

- **-cATS0=1**: 启动时发送字符串 "ATS0=1<CR>" 允许调制解调器侦测端口的运行波特率。DCD 低时放弃传输。等待拨入呼叫拉高 DCD。
- **-d**: 数据仅在 DSR 高时传输。
- **-di**: 当 DSR 低时，接收的数据将被忽略。
- **-dMS**: 传输信息时，驱动程序将等待 2000 毫秒以便 DSR 拉高，然后在传输之前再等待 MS 毫秒。
- **-e**: 可由此访问输出信号线。格式为 "~EIAWXYZ"，其中 WXYZ 代表以下选项之一：

S	模拟接收的 XOFF
S	模拟接收的 XON
RT	将 RTS 设置为高
RT	将 RTS 设置为低
D	将 DTR 设置为高
D	将 DTR 设置为低
B	设置设备断行
B	清除设备断行

范例: "~EIADTR1"将 DTR 设置为高

- **-h:** 数据仅在 CTS 高时传输。
- **-hMS:** 传输信息时, 驱动程序将等待 2000 毫秒以便 CTS 拉高, 然后在传输之前再等待 MS 毫秒。
- **-i:** 每次初始化端口时都会发送字符串。代字号 (~) 和 "\M" 字符表示以下特殊说明:
 - ~: 500 毫秒延迟
 - \M: 发送回车

范例:

~Fred: 等待 500 毫秒, 然后发送 "Fred"

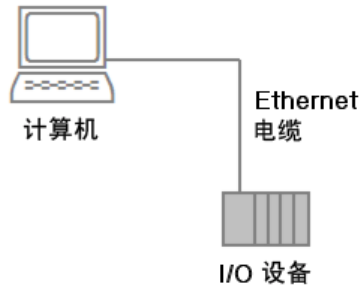
Fred\Mays: 先后发送 "Fred"、回车和 "Mary"

注意: 此选项不适用于可拨号的设备 (即端口号为 -1 时)。

- **-nt:** 对于某些串行接口, 线路故障可能会导致 COMx 读取线程关闭。如果发生此情况, 驱动程序不会在故障过后恢复。但对于 -nt (非停止) 选项设置, 线程不会关闭, 这使系统能在故障被校正后恢复。
- **-nts:** 如果 COMx 驱动程序正在启动时出错, 它不会停止, 将继续尝试打开 COMx 端口。
- **-r:** 仅在传输时, 驱动程序才拉高 DTR。
- **-ri:** DTR 在输入缓冲器中有足够的空间接收输入字符时被拉高, 在输入缓冲器中没有足够空间时被拉低。
- **-rPRE,POST:** 传输信息时, 驱动程序将拉高 DTR PRE 毫秒, 传输数据, 等待 POST 毫秒, 然后拉低 DTR。
- **-sc:** 使用 XON 和 XOFF 激活软件流控制
 - **XonLim:** 一个以字节为单位的数字。这表示发送 XON 字符之前达到的输入缓冲器级别 (30 字节)。
 - **XoffLim:** 发送 XOFF 字符前输入缓冲器接受的最大字节数。这通过用输入缓冲器的大小除以 (以字节为单位) 100 计算得出。
- **-t:** 仅在传输时, 驱动程序才拉高 RTS。
- **-ti:** RTS 在输入缓冲器中有足够的空间接收输入字符时被拉高, 在输入缓冲器中没有足够的空间时被拉低。
- **-to:** RTS 仅在要有要传输的字符时被拉高。
- **-tPRE,POST:** 传输信息时, 驱动程序将拉高 RTS PRE 毫秒, 传输消息, 等待 POST 毫秒, 然后拉低 RTS。

使用 Ethernet 卡

Ethernet 是高速与 I/O 设备通讯的最常见方法之一。其配置非常简单。通常, 您也可以使用标准 Ethernet 卡或集成 Ethernet 端口连接到使用 Ethernet 的 I/O 设备。



如何设置 Vijeo Citect 使用您的计算机 Ethernet 卡：

1. 确保您的 Ethernet 卡可以正确安装，并能够在 Windows 下正常工作。
2. 确保将板卡的类型配置为 TCP/IP，将地址设置为 0。I/O 端口、中断和特殊选项可以保留为空。
3. 在端口格式中，使用特殊选项字段输入目标 IP 地址，使用以下格式：

-Ia -Pn -T

其中：

a = 标准 Internet 点格式中的目标 IP 地址。（例如，192.9.2.60）

n = 目标端口号。通常，一个物理端口有多个虚拟端口，用于多个使用目的。只有当需要覆盖缺省值 2222 时才使用此选项。

T = 强制驱动程序使用 TCP（默认），而不使用 UDP (-U)。

（有关详细信息，请参见 [TCP/IP 驱动程序专用选项参考](#)）

保留所有其他字段为空。现在，您可以定义您的 I/O 设备单元。

注意： 您只需要定义 TCP/IP 板卡一次。然后增加几个使用相同 Vijeo Citect 板卡的端口。

另请参见

[使用串行板卡](#)

TCP/IP 驱动程序特殊选项参考

特殊选项（在端口表格中）是用空格分隔的选项，以破折号 (-) 开始，后面紧跟着选项字符。可将以下特殊选项用于 TCP/IP：

- **-A:** 支持通过 Cicode 使用 TCPIP 驱动程序。
- **-Ia.b.c.d:** 定义要连接到的远程 IP 地址。
- **-FC:** 支持假连接。这会创建一个假连接（不会创建实际 IP 连接）。该选项旨在用于以下两种情况：驱动程序希望在不和设备通话的情况下支持虚设连接，或有虚拟单元。可以通过虚拟单元访问无需和设备通话的驱动程序地址。
- **-K:** 设置插口 SO_KEEPALIVE 标志。
- **-LIa.b.c.d:** 定义本地 IP 地址。
- **-LPn:** 定义本地 PORT。
- **-Ma.b.c.d:** 定义多点传送 IP 地址。

- **-Pn**: 定义要连接到的远程 PORT。
- **-RC**: 激活重新尝试接收 FD_CLOSE 事件的重新连接。只有 Keepalive 选项被激活才能接收 FD_CLOSE。-RC 可以解析 TCP/IP 驱动程序被通知连接关闭的问题。要进行更全面的处理，需使用 -k 选项。还可以通过以下高层驱动程序调用激活重新尝试：

```
COMSetParam((SHORT)ChannelNumber, (UCHAR*)"DO_RECONNECT", NULL);
```

- **-U**: 针对 UDP（数据报）运行设置此端口

其中：

- **a.b.c.d**: 使用十进制数字 0- 255 的句点符号中的标准 IP 地址。（添加 IP 地址时不要以 0 开头）。
- **n**: 所需服务的端口 ID 的十进制数值。
- **-T**: 针对 TCP（数据流）运行设置此端口。

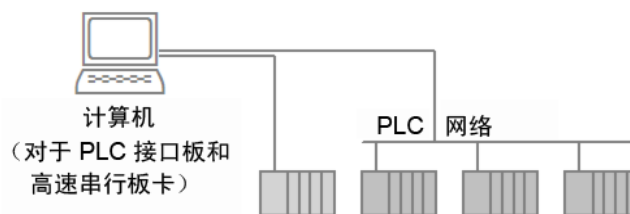
使用专用板卡和/或中间软件

通过某些品牌的 I/O 设备，可以在计算机中安装专用接口板卡或中间软件。此 PLC 接口板卡/软件由 PLC 制造商提供，用户可将其连接到单个 PLC 或 PLC 网络。



注意：对于某些 PLC，当系统连接到多个 PLC 时，高速串行板卡提供能的性能比 PLC 接口板提供的性能好。

一台计算机中既可以有 PLC 接口板卡，也可以有高速串行板卡。例如，可以将 PLC 网络连接到 PLC 接口板卡，将各个 I/O 设备连接到高速串行板卡。



对于 Vijeo Citect 应用程序有很多可能的硬件排列。Vijeo Citect 是非常灵活的系统，对可以使用的 I/O 设备类型（制造商）和将其连接到计算机的方式限制很少。

如何将 Vijeo Citect 设置为使用专用板卡/软件:

如果要使用专用板卡/软件（也就是由 PLC 制造商提供的板卡），请执行以下操作：

1. 在计算机中安装板卡/软件，并根据附带的说明在 Windows 下对其进行设置。使用制造商提供的最新驱动程序。
2. 如果可能，在配置 Vijeo Citect 前对板卡运行诊断程序，以检查板卡是否工作正常。
3. 检查 I/O 端口和中断设置是否正确。
4. 按照特定于 PLC 的帮助中的说明配置板卡和端口。

创建通信测试工程

为了确认 Vijeo Citect 能够将 I/O 设备置于在线状态，建议创建一个测试工程。

测试工程应尽可能简单。例如，如果系统最终将通过 COM 端口、KTX 卡、SA85 卡并通过 TCP/IP 进行通信，第一次尝试时请不要将它们一同测试。一次只使用一个元素来进行测试工程。

首先添加并测试 COM 端口。如果它运行正常，再对 KTX 卡进行测试，然后再对 SA85 卡进行测试，最后再测试 TCP/IP。在确信每个元素都能正常运行后，再开始将它们放在一起测试。

检查能够进行通信的工程中只有绝对少量的信息时，确认没有重复记录。

可靠的方法是打开一个视窗，然后检查视窗左下方显示的记录号。确保它位于记录 1 上，然后点击视窗右侧的滚动条按钮，并向下拉。到达滚动条底部时，释放按钮。如果它弹回到视窗的顶部，且记录号仍保持 1，就可确定只有一条记录。

必须拖拉滚动条，因为所有视窗都在 I/O 服务器名称上编制了索引。通常，在数据库文件中仍然存在来自以前 I/O 服务器名称的“孤立”记录。如果找到任何多余记录，请将它们删除并将工程打包。通信数据库中的重复或孤立记录可能对通信性能产生负面影响。

另请参见

[运行测试工程](#)

运行测试工程

运行测试工程之前，请确认在 Vijeo Citect 启动时启用了内核，以便可以按照启动过程逐步完成操作。

为此，请编辑 citect.ini 文件，使其包含以下内容：

```
[DEBUG]
```

```
Kernel=1
```

这样将会在 Vijeo Citect 启动时显示内核。

然后，运行计算机设置向导。确认将计算机定义为单机 Vijeo Citect 系统，并配置为运行测试工程。

启动工程的运行。显示内核时，双击主窗口中的标题栏，然后双击内核的标题栏。按顺序进行非常重要，因为它会将主窗口最大化，然后将整个内核最大化。如果没有按顺序进行，则不能查看发生的情况，因为内核中的信息不能滚动，并且从屏幕上消失后即丢失。这可能需要多实践几次，因为启动过程在速度很快的计算机中可能只需 1-2 秒或更少。

工程开始运行后（并假设每个部件都正常运行），内核主窗口的最后一行应通知用户 I/O 设备现在在线。不要将通知用户端口通道在线的信息与之混淆，Vijeo Citect 应先报告可以通过设置的端口进行通讯，然后报告可以与 I/O 设备通信。

另请参见

[测试工程不能编译时](#)

当测试工程不能通信时

检查每一个设备和设置，然后重新运行。使用不同的端口驱动，有不同的调试方法。然而，它们基本上是一样的。每次添加一个条目进行测试。

另请参见

[排除设备通信故障](#)

使用通讯快速向导

可以使用快速通讯向导设置与 I/O 设备的通讯。

根据用户的选择，快速通讯向导提供与 I/O 设备通讯要求关联的缺省参数值和设置。

如何访问通讯快速向导：

1. 打开工程编辑器。
2. 选择**通讯 | 快速向导**，或打开 Citect 浏览器。
3. 双击当前工程的“通讯”文件夹中的**快速 I/O 设备设置**图标。
4. 完成向导。

注意：每个 I/O 设备/协议组合需要板卡、端口和 I/O 设备表单的唯一设置。请参见每个设备的特定帮助。

另请参见

[快速通讯向导 - 简介](#)

快速通信向导 - 简介

向导将要求您选择 I/O 服务器、选择名称并指明 I/O 设备的类型（外部、内存、磁盘）。从可用制造商列表中选择 I/O 设备的制造商、型号和通讯方法。如果连接外部设备或在计算机中使用所有权板卡，则向导可能要求您指定地址和通讯端口。

完成设置之后，“摘要页面”将汇总 I/O 设备和/或内部板卡的配置。点击**完成**保存列出的配置，或点击**后退**更改前面的选择。

另请参见

[快速通信向导 - 服务器选择](#)

快速通信向导 - 服务器选择

选择一个在当前工程中定义的现有 I/O 服务器，或创建一个新的 I/O 服务器。

当您创建一个新的 I/O 服务器时，Vijeo Citect 将自动建议它的名称是 **IOServer1**。如果希望，可以输入其他名称。您指定的名称不能超过 16 个字符，并且应使用字母数字字符（A-Z、a-z、0-9）。还可以使用下划线字符（_）。

注意：如果您添加新的 I/O 服务器，首先需要在指定的计算机上运行“计算机设置向导”，然后再尝试运行工程。这是使计算机用作 I/O 服务器所要求的。

另请参见

[快速通信向导 - 设备选择](#)

快速通信向导 - 设备选择

输入要创建的新 I/O 设备的名称，或接受 Vijeo Citect 自动建议的名称 **IODev**。

另请参见

[快速通信向导 - I/O 设备类型](#)

快速通信向导 - I/O 设备类型

选择 I/O 设备的类型。您可以选择**外部 I/O 设备**或**磁盘 I/O 设备**。

另请参见

[快速通信向导 - I/O 设备通讯选择](#)

快速通信向导 - I/O 设备通讯选择

从可用制造商列表中选择特定于 I/O 设备的制造商、型号和通讯方法。

如果选择了内存或磁盘 I/O 设备，将在树的顶部包括 Vijeo Citect 通用协议。

另请参见

[快速通信向导 - TCP/IP 地址](#)

快速通信向导 - TCP/IP 地址

输入 IP 地址，采用标准 Internet 点格式（例如 192.9.2.60）。这个地址在您的 I/O 设备上设置，或由您的 I/O 设备指定。

I/O 设备的端口号和协议（TCP 或 UDP）域已经被设置为缺省值。只有在必要时才应更改这些域。

有关特定 I/O 设备地址的详细信息，请点击[驱动程序地址帮助](#)，浏览 I/O 设备的特定于驱动程序的信息。

另请参见

[快速通信向导 - I/O 设备地址](#)

快速通信向导 - I/O 设备地址

输入 I/O 设备的地址。在此字段中输入的内容取决于使用的 I/O 设备类型和协议类型，因为每种类型的寻址策略不同。

有关特定 I/O 设备地址的详细信息，请点击[驱动程序地址帮助](#)，浏览 I/O 设备的特定于驱动程序的信息。

另请参见

[快速通信向导 - I/O 设备连接调度](#)

快速通信向导 - I/O 设备连接调度

这个表格允许您定义 I/O 设备的通讯调度的详细信息，并通过选中 PSTN 框指明 I/O 设备是远程设备。

选项	描述
连接 I/O 设备到 PSTN	<p>选中此框，指明此 I/O 设备是拨号 I/O 设备（连接到 PSTN - 公共交换电话网络）。</p> <p>注意： 即使您的 I/O 设备没有通过调制解调器连接，您也必须选中此框以调度通讯（但应将“拨打的电话号码”和“呼叫者 ID”域保留为空）。</p> <p>一旦您使用此向导完成了 I/O 设备设置，您必须转到“端口”表单，将端口号更改为实际的 COM 端口号。</p> <p>您可以选择定义通讯周期，使用月、周、日或小时、分钟和秒。或者，您可以选择仅在启动时通讯（永久连接）。点击一个单选按钮做出选择，然后输入开始时间和周期，如下所述：</p>
同步时间	<p>I/O 服务器将在这个时刻尝试与 I/O 设备进行通讯，然后按下面定义的时间间隔进行通讯。此时间对 Vijeo Citect 来说仅仅是一个参照。如果您</p>

选项	描述
	在这个时间之后运行某个工程，那么 I/O 服务器不会等到第二天才开始通讯。工程运行在开始时间之前还是之后没有差别。
重复间隔	连续的通讯尝试之间相隔的时间。（请参见下面的示例。）
拨打的电话号码	启动与 I/O 设备的联系所需拨打的电话号码。

注意：也可以使用“工程编辑器”中的“I/O 设备”表单设置这些值。

示例

均以**同步时间 10:00:00**为基准：

- 如果**重复间隔**字段中输入 12: 00: 00，并在上午 9 点启动工程，则 I/O 服务器将在上午 10 点与 I/O 设备通讯，此后每 12 小时通讯一次（即下午 10 点，然后是第二天上午 10 点，依此类推）。
- 如果您在**重复间隔**域中输入 12:00:00，并在下午 4 点启动您的工程，I/O 服务器将在下午 10 点与 I/O 设备通讯，然后在第二天的上午 10 点再次通讯，等等。它将假设通讯是上午 10 点建立的，因此将继续按此假设进行推理，即在上午 10 点后每隔 12 小时重复通讯一次。
- 如果您在**重复间隔**域输入 3 天，并在星期三上午 9 点启动工程，则 I/O 服务器将在上午 10 点与 I/O 设备通讯，此后每 3 天通讯一次（即星期六上午 10 点，然后是下周的星期二上午 10 点，依此类推）。
- 如果您在**重复间隔**中输入 12 月 6 日，并在 11 月启动您的工程，那么 I/O 服务器将在 12 月 6 日上午 10 点与 I/O 设备通讯，然后在下一年的 12 月 6 日再次通讯。
- 对长期连接，选择**启动时**。要断开长期连接，必须调用类型为 8 的 IODeviceControl() 函数。

另请参见

[呼叫者 ID 和命令](#)

[快速通信向导 - 链接到外部数据库](#)

呼叫者 ID 和命令

呼叫者 ID

远程 I/O 设备回拨到 I/O 服务器时识别远程 I/O 设备的唯一标识符。呼叫者 ID 可以是任何字母数字和字符“_”（下划线）的组合。

此 ID 仅在 I/O 设备启动对 I/O 服务器的呼叫时使用。调制解调器启动呼叫时也必须要在调制解调器上设置呼叫者 ID。

注意：如果使用的是多路调制解调器，应使用 I/O 设备发出呼叫者 ID，而不是使用调制解调器。这是因为无论调用与哪一个 I/O 设备相联系，使用调制解调器指定 ID 都将发送相同的 ID，从而很难识别触发调用的 I/O 设备。

通过使用 I/O 设备发放 ID，I/O 服务器将收到每个 I/O 设备的唯一呼叫者 ID。但不是所有 I/O 设备都能发放呼叫者 ID。如果使用的是多点式通讯，则应使用能够发放呼叫者 ID 的 I/O 设备。

选项	描述
【事件命令】连接时	一旦连接调制解调器和 I/O 设备开始在线后（即在处理任何读写请求之前）要执行的 Cicode。
【事件命令】断开时	终止与 I/O 设备的连接之前（以及处理了所有读写请求之后）要执行的 Cicode。

注意：也可以使用工程编辑器中的“I/O 设备”视窗设置这些值。

另请参见

[快速通讯向导 - 链接到外部数据库](#)

快速通信向导 - 链接到外部数据库

这个屏幕允许您链接到外部数据源。

选项	描述
链接 I/O 设备到一个外部标签数据库	<p>确定是否希望将 I/O 设备链接到一个外部数据源。如果链接到一个外部数据源，每次执行刷新时，Vijeo Citect 都会随着外部数据源的任何变化而更新。</p> <p>如果您断开一个现有链接的连接，可选择制作数据源中所有标签的本地副本，或者可以从 Vijeo Citect 的标签变量数据源中将它们完全删除。</p>
数据库类型	外部数据源参考的数据格式。
外部标签数据库	<p>指定外部数据库的位置。这可以是 I/O 设备的外部数据源的路径和文件名，或是 IP 地址/目录、计算机名或一个数据服务器的 URL 等（例如，"Work.CSV" 或 "127.0.0.1", "139.2.4.41\HMI_SCADA" 或 "http://www.abicom.com.au/main/scada" 或 "\\coms\data\scada"）。</p> <p>根据选定的数据库类型，浏览按钮可以：</p> <ul style="list-style-type: none"> 显示一个标准 Windows 文件浏览对话框，浏览外部数据库。 显示一个模式对话框，浏览与这台计算机相连的网络上的所有服务器的树状

选项	描述
	视图。 显示 联合链接配置属性 对话框。
连接字符串	输入连接字符串，提供数据源的连接详细信息。这类似于 ODBC 连接字符串。例如： UserID = XXX; Password = YYY 或 ServerNode=111.2.3.44; Branch=XXX 不是所有的数据源都需要连接字符串。
为外部链接标签添加前缀	如果希望在 Variable.DBF 中的所有链接标签的名称前插入一个前缀，请选中此框。
标签前缀	将被插入到 Variable.DBF 中的链接标签的名称前的前缀（仅为 I/O 设备）。要更改此前缀，应该首先删除该前缀，然后执行手动刷新，最后添加新的前缀。
自动刷新标签	确定当外部数据源变化时（例如手动更改一个域等），Vijeo Citect 的标签变量数据库中的链接标签是否更新。第一次链接到数据源时将执行刷新，随后尝试读取已更改的变量（例如，编译工程、使用“变量标签”表显示变量或粘贴标签）时也执行刷新。 没有自动刷新时，您需要执行手动刷新来更新 Vijeo Citect 中的链接标签。
实时更新	只有当您安装了 Vijeo CitectSpeedlink 产品之一时，此域才可用。有关 FastLink 的详细信息，请参见 http://www.citect.com 。 控制 Vijeo Citect 和外部标签数据库中的链接标签在任何数据库发生变化时是否同步。要启用实时链接，从 实时更新 菜单中选择 是 ，并确认没有选择 自动刷新 复选框。（实时更新和自动刷新相互排斥。） 当启动实时更新并且访问 Vijeo Citect 标签变量数据库时（例如在工程编译期间或填充下拉列表时），Vijeo Citect 将查询外部标签数据库以确定它是否已被修改。如果是这样，Vijeo Citect 会将更改合并到本地变量标签数据库中。反过来，任何对本地标签变量数据库所作的更改也将被无缝结合到外部标签数据库中。

另请参见
[快速通信向导 - 串行设备](#)

快速通信向导 - 串行设备

因为您的协议基于串行通讯，所以您必须选择用于 I/O 设备的计算机端口。

已从您的操作系统注册表中检测到列出的串行端口。如果您已经正确安装了专有串行板卡（例如 Digiboard）和相关的驱动程序，将列出可用端口。

另请参见
[快速通信向导 - 摘要](#)

快速通信向导 - 摘要

使用您提供的信息汇总 I/O 设备设置。摘要显示了 Vijeo Citect 通讯设置和 I/O 设备和/或内部板卡的建议配置。

手动配置通信

通常情况下，使用快速通信向导就足够用来完成通信设置。不过，如果需要手动配置通信，可按照下述步骤进行：

1. 在 [I/O 服务器属性](#) 表中定义一个 I/O 服务器。此步骤定义 I/O 设备将与之通信的 Vijeo Citect 服务器的名称。
2. 完成 [板属性](#)。此步骤定义（您的 Vijeo Citect 计算机上）将用来通信的板卡（主插件卡、网卡、串行板卡或 PLC 通信卡）。拨号远程 I/O 设备必须使用 COMx 板卡。
3. 完成 [端口属性](#)。通常，板卡有多个通信端口，您必须指定要使用的端口。某些设备允许为一个物理端口指定多个逻辑（虚拟）端口。如果使用调制解调器，必须为每个端口指定唯一端口名称，并将该端口号设为-1。而且还必须指定该端口的通信参数和任何特殊行为。
4. 完成 [I/O 设备属性](#)。此步骤通过指定地址来定义 Vijeo Citect 将与之通信的 I/O 设备。协议也在此级别定义。
5. 运行计算机设置向导完成配置。这使您可以将您的 Vijeo Citect 计算机定义为上面定义的 I/O 服务器。这一步骤通常在编译工程之后进行。
6. 如果使用的是拨号远程 I/O 设备，必须完成“调制解调器”对话框。此步骤定义 Vijeo Citect 如何使用调制解调器来与远程 I/O 设备进行通信。

注意： 如果没有要读取或写入的数据，Vijeo Citect 将不与 I/O 设备进行通信，无论它定义与否。必须创建一个标签变量，并在 Vijeo Citect 进行读请求之前使用它。例如，使用整型变量在画面上显示一个数字。

I/O 服务器属性

要定义 I/O 服务器，必须在 I/O 服务器表单中指定一个名称。此名称将用于引用 I/O 服务器，并且此名称应为逻辑名称。例如，对于单一 I/O 服务器，可以使用名称 IOserver。指定服务器名称时，不能使用本地化的字符。请参见 [I/O 服务器定义](#)。

如果使用的是用于冗余的多个 I/O 服务器（或用于分离通讯），则必须为每个服务器都添加数据库记录。为每个 I/O 服务器选择唯一的名称，例如，IOserver1 和 IOserver2。

注意： 如果已对记录进行了更改，则必须将记录添加到工程数据库（使用表单底部的“添加”按钮）或替换记录（使用表单底部的“替换”按钮）。

添加 I/O 服务器

为将一个计算机定义成一个 I/O 服务器，必须在此计算机上运行“计算机设置向导”。该向导允许您选择已在项目中配置的所有 I/O 服务器。

板卡属性

板卡的属性取决于安装在 I/O 服务器计算机上的板卡的类型。

注意：“板卡”是一个传统术语，起源于计算机上的大多数端口都由附加板卡提供的年代。例如，以太网卡。即使物理板卡可能都不存在了，逻辑板卡概念仍然会有用（它可以是软件）。

板卡具有下列属性：

板卡名

板卡的名称。最多 16 个字符。例如，Server1_Board1。

如果 I/O 服务器计算机中有多个板卡，每个板卡的名称必须唯一。如果有多个 I/O 服务器，板卡名只要在每个服务器内是唯一的即可。例如，Server1_Board2

板卡类型

板卡的类型。最多 16 个字符。

如果使用串行板卡或计算机的 COM 端口，应输入 COMx。

地址

板卡的起始地址。例如，0xCC00。最多 8 个字符。

将板卡安装到计算机之后，必须指定与板卡上的开关设置匹配的地址。如果使用的是串行板卡或计算机的 COM 端口，应输入地址 0。

注意：如果在同一台计算机上安装了多个板卡，请对每个板卡使用不同的内存地址。

I/O 端口

板卡的 I/O 端口地址。最多 8 个字符。

将板卡安装到计算机之后，必须指定与板卡上的开关设置匹配的地址。

注意：如果使用的是计算机的 COM 端口，则不应在此处输入端口地址。应在“端口”视窗中指定端口号。

中断号

板卡使用的中断号。如果使用的是计算机的 COM 端口，则不需要此端口号。

特殊选项

板卡支持的所有特殊选项。最多 32 个字符。请检查针对特定 I/O 设备的硬件配置帮助主题，确定是否需要填写特殊选项。

注释

任何有用的注释。最多 48 个字符。

端口属性

端口的属性取决于安装在 I/O 服务器中板卡的类型以及与此端口相连的 I/O 设备。端口具有下列属性：

端口名称

连接到 I/O 设备的端口的名称。最多 32 个字符。每个端口都必须有一个唯一名称（也就是说，不能为系统中的两个端口指定相同的名称）。可以使用任意名称（最多为 16 个字符），例如：`Board1_Port1`

如果在计算机中有多个板卡，可以使用端口名称来标识板卡，例如：`Board2_Port1`

端口号

I/O 设备连接到的端口号。最多 4 个字符。除非通过调制解调器连接到拨号远程 I/O 设备（参见下面的注意），否则请不要将相同端口号指定给一个板卡上的两个端口。不同板卡上的端口可以分配相同的端口号。

如果使用计算机的 COM 口，则应在这里输入端口号 - 端口号在 Windows 控制面板的“端口”部分定义。

注意：如果要（通过调制解调器）连接到可拨号的远程 I/O 设备，必须在 I/O 服务器上为每个可拨号远程 I/O 设备定义唯一的端口名称，且该端口号必须为 -1。

板卡名

用于板卡的名称。最多 16 个字符。这在将端口链接到板卡时是必需的。例如，`Server1_Board1`

波特率

通信通道的波特率（Vijeo Citect I/O 服务器与 I/O 设备之间）。最多 16 个字符。

注意：I/O 设备硬件和串行板卡可能支持其他波特率。如果确实要选择其他波特率，确认 I/O 设备和串行板卡都支持该新波特率。

数据位

数据传送中使用的数据位数。必须将 I/O 设备设置到同一值，否则无法建立通信链接。

停止位

用来表示通信完成的停止位数。必须将 I/O 设备设置为与此值相同。

奇偶校验

数据传送中使用的数据奇偶校验。

特殊选项

端口所支持的所有特殊选项。最多 32 个字符。请查阅所使用 I/O 设备的硬件设置帮助主题，以确定是否需要填写特殊选项。

注释

任何有用的注释。最多 48 个字符。

I/O 设备属性

配置 I/O 设备包括使用工程编辑器指定其属性。这些属性取决于协议和 I/O 设备。

如何配置 an I/O 设备:

1. 在项目编辑器中，选择**通讯 | I/O 设备**。将显示“I/O 设备”对话框。
2. 在**名称**字段中，键入 I/O 设备 (PLC) 的名称。Vijeo Citect 系统中的名称必须唯一，除非 I/O 设备是在其它提供冗余的 I/O 服务器中定义的。如果使用冗余，则每个 I/O 服务器中 I/O 设备的 I/O 设备号和地址必须相同。应对主要和备用 I/O 设备使用不同的 I/O 设备名称，否则 I/O 设备 Cicode 函数无法区分它们。最多 31 个字符。
3. 在**编号**字段中，输入 I/O 设备的唯一编号 (0-16383)。最多 8 个字符。Vijeo Citect 系统中的编号必须唯一，除非 I/O 设备是在其它提供冗余的 I/O 服务器中定义的。如果使用冗余，则每个 I/O 服务器中 I/O 设备的 I/O 设备号和地址必须相同。可以使用相同的设备名称，但如果希望使用 I/O 设备 Cicode 函数，则 I/O 设备名称不同更容易进行区分。
4. 在**地址**字段中，输入 I/O 设备的地址。最多 64 个字符。在此字段中输入的内容取决于使用的 I/O 设备类型和协议类型，因为每种类型的寻址策略不同。
5. 在**协议**字段中，选择要用于与 I/O 设备通讯的协议。最多 16 个字符。许多 I/O 设备都支持多种协议，这取决于所选择的通讯方法。
6. 在**端口名**字段中，在 I/O 设备连接到的板卡中指定端口。最多 32 个字符。这在将 I/O 设备链接到端口时是必需的。例如 Board1_Port1。

注意：一个服务器最多只能有 255 个 COMx 端口。要避免对远程 I/O 设备数量的限制，可以将多个远程 I/O 设备连接到同一个端口，只要通讯详细信息（电话号码、波特率、数据位、停止位和奇偶校验）相同。

7. 在**内存**字段中，指定 I/O 设备是否在“内存”模式下运行。最多 8 个字符。缺省参数值为 FALSE。如果选择 TRUE，则 I/O 设备将在内存中创建，并且其参数值在运行时存储

在内存中。如果要在连接物理 I/O 设备之前测试系统，这将非常有用，因为内存模式会阻止 Vijeo Citect 与 I/O 设备进行通讯。请参见 [使用内存模式](#)。

8. 在**注释**字段中，键入任何有用的注释。最多 48 个字符。此字段是可选的，不在运行时使用。
9. 单击“添加”添加新的记录，或单击“替换”替换现有记录。

扩展形式的字段

以下字段通过扩展形式实现（按 **F2**）。

链接

确定 I/O 设备是否连接到外部数据源。最多 12 个字符。如果链接到外部数据源，则会使用执行刷新时对外部数据源所做的任何更改对 Vijeo Citect 进行更新。

如果断开现有链接，则可以选择在数据库中创建所有标签的本地副本，或从 Vijeo Citect 的变量标签数据库中将其全部删除。

如果 I/O 设备链接到外部数据源，“数据库类型”、“外部数据库”、“连接字符串”、“标签前缀”和“实时更新”字段将灰显。

链接外部标签数据库

I/O 设备的外部数据源的路径和文件名。最多 253 个字符。另外，也可以输入 IP 地址/目录、计算机名或数据服务器的 URL 等（例如，“C:\Work.CSV”、“127.0.0.1”、“139.2.4.41\HMI_SCADA”、“http://www.abicom.com.au/main/scada”或“\\coms\data\scada”）。

单击**浏览**可选择路径和文件名。

连接字符串

输入连接字符串可为数据源提供连接详细信息。最多 128 个字符。这类似于 ODBC 连接字符串。例如：

```
UserID = XXX; Password = YYY
```

或

```
ServerNode=111.2.3.44; Branch=XXX
```

不是所有的数据源都需要连接字符串。

链接数据库类型

外部数据源引用的数据格式。最多 128 个字符。

标签前缀

Vijeo Citect 的变量标签库（仅适用于此 I/O 设备）中所有链接标签的名称前面插入的前缀。最多 8 个字符。要更改此前缀，应首先将其删除，然后执行手动刷新，最后添加新的前缀。

自动刷新标签

确定外部数据源发生变化时（例如，手动更改字段等）是否更新 Vijeo Citect 变量标签库中的链接标签。最多 8 个字符。第一次链接到数据源时将执行刷新，随后尝试读取已更改的变量（例如，编译工程、使用“变量标签”表显示变量或、修改或粘贴标签）时也执行刷新。

如果不能自动刷新，则需要执行手动刷新来更新 Vijeo Citect 中的链接标签。

实时更新

注意：此字段仅在安装了 Vijeo Citect FastLinx 产品之一时可用。有关 FastLinx 的详细信息，请参见 <http://www.citect.com>。

控制 Vijeo Citect 和外部标签数据库中的链接标签在任何数据库发生变化时是否同步。要启用实时链接，请从“实时更新”菜单中选择“是”。

启用实时更新并访问变量标签数据库时（例如，工程编译期间或填充下拉列表时），Vijeo Citect 可查询外部标签数据库以确定是否对其进行修改。如果是这样，Vijeo Citect 会将更改合并到本地变量标签数据库中。反过来，任何对本地标签变量数据库所做的更改也将被自动结合到外部标签数据库中。

启动模式

I/O 设备冗余的类型。

主要	启用立即使用此通讯通道。如果未指定模式，则此模式为缺省模式。
备用	除非 I/O 设备配置中主通道发生故障，否则此通道将保留不用。
StandbyWrite	<p>除非 I/O 设备配置中主通道发生故障，否则此通道将保留不用。所有发送至主要通道的写请求也发送至此通道。</p> <p>注意：对已调度的 I/O 设备，请不要使用此模式，因为有可能再也无法清除备用 I/O 设备的写队列。如果向未建立通信的已调度的 I/O 设备中写入，则写请求会加入队列，直到通信恢复。在使用 I/O 设备冗余和 StandbyWrite 模式下，主要和备用 I/O 设备都有其各自的写请求等待队列。如果 I/O 设备通信由通信计划表自动控制，则当按计划表恢复通信时将转存这两个队列。对于 y 由 IODeviceControl Cicode 函数控制的手动通信，则只有主要 I/O 设备写队列被转存，而备用 I/O 设备写队列一直不会被清除。</p>

注意：要使用备用模式或辅写模式，还必须通过相同的 I/O 设备名称、编号和地址配置主要 I/O 设备。

优先级

如果主要 I/O 设备在运行时出现故障，则需要设置备用设备的提升顺序。最多 8 个字符。

如果为集群配置了多个备用 I/O 设备，则可以使用此字段为特殊备用设备提供优先权。如果此字段保留为空，系统会根据其它备用设备和/或设备的配置顺序在工程编译期间将优先权自动分配给设备。

请参见 [I/O 设备提升](#)。

记录写

启用/禁用对 I/O 设备的写操作进行记录。启用此功能时，所有写操作都记录到安装期间选择的 Vijeo Citect **用户和数据** 文件夹 **记录** 文件夹的 SYSLOG.DAT 文件中，并在 INI 文件中指定为 [CtEdit]Logs。有关 TagWriteEventQue 和记录标签数据的信息，请参见“标签函数”。

注意：记录对 I/O 设备的写操作会使通讯变慢，因为 Vijeo Citect 系统会将大量数据写入磁盘。但写操作的记录在调试系统时非常有用。

记录读

启用/禁用对 I/O 设备的读操作进行记录。启用此功能时，所有读操作都记录到 Vijeo Citect SYSLOG.DAT 文件中。

注意：记录对 I/O 设备的读操作会使通讯变慢，因为 Vijeo Citect 系统会将大量数据写入磁盘。但读操作的记录在调试系统时非常有用。

缓存

启用/禁止数据缓存。启用此功能时，I/O 设备内存的缓存保留在 I/O 服务器中，从而能够缩短通讯往复时间。

注意：应对调度的 I/O 设备启用缓存，而对内存或磁盘 I/O 设备禁止缓存。

缓存时间

以毫秒为单位指定的缓存时间。最多 8 个字符。启用缓存时，所有从 I/O 设备读取的数据都被临时存储在 I/O 服务器的内存中。如果在缓存时间内从同一台或另一台客户端对同样的数据提出了另一个请求，则 Vijeo Citect I/O 服务器将返回内存中的参数值，而不是再次读取 I/O 设备。当许多客户机需要相同数据时，数据缓存能够在总体上提高响应速度。

建议将缓存时间设置为 300 毫秒。

调度的 I/O 设备的缓存时间是自动计算的。可以使用“内核页面单元”命令查看调度的 I/O 设备的缓存时间。

调度

确定是否为调度的通讯配置 I/O 设备。这通常使用快速通讯向导进行设置。

注意：如果未对可拨号的远程 I/O 设备指定调度，则系统会在启动时建立连接并保持连接状态，直到系统关闭为止。

请参见 [调度的通讯](#)。

时间

I/O 服务器将在这个时刻尝试与 I/O 设备进行通讯，然后按下面定义的时间间隔进行通讯。这个时刻只是对 Vijeo Citect 的标记。如果在这个时刻之后运行工程，则 I/O 服务器不会等到第二天开始通讯，而是继续操作，如同工程已自开始时间之前运行。

周期

连续通讯尝试之间相隔的时间。

范例（都基于 10:00:00 时的同步）：

1. 如果“重复间隔”字段中输入 12: 00: 00，并在上午 9 点启动工程，则 I/O 服务器将在上午 10 点与 I/O 设备通讯，此后每 12 小时通讯一次（即下午 10 点，然后是第二天上午 10 点，依此类推）。
2. 如果输入 12: 00: 00，并在下午 4 点启动工程，则 I/O 服务器将在晚上 10 点与 I/O 设备通讯，然后是第二天上午 10 点再次通讯，依此类推。I/O 服务器会假设通讯是在上午 10 点建立的，因此按照这个假设在上午 10 点之后每 12 小时通讯一次。
3. 如果输入 3 天，并在星期三上午 9 点启动工程，则 I/O 服务器将在上午 10 点与 I/O 设备通讯，此后每 3 天通讯一次（即星期六上午 10 点，然后是下周的星期二上午 10 点，依此类推）。
4. 如果“重复间隔”中输入 12 月 6 日，并在 11 月期间启动工程，则 I/O 服务器将在 12 月 6 日上午 10 点与 I/O 设备通讯，然后是第二年 12 月 6 日再次通讯。

对长期连接，选择“启动时”。要断开长期连接，必须调用类型为 8 的 IODeviceControl() 函数。

连接动作（254 个字符）

建立与 I/O 设备的通讯后、处理任何读请求或写请求前执行的 Cicode。

断开动作（254 个字符）

终止与 I/O 设备的通讯并处理了所有读请求和写请求后执行的 Cicode。

电话号码（64 个字符）

仅适用于 Windows 2000

启动与 I/O 设备的联系所需拨打的电话号码。（即，用于可拨号的远程 I/O 设备）

呼叫者 ID（32 个字符）

仅适用于 Windows 2000

远程 I/O 设备回拨到 I/O 服务器时识别远程 I/O 设备的唯一标识符。调用方 ID 可以是字母数字字符和/或字符“_”（下划线）的任何组合。

此 ID 仅在 I/O 设备启动对 I/O 服务器的呼叫时使用。如果调制解调器启动呼叫，则必须在调制解调器上设置呼叫者 ID。

注意：如果对单个调制解调器进行多点式通讯，则应使用 I/O 设备发放呼叫者 ID，而不是调制解调器。这是因为使用调制解调器发放 ID 时，无论调用与哪个 I/O 设备相关，它都发送相同的 ID。这样会难以识别触发调用的 I/O 设备。

通过使用 I/O 设备发放 ID，I/O 服务器将收到每个 I/O 设备的唯一呼叫者 ID。但不是所有 I/O 设备都能发放呼叫者 ID。如果使用的是多点式通讯，则应使用能够发放呼叫者 ID 的 I/O 设备。

后台轮询

指定特殊设备的所有标记以最小后台轮询率连续轮询。选项为 True 或 False。如果在低速网络中操作、设备运行速度缓慢或在通常只进行很少轮询标签的位置，则将此字段设置为 True。

速率

将“后台轮询”设置为 True 时，指定设备应按其轮询的最小速率。可以从下拉列表中选择任何预定义的参数值，或采用 HH:MM:SS 格式输入自己的参数值。如果不输入参数值，将使用缺省值 30 秒。

另请参见

[使用可拨号的远程 I/O 设备通讯](#)

使用设备驱动程序

驱动程序使 Vijeo Citect 能够与系统设备通讯。在 Vijeo Citect 里的每一个通讯设置都需要驱动程序实施协议和传输。往往这意味着两个不同的驱动程序，但是在某些情况下，它们会被结合为一个驱动程序 - 叫做“板”驱动程序。

Vijeo Citect 拥有超过 140 个可用设备驱动程序，可通过几个通信类型实现与大批产品设备的通信。这些包括支持工业标准协议的通用驱动程序，例如 OPC 和 Modbus。

用户与特定设备通信需要的驱动程序由下面确定：

- 设备本身

- 设备支持的通信协议
- 用来连接设备与 Vijeo Citect I/O 服务器的通信通道。

为了在 Vijeo Citect 工程里添加某个特定设备，用户要确定支持此设备的驱动程序。如果特定的驱动程序没安装在 Vijeo Citect 的用户当前版本上，必须获得驱动程序包并单独安装。

注意：涉及通信时往往会用到“驱动程序”。驱动程序是执行某个协议或传输（或两者皆有）的一个软件组件部分。驱动程序是模块化的，因此新的驱动程序插入到现有系统很容易。

另请参见

[确定要使用的驱动程序](#)

[安装驱动程序包](#)

确定要使用的驱动程序

要将设备连接到 Vijeo Citect，首先要确定需要使用哪种驱动程序才能支持与 I/O 服务器之间的通讯。大多数情况下，您可以使用以下方法之一：

- 本地驱动程序，专门针对设备构建
- 通用驱动程序，适用于支持行业标准协议（如 Modbus 或 DNP）的设备
- OPC 服务器，通过第三方应用程序进行通讯

应首先检查本地驱动程序是否可用。为此，进入**支持的设备**画面（位于**驱动程序参考帮助**中），然后查找要连接到的设备。此列表根据制造商名称按字母顺序排序。（要访问“驱动程序参考帮助”，请从 Citect 工程管理器 and 工程编辑器中的**帮助**菜单中选择**驱动程序帮助**。）

如果此设备未包含在列表中，您应该考虑设备的通讯能力，进而确定它是否支持行业标准协议（如 Modbus、DNP、COMLI 或 SNMP）。如果是这种情况，可以使用支持这些标准的通用 Vijeo Citect 驱动程序。

与之类似，您可以使用 OPC 将设备连接到 Vijeo Citect，但这需要购买第三方应用程序以建立 OPC 服务器。如果采用此方法，应该使用试图连接的设备的制造商所推荐的 OPC 服务器应用程序。

如果这些选项无法提供合适的解决方案，可以联系 [技术支持](#)。

注意：有关驱动程序更新和新驱动程序包发布的信息，应该查看 Citect Driver Web (www.citect.com/driverweb)。

另请参见

[安装驱动程序包](#)

安装驱动程序包

如果所需驱动程序是新的，或者最近已经更新，则它可能未包括在随当前版本的安装的驱动程序中。如果是这种情况，您需要获取驱动程序包并独立安装它。参见 Citect 驱动程序网站 (www.citect.com/driverweb)。

必须在将连接到目标设备的计算机上安装驱动程序包。此计算机必须配置为 Vijeo Citect I/O 服务器。它还应该安装在编译 Vijeo Citect 工程的任何计算机上。

安装驱动程序包之前，请关闭您的 Vijeo Citect 运行环境和 Citect 工程管理器。您需要有该 I/O 服务器 PC 的管理员权限。

安装驱动程序包：

1. 将驱动程序包安装到 I/O 服务器 PC 上的适当位置。
2. 双击 EXE 文件以启动安装。
3. 按照安装向导提供的说明操作。
4. 系统会提示您选择 Vijeo Citect 安装文件夹。如果要安装到缺省文件夹之外的文件夹中，请浏览到该位置。如果选择的文件夹不正确，会显示一条提醒消息提醒您。
5. 点击**完成**完成安装。

安装驱动程序包之后，相关帮助将自动合并到驱动程序参考帮助中。

另请参见

[使用驱动程序参考帮助](#)

使用驱动程序参考帮助

每个驱动程序（无论是随 Vijeo Citect 一起安装还是作为驱动程序包的一部分安装）都受驱动程序参考帮助的支持。

可以在驱动程序参考帮助中查找与每个驱动程序相关的特定参考信息。此信息包括：

- 每个驱动程序支持的设备
- 用于识别和查找设备的地址
- 任何所需的设备设置信息
- 在 Vijeo Citect 工程中配置设备所需的信息
- 每个驱动程序支持的数据类型
- 可用于自定义驱动程序的参数
- 特定于驱动程序的错误

对于大多数信息，需要建立与设备的通讯；其中的一些信息更适合高级工程任何和疑难解答。

要访问“驱动程序参考帮助”，请从 Citect 工程管理器 and 工程编辑器中的**帮助**菜单中选择**驱动程序帮助**。

如果熟悉 [设备通讯](#) 中说明的过程，系统会指导用户如何使用此信息在 Vijeo Citect 工程内配置设备。

使用 Citect.ini 参数自定义通讯



警告

不希望的设备操作

- 无论什么情况，请勿更改或删除任何隐含的 `citect.ini` 参数。
- 在删除 `citect.ini` 文件的区段之前，请确认不会删除任何隐含参数。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

注意： 有关隐含功能的信息，始终应咨询 Schneider 支持人员。

可以使用 `Citect.ini` 参数调整 Vijeo Citect 驱动程序的性能，并执行运行时维护诊断。

可以自定义 Vijeo Citect 与驱动程序（甚至单个 PLC）通讯的方式，方法是创建或编辑以特定驱动程序命名的 `citect.ini` 文件的段，例如 `[MODBUS]`。

有些 `ini` 参数通用于所有 Vijeo Citect 驱动程序，而自定义参数与驱动程序一一对应。

有关与每一驱动程序相关的 `ini` 参数及其缺省值，请参见 [驱动程序参考帮助](#)（[帮助菜单](#) | [驱动程序帮助](#)）。

Vijeo Citect 在运行时启动时，它会从存储在本地的 `citect.ini` 文件中读取配置值。因此，所有配置设置都必须包含在充当以下角色的计算机上的 `citect.ini` 文件中：系统内各设备的 I/O 服务器。

缺省情况下，Vijeo Citect 会在 `Vijeo Citect project\bin` 目录中查找 `citect.ini` 文件。如果无法在此处找到该文件，它会搜索缺省的 Windows 目录。

使用磁盘 I/O 设备

除了连接到实际 I/O 设备，Vijeo Citect 可以支持磁盘 I/O 设备的配置，该设备仅存在于您的计算机中。

磁盘 I/O 设备提供了长期的存储。磁盘 I/O 设备中的每一个变量的值存放于计算机硬盘中。使用磁盘 I/O 设备与在内存模式中配置 I/O 设备一样，不同在于保存数据。有关详细信息，请参见 [使用内存模式](#)。

配置后，磁盘 I/O 设备与系统中的任意其他 I/O 设备相同，但不与任意字段设备相连接。磁盘 I/O 设备可保留由 Vijeo Citect 支持的任意类型的变量，且您可以将其配置为枚举任意支持的 I/O 设备。您也可以指定磁盘 I/O 设备的一般协议。

系统计划中关机或意外关机后仍需存放工厂状态时使用磁盘 I/O 设备非常有用。可以配置您的系统，使得磁盘 I/O 设备不断地被更新为定义工厂状态的变量子集。在关闭后重启系统时，Vijeo Citect 会立即恢复这个状态。

也可以使用磁盘 I/O 设备来存储预定义的数据，过程需要时必须立即恢复该数据（例如，一个简单的处方系统）。

注意：如果在计算机中为磁盘 I/O 设备创建 RAM 磁盘，您将不需要创建或复制磁盘文件到该 RAM 磁盘。Vijeo Citect 将自动在启动时创建磁盘文件。如果希望写入磁盘 I/O 设备的数据不受服务器关机的影响，请不要将它设置在 RAM 磁盘上。

另请参见

[磁盘 I/O 设备的设定](#)

[冗余的磁盘（虚拟）I/O 设备](#)

[使用内存模式](#)

磁盘 I/O 设备的设定

要设定与一个设备的通信，请遵循 I/O 设备设定步骤中介绍的基本步骤。

有时，您可能需要直接编辑通信表单。它们需要下列特定的信息。

- 无需完成“通讯板”对话框。
- 无需完成“端口”对话框。
- 按下列步骤完成“I/O 设备”对话框。

通过输入下列信息完成设定步骤：

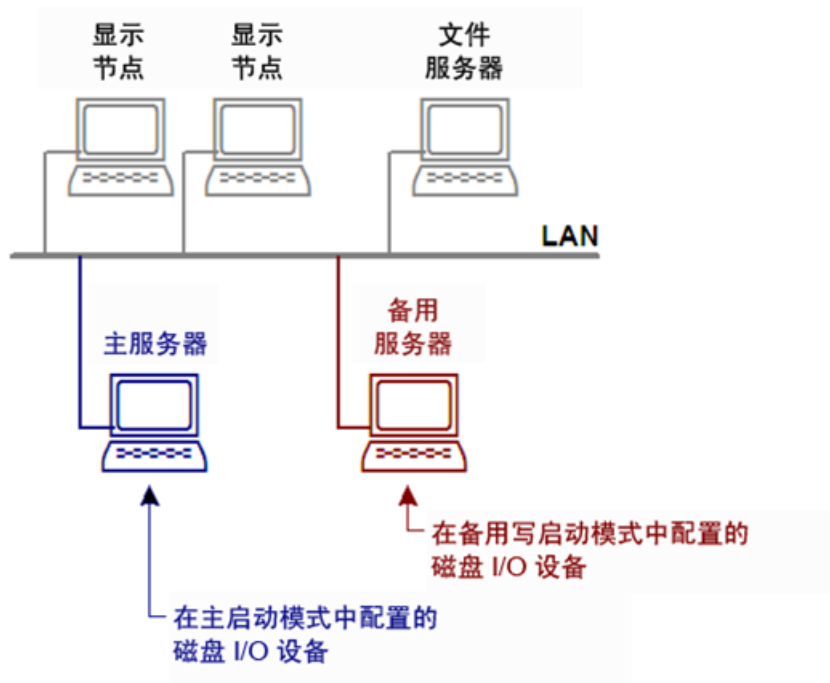
文本框	必填信息
I/O 设备名称	一个名称或者您的磁盘 I/O 设备，例如：DISK_PLC。在 Vijeo Citect 系统中，每个 I/O 设备必须具有唯一的名称。
I/O 设备号	唯一的磁盘 I/O 设备号 (1-4095)。在 Vijeo Citect 系统中，每个 I/O 设备必须具有唯一的设备号。
I/O 设备地址	磁盘文件的路径和文件名称，例如： [RUN]:DSKDRV.DSK 如果您使用的是冗余磁盘 I/O 设备，则必须同时在两个磁盘 I/O 设备的配置中指定主文件和辅文件的路径。 例如，如果是主磁盘 I/O 设备，请输入： Primary_File, Standby_File

文本框	<p>必填信息</p> <p>如果是辅磁盘 I/O 设备，则请输入： Standby_File, Primary File</p> <p>Primary_File 是主磁盘 I/O 设备文件的名称（以及路径）。在字段的这个部分您可以使用路径替代。</p> <p>Standby_File 是辅磁盘 I/O 设备文件的名称（以及路径）。在字段的这个部分您可以使用路径替代。</p> <p>注意：</p> <ul style="list-style-type: none"> • 如果未找到指定的磁盘 I/O 设备文件，Vijeo Citect 将使用指定的文件名创建新的（空）文件。 • 要使用冗余的磁盘 I/O 设备，必须使用支持点对点通信的网络。 • 磁盘文件（在主服务器和辅服务器上）必须拥有写权限。 • 此 Vijeo Citect 向磁盘 I/O 设备写入数据的频率由 [DiskDrv]UpDateTime 参数决定。
I/O 设备协议	<p>要使用 Vijeo Citect 一般协议，请输入：GENERIC</p> <p>- 或者 -</p> <p>要选择 Vijeo Citect 所支持的特定协议，请参见 I/O 设备的在线帮助。</p>
I/O 设备端口名	<p>必须使用：DISKDRV</p>
I/O 设备注释	<p>任何有用的注释。</p>
I/O 设备启动模式	<p>如果未使用冗余的磁盘 I/O 设备，请将此属性保留为空。</p> <p>如果使用冗余的磁盘 I/O 设备，请使用下列两者之一：</p> <p>主： 启用立即使用磁盘 I/O 设备</p> <p>辅写： 此磁盘 I/O 设备将保持不可用，直到配置了主磁盘 I/O 设备的计算机出现不正常。所有发送到主磁盘 I/O 设备的写请求同时也将发送到此磁盘 I/O 设备。</p> <p>要使用辅写模式，必须在主服务器中配置 I/O 磁盘设备。两个 I/O 设备必须有相同的 I/O 设备名称和设备号。</p>
I/O 设备记录写、记录读、缓存和缓存时间	<p>将这些属性保留为空。</p>

另请参见
[冗余的磁盘 I/O 设备](#)

冗余的磁盘（虚拟）I/O 设备

如果使用网络，您可以配置一个冗余的磁盘（虚拟）I/O 设备，以将数据丢失的可能性最小化（在服务器发生故障的事件中）。下图说明了冗余磁盘（虚拟）I/O 设备的使用：



系统运行时，Vijeo Citect 会将运行时数据读写到在主服务器中配置的磁盘（虚拟）I/O 设备。同时，会向在备用服务器中配置的磁盘（虚拟）I/O 设备写入运行时数据。（Vijeo Citect 将保持磁盘（虚拟）I/O 设备相同。）

如果主服务器发生故障，备用服务器中的磁盘（虚拟）I/O 设备将被激活，而不会造成系统的中断。主服务器被激活时，Vijeo Citect 自动地将控件返回到主服务器，并将磁盘（虚拟）I/O 设备从备用服务器复制到主服务器。备用服务器中的磁盘（虚拟）I/O 设备将恢复到备用状态。

定义一个冗余磁盘（虚拟）I/O 设备：

1. 配置一个新的磁盘（虚拟）I/O 设备
2. 在启动模式中选择**辅写**。

对于冗余磁盘（虚拟）I/O 设备，您必须使用 Microsoft 网络或者其他点对点网络，且必须共享备用服务器的硬盘（磁盘（虚拟）I/O 设备文件存放的目录）。使用 Windows Explorer 浏览器设定共享的目录。

另请参见
[I/O 设备属性](#)
[性能考虑](#)

使用内存模式

配置 I/O 设备时，可使用某些选项设置内存模式。这表示 I/O 设备会在内存中创建，并且其参数值在运行时存储在内存中。

有关如何配置 I/O 设备的详细信息，请参阅 [I/O 设备属性](#)。

使用内存模式的设备不会连接到任何硬件，也不会将参数值写入缓存。I/O 设备参数值可由很多过程读取。本地变量与内存模式下的设备之间的区别在于，内存模式下的 I/O 设备会驻留在 I/O 服务器的内存中，并观察所有标准网络和标准 I/O 设备的冗余规则。

第一次配置系统时，内存模式非常有用，因为可以在连接物理 I/O 设备之前设计和测试系统。

针对本地变量，仅使用内存模式的 I/O 设备中的参数值在关机时不会保留。有关本地变量的详细信息，请参阅 [配置本地变量](#)。

另请参见
[I/O 设备属性](#)

故障排除设备通讯

Vijeo Citect 包含一些调试工具，可用于确定系统中的故障。

如果您无法建立与某设备的通讯，请检查以下各项是否出现问题：

- 硬件报警画面
- syslog.dat 文件
- Citect 内核

请参见 [驱动程序错误消息](#)。

硬件报警

Vijeo Citect 操作中出现故障时，生成硬件报警。通常，硬件报警显示在工程中的专用画面上，允许操作员监控系统基础设施的状态。

硬件报警表示是否出现通讯故障、Cicode 是否无法执行、服务器是否出现故障等。如果与设备通讯出现故障，您应首先检查硬件报警画面是否出现故障。

请参见 [硬件报警](#)。

Syslog.dat 文件

syslog.dat 文件在运行时进行维护，并包含系统记录信息，记录信息范围从低级驱动程序通讯和内核消息到用户定义的消息。

在 I/O 设备属性中的 **记录读取** 和 **记录写入** 字段控制是否记录每个 I/O 设备。

Vijeo Citect 在运行时锁定 syslog.dat 文件。但是，用户仍可使用内核中的 SysLog 命令来查看。

请参见 [syslog.dat 文件](#)。

Citect 内核诊断

Citect 内核是在运行时进入 Vijeo Citect 内部工作的重要方法，并具有诊断和调试目的。

内核可以显示多个不同的诊断窗口，每个提供一种 Vijeo Citect 运行时的活动视图。

通过画面驱动程序内核命令启动的 Citect 内核驱动程序窗口可以显示有关 Vijeo Citect 系统中每个驱动程序的信息。它所提供的统计数据包括读取请求、物理读取、每秒数字量读取、寄存器读取、缓存读取、错误计数、超时等。

请参见 [使用 Citect 内核](#)。

错误消息

Vijeo Citect 有两种协议驱动程序错误：

- 通用
- 特定于驱动程序

通用错误是硬件错误 0-31，并且是所有协议共有的。

驱动程序有自己的特定错误，它是唯一的，因此不能被硬件报警系统识别。驱动程序转换特定错误为 I/O 服务器可以识别的通用错误。

例如，当驱动程序有错误时，通常既有驱动程序特定错误也有相应的通用错误。

注意：有关设备驱动程序实施的参考信息，包括驱动程序提醒消息，请参考[驱动程序参考帮助](#)。

另请参见

[一般驱动程序错误](#)

[特定于驱动程序的错误](#)

[使用驱动程序参考帮助](#)

一般错误

下表描述一般协议错误。

错误代码	错误标题	描述
1	地址超界	请求并不存在的设备地址。例如，在 I/O 设备中只有 200 个寄存器时尝试读取寄存器编号 4000。请检查变量标签数据库以查找出错的变量。
2	命令被取消	服务器取消了正在由驱动程序处理的命令。驱动程序可能已经花费了太多时间去处理命令。如果驱动程序未能在指定的时间限制内响应，Vijeo Citect 将取消此命令。时间限制是超时期间和每次超时之后重试命令的次数。你可以在协议的超时和重试参数中更改这些值。还应该检查 WatchTime 参数，以确定驱动程序链接到 I/O 设备的频率。同时检查通信错误。
3	未知的数据类型	请求指定协议不支持的数据类型。该错误应该不会在正常操作中出现。请尝试重启计算机，以重新设置所有驱动程序和硬件。如果问题仍然存在，请联系 Citect 技术支持。如果您已写入了自己的协

错误代码	错误标题	描述
		议驱动程序，此错误是由编译器规格和驱动程序的数据库不匹配引起的。
4	未知的数据格式	写请求包含无效数据，例如，尝试写入具有无效浮点数的浮点地址。检查 Vijeo Citect 数据库。
5	命令未知	服务器发送了一个驱动程序不能识别的命令。该错误应该不会在正常操作中出现。请尝试重启计算机，以重新设置所有驱动程序和硬件。如果问题仍然存在，请联系 Citect 技术支持。
6	响应错误或混淆	通信通道存在的问题，导致了数据传输错误。检查通信通道硬件设置。例如，发送器和接收器之间的奇偶校验、波特率、停止位或数据位可能不匹配。检查 I/O 设备的设置与 Vijeo Citect 端口和板卡表中的域数据是否相同。
7	I/O 设备未响应	I/O 设备未响应读请求或写请求。驱动程序向 I/O 设备发送了一个命令，I/O 设备未在超时期间内响应。这通常是通信故障的第一个显示。检查 I/O 设备与服务器连接是否正确并且已打开。如果超时期间太短也会发生此错误。请尝试增加协议超时参数的超时期间。还可以增加延时参数，从而增加接收响应与发送下一个命令之间的延时时间。
8	一般错误	<p>Vijeo Citect 已经建立了与 I/O 设备之间的通信，然而，I/O 设备检测到协议中存在错误。此错误可能是由通信链接中的故障或梯形逻辑（I/O 设备中）中的错误引起的。</p> <p>解决方法：</p> <ol style="list-style-type: none"> 1. 检查 I/O 设备是否正常运行。 2. 检查通讯电缆是否已正确连接（在两端）。 3. 使用通信快速向导检查 I/O 设备的配置（特别是地址和特殊选项域）是否与推荐的设置和 I/O 设备中的设置相一致。 4. 如果使用的是串行通信，使用通信快速向导检查端口配置（特别是波特率、数据位、停止位和奇偶校验位）是否与推荐的设置和 I/O 设备中的设置相一致。 5. 显示硬件报警画面，注意显示的协议错误。 6. 使用 I/O 设备、网络和通讯板卡随附的文档来查找错误。 7. 检查 I/O 设备中的梯形逻辑是否存在错误。 8. 运行计算机设置向导。 9. 重新编译工程并启动 Vijeo Citect 运行期。
9	写位置受保护	尝试对已进行未授权修改保护的位置执行写操作。要允许写操作，请更改这一位置的访问权限。
10	硬件错误	通信通道、服务器、或 I/O 设备硬件存在问题。检查所有硬件组件。尚未处理命令或数据请求。服务器的运行可能已不再可靠。
11	I/O 设备警告	服务器与 I/O 设备之间的通信链接正常发挥作用，但 I/O 设备有一些警告条件被激活，例如 I/O 设备处于编程模式下。检查 I/O 设备是否处于正确的模式下。
12	I/O 设备离线，无法通信	<p>I/O 设备处于离线模式，不能进行任何外部通信。</p> <p>解决方法：</p> <ol style="list-style-type: none"> 1. 检查 I/O 设备是否正常运行。 2. 检查通讯电缆有无破损。

错误代码	错误标题	描述
		<p>3. 检查通讯电缆是否已正确连接（在两端）。</p> <p>4. 如果使用的是串行通信，检查通信电缆与帮助系统中的图表是否一致。</p> <p>5. 使用通信快速向导检查 I/O 设备的配置（特别是地址和特殊选项域）是否与推荐的设置和 I/O 设备中的设置相一致。</p> <p>6. 如果使用的是串行通信，使用通信快速向导检查端口配置（特别是波特率、数据位、停止位和奇偶校验位）是否与推荐的设置和 I/O 设备中的设置相一致。</p> <p>7. 运行计算机设置向导。</p> <p>8. 检查 <code>citect.ini</code> 文件的以下设置：</p> <pre>[IOSERVER] Server=1 Name=<name></pre> <p>其中：</p> <p><name> 是在 Vijeo Citect 工程中配置的服务器名称。（使用自定义设置检查服务器名称。）</p> <p>9. 重新编译工程并启动 Vijeo Citect 运行系统。</p> <p>注意： 如果您配置了备用 I/O 设备，该错误将导致所有备用 I/O 设备被激活。I/O 设备离线时的命令或数据请求尚未完成。</p>
13	驱动程序软件错误	驱动程序内部发生软件错误。该错误应该不会在正常操作中出现。请尝试重启计算机，以重新设置所有驱动程序和硬件。如果问题仍然存在，请联系 Citect 技术支持。
14	用户访问冲突	未被授权的用户试图访问信息。请检查用户的访问权限。
15	内存耗尽 - 严重	服务器内存耗尽，无法继续运行。最大限缩小缓冲区和队列分配的大小，或扩展服务器计算机的内存。尚未处理命令或数据请求。
16	没有缓冲区，不能继续	没有可以分配的通信缓冲区，或是计算机内存耗尽。服务器性能可能会下降，但它可以继续运行。增加内存。
17	缓冲区不足警告	该错误可能发生在高速瞬间加载期间，没有坏的影响。如果这个错误频繁发生，请增加通信缓冲区的数量。
18	向驱动程序发送了过多命令	向驱动程序发送了过多命令。
19	驱动程序不响应	服务器无法接收来自驱动程序的任何响应。该错误应该不会在正常操作中出现。请尝试重启计算机，以重新设置所有驱动程序和硬件。如果问题仍然存在，请联系 Citect 技术支持。
20	打开的通道过多	每个驱动程序只能支持几个通信通道。您已经超过了这个限制。该错误通常发生在非正常终止并重启服务器时。请尝试重启计算机，以重新设置所有驱动程序和硬件。如果问题仍然存在，请联系 Citect 技术支持。命令或数据请求未完成。
21	通道离线，不能	通信通道当前处于离线状态，禁止了通信。服务器无法初始化通信通道或通道在运行时离线。检查通道硬件是否存在故障。该错误发生时，与此通道连接的所有 I/O 设备都被认为已离线，备用 I/O

错误代码	错误标题	描述
	通讯	设备被激活。命令或数据请求未完成。
22	通道尚未被打开	服务器试图与未打开的通道通信。请尝试重启计算机，以重新设置所有驱动程序和硬件。如果问题仍然存在，请联系 Citect 技术支持。命令或数据请求未完成。
23	通道尚未被初始化	服务器试图与尚未初始化的通道通信。该错误应该不会在正常操作中出现。请尝试重启计算机，以重新设置所有驱动程序和硬件。如果问题仍然存在，请联系 Citect 技术支持。命令或数据请求未完成。
24	每个通道的 I/O 设备数过多	通道具有太多与之相连的 I/O 设备。该错误应该不会在正常操作中出现。命令或数据请求未完成。请尝试重启计算机，以重新设置所有驱动程序和硬件。如果问题仍然存在，请联系 Citect 技术支持。
25	数据尚未有效	被请求的数据仍在被处理，将在适当的时候返回。此错误只发生在必须建立复杂的通信以从 I/O 设备中检索数据的驱动程序中。请忽略此警告。
26	无法取消命令	服务器试图取消某个命令，但驱动程序找不到该命令。该错误应该不会在正常操作中出现。请尝试重启计算机，以重新设置所有驱动程序和硬件。如果问题仍然存在，请联系 Citect 技术支持。
27	备用 I/O 设备被激活	通讯已经从主 I/O 设备切换到备用 I/O 设备。服务器会在出现“紧急”转换时返回此消息。修正主 I/O 设备中的故障。
28	消息溢出	响应的长度超过了响应缓冲区。如果该错误发生在串行通信驱动程序中，可能会收到乱码。检查通讯链接和驱动程序的波特率。
29	错误的用户参数	存在配置错误，例如设置了无效的特殊选项。
30	备用 I/O 设备错误	备用 I/O 设备中存在错误。修正备用 I/O 设备中的错误。
31	I/O 服务器请求超时	一个或多个发送到 I/O 服务器的请求未在超时期间内完成。I/O 服务器离线或 I/O 服务器用于完成请求的时间过长。请检查 PLC 通信链接、PLC 超时、PLC 重试次数和网络通信。即使您没有网络（即，如果 I/O 服务器与控制客户机是同一台计算机），此错误也可能发生。如果错误仍然存在，请增加 [LAN] TimeOut 参数。缺省超时值为 8000 毫秒。

协议特定错误

尽管每一个协议都有多种唯一的错误，但最初的 34 个协议特定错误是所有协议的标准。报告的所有协议特定错误的错误号必须在 1 到 31。尽管这些错误有自己的错误号（以 16 进制显示），但它仅作为一个符号使用。

注意： 协议特定错误显示在每个协议的协议特定错误帮助主题下。如果您不能确定错误种类，请参考 I/O 设备随附的文档。

错误代码	错误标题	描述
1 (0x01)	处理接收字符的速度不够快	处理接收字符的速度不够快。波特率较低或使用计算机的速度较快。如果错误持续存在，请联系 Citect 支持。
2 (0x02)	奇偶校验错误	接收到的信息有奇偶校验错误。检查在 Citect 端口表中是否指定了正确的波特率、奇偶校验、停止位和数据位。此错误可能是由连接到 I/O 设备的电缆故障所引起，也可能是由通讯链接的额外噪音所引起。
3 (0x03)	检测到接收线路中断	检测到接收线路发生中断。此错误可能是由连接到 I/O 设备的故障电缆所引起，也可能是由通讯链接的额外噪音所引起。
4 (0x04)	帧错误	指定的波特率可能错误。检查是否在 Citect 端口表中定义了正确的波特率。
5 (0x05)	消息太长	从 I/O 设备中接收的消息太长。此错误可能是由连接到 I/O 设备的故障电缆所引起，也可能是由通讯链接的额外噪音所引起。如果错误持续存在，请联系 Citect 支持。
6 (0x06)	校验和无效	接收到的消息中的校验和与计算值不符。检查在 Citect 端口表中是否指定了正确的波特率、奇偶校验、停止位和数据位。此错误可能是由连接到 I/O 设备的电缆故障所引起，也可能是由通讯链接的额外噪音所引起。您也可以尝试增加协议的重试参数数值。
7 (0x07)	正文起始丢失	接收到的消息中没有正文起始 (STX) 字符。检查在 Citect 端口表中是否指定了正确的波特率、奇偶校验、停止位和数据位。此错误可能是由连接到 I/O 设备的电缆故障所引起，也可能是由通讯链接的额外噪音所引起。
8 (0x08)	正文结束丢失	接收到的消息中没有正文终止 (ETX) 字符。检查在 Citect 端口表中是否指定了正确的波特率、奇偶校验、停止位和数据位。此错误可能是由连接到 I/O 设备的电缆故障所引起，也可能是由通讯链接的额外噪音所引起。
10 (0x0A)	不能传输消息	Vijeo Citect 不能传输消息。此错误可能是由连接到 I/O 设备的故障电缆所引起，也可能是由通讯链接的额外噪音所引起。

错误代码	错误标题	描述
		起。
11 (0x0B)	不能复位串行驱动程序	串行驱动程序 (COMxI、PCXI 或 COMx) 发生故障。请尝试重启计算机, 以复位所有驱动程序和硬件。
12 (0x0C)	请求长度不一致	请求长度与驱动程序的要求不一致。
15 (0x0F)	服务器命令无效	服务器命令无效。请联系 Citect 支持。
16 (0x10)	不能为驱动程序分配计时器资源	不能为驱动程序分配计时器资源。请联系 Citect 支持。
17 (0x11)	为驱动程序指定了太多通道	为设备指定了太多通道。请联系 Citect 支持。
18 (0x12)	服务器的通道号不能打开	服务器的通道号不能打开。请联系 Citect 支持。
19 (0x13)	命令不能取消	驱动程序命令不能被取消。请联系 Citect 支持。
20 (0x14)	通道离线	通道离线。如果超时, 可能发生这个错误。此错误可能是由连接到 I/O 设备的故障电缆所引起, 也可能是由通讯链接的额外噪音所引起。
21 (0x15)	超时错误	在指定的超时时间内没有收到来自 I/O 设备的响应。此错误可能是由连接到 I/O 设备的电缆故障所引起, 也可能是由通讯链接的额外噪音所引起。您可以增加协议重试参数的数值。
22 (0x16)	来自服务器的 I/O 设备号不能激活或超界	来自服务器的 I/O 设备号不能激活或超界。请联系 Citect 支持。
23 (0x17)	I/O 设备离线	检查在 Citect I/O 设备表中指定的 I/O 设备地址是否与在 I/O 设备中配置的相同。
24 (0x18)	服务器的数据类型不能被驱动程序识别	驱动程序不能识别服务器的数据类型。请联系 Citect 支持。
25 (0x19)	服务器的 I/O 设备类型不能被驱动程序识别	服务器的 I/O 设备类型不能被驱动程序识别。请联系 Citect 支持。
26 (0x1A)	为通道指定的 I/O 设备太多	为通道指定的 I/O 设备太多。请联系 Citect 支持。
27 (0x1B)	向驱动程序发送的命令太多	向驱动程序发送的命令太多。请联系 Citect 支持。

错误代码	错误标题	描述
28 (0x1C)	读取的数据无效	读取的数据无效。请联系 Citect 支持。
29 (0x1D)	命令被取消	驱动程序命令被取消。请联系 Citect 支持。
30 (0x1E)	地址无效或超界	试图访问的地址的数据类型无效或超界。检查您使用的数据类型或地址范围是否对 I/O 设备有效。
31 (0x1F)	来自服务器的数据长度不正确	来自服务器的数据长度错误。请联系 Citect 支持。
32 (0x20)	不能从设备读取数据	Vijeo Citect 不能读取来自 I/O 设备的数据。请联系 Citect 支持。
33 (0x21)	指定的设备不存在	指定的设备不存在。请联系 Citect 支持。
34 (0x22)	指定的设备不支持中断	指定的 I/O 设备不支持中断处理。您已经在板卡表上或通过 PollTime 参数设置为 0 的方式，为不支持中断的硬件设备指定了一个中断。检查是否在板卡上设置了中断，或设置了协议的 PollTime 参数。

调试 I/O 设备和协议

在试运行任何系统前，必须测试 Vijeo Citect 和 I/O 设备间的所有通讯。许多人将此操作留在最后，只为发现他们不能解决的通讯问题。

以下信息旨在鼓励用户在通讯成为工作中的高时间要求因素前对其进行完整测试。它还将帮助您自行调试通信和协议状况。

另请参见

[创建通讯测试工程](#)

[调试 COMx 驱动程序](#)

[调试 TCP/IP 驱动程序](#)

[调试使用串行通讯的协议驱动程序](#)

[调试专用板卡驱动程序](#)

[联系 Citect 支持](#)

调试 COMx 驱动程序

COMx 驱动程序是用于大多数串行通讯的板卡驱动程序。可利用 2.01 版 COMx 驱动程序转储调试信息。针对每个 com 端口将产生三个文件：写文件、读文件和状态文件。调试文

件通过 `citect.ini` 中的设置进行配置，被写入，然后被写入在 `[CtEdit]Logs` 中指定的路径。使用以下 `citect.ini` 条目：

```
[COMx]
WritePortName=X1,X2...
WriteDebugLevel=Y
WriteFileSize=Z
ReadPortName=X1,X2...
ReadDebugLevel=Y
ReadFileSize=Z
StatusPortName=X1,X2...
StatusDebugLevel=Y
StatusFileSize=Z
```

其中：

Xn =	出现在 Citect 通讯 端口表格中的端口名称。例如，PORT_1。
Y =	1 可启用调试，0 可禁止调试。
Z =	以 Kb 为单位的文件大小（缺省值为 1000 Kb）。

范例

```
[COMx]
WritePortName=PORT_1,PORT_2
WriteDebugLevel=1
WriteFileSize=2000
ReadPortName=PORT_1
ReadDebugLevel=1
ReadFileSize=1000
StatusPortName=PORT_2
StatusDebugLevel=1
StatusFileSize=10
```

以上范例将：

- 将由 Vijeo Citect 驱动程序发送到 PORT_1 和 PORT_2 的最多 2000Kb 的数据记录到“写”文件；
- 将由 Vijeo Citect 驱动程序从 PORT_1 接收到的 1000Kb 的数据记录到“读”文件；
- 将最多 10 Kb 的状态数据从 PORT_2 记录到“状态”文件。

这还将导致下列文件被创建：

- WPORT_1.dat
- WPORT_2.dat
- RPORT_1.dat
- SPORT_2.dat

通常情况下，文件名的格式为“R”、“W”或“S”加请求的端口名称加“.dat”。文件名代表相应的“读”、“写”和“状态”文件。

文件格式

写文件会采用以下格式：

LINE 1	WRITE Debug file Started PORTNAME Debug Level 1 DOW MONTH DOM HH:MM:SS.sss
LINE 2	HH:MM:SS.sss
LINE 3	Out W In X nBytes Y Status Z
LINES 4...	AA BB CC ..

其中：

W & X =	缓冲区指针值
Y =	写入的字节数
Z =	WriteFile 的返回状态
AA BB CC =	写入的每个字节的值（十六进制）

范例：

```
WRITE Debug file Started PORT1_BOARD1 Mon Dec 15 16:07:07.998
16:07:09.810
Out 0 In 8 nBytes 8 iStatus 997
0e 02 00 00 00 10 00 00
16:07:10.802
Out 8 In 16 nBytes 8 iStatus 997
0e 02 00 00 00 10 00 00
..
```

读文件会采用以下格式：

LINE 1	READ Debug file Started PORTNAME Debug Level 1 DOW MONTH DOM HH:MM:SS.sss
LINE 2	HH:MM:SS.sss
LINE 3	Out W InRx X nBytes Y Status Z

```
LINES 4...   AA BB CC ..
```

其中：

W & X =	缓冲区指针值
Y =	读取的字节数
Z =	保留在缓冲区中的字符数
AA BB CC =	写入的每个字节的值（十六进制）

范例

```
READ Debug file Started PORT1_BOARD1 Mon Dec 15 16:07:07.998
16:07:09.830
Out 0 In 0 nBytes 8 iStatus 0
0e 02 00 00 00 10 00 00
16:07:10.822
Out 0 In 8 nBytes 8 iStatus 0
0e 02 00 00 00 10 00 00
```

状态文件会采用以下格式：

LINE 1	STATUS Debug file Started PORTNAME Debug Level 1 DOW MONTH DOM HH:MM:SS.sss
LINE 2	HH:MM:SS.sss
LINE 3	modemStatus X

范例

```
STATUS Debug file Started PORT_1 Debug Level 1 Wed Nov 05
15:28:55.310
15:29:55.950
modemStatus 34
..
```

以后还可能添加更多参数，所以请定期检查 COMx 信息和 Citect 知识库。

另请参见
[调试 TCP/IP 驱动程序](#)

调试 TCP/IP 驱动程序

TCP/IP 驱动程序是可用于任何 TCP/IP 通讯的低层驱动程序。它可以通过以太网、令牌环或 Arcnet 传送。此驱动程序在 Vijeo Citect 和 Winsock 间通讯，因此，它实际上可以使用 Windows 的所有正常联网功能。用于 TCP/IP 的参数有：

[TCPIP]

- Log - Dumps all traffic between the Vijeo Citect TCPIP.DLL and Winsock to a text file called TCPIP.DAT in your Logs directory. The size of the log is governed by the LogSize parameter
 允许的值
 1- 启用记录
 0 - 禁用记录
 缺省值 - 0
- LogSize - 设置 TCP/IP 驱动程序的日志文件在结束之前的缺省最大大小（以 Kb 为单位）。
 允许的值
 任何整数
 缺省值 - 200
- PortName - 日志的端口名称。
 缺省值 - *, 缺省跟踪所有端口 ("*")。
- LogTxRx - 除了设置信息之外的日志实际数据 Tx & Rx。
 允许的值
 1- 启用对 Tx & Rx 的记录
 0 - 禁用记录
 缺省值 - 0
- NoBufferSleepTime - 在获取 WSAENOBUFFS（无系统缓存消息）之后、读取另一缓存之前的等待休眠时间（毫秒）。您应不再需要对此值做调整。
 缺省值 - 0

调试步骤

要进行调试，请执行以下步骤：

1. 检查能否 PING 通目标 I/O 设备。如果不能，Vijeo Citect 将不能与其对话。为此，打开一个命令窗口，键入 **PINGaaa.bbb.ccc.ddd**，其中 a.b.c.d 是试图连接到的 I/O 设备的 IP 地址。如果 PING 无效，需要返回 Windows 网络连接并解决此问题。
2. 坚持使用尽可能简单的工程 (SAPP)。
3. 删除所有旧的 tcpip.dat 文件。
4. 在 citect.ini 文件中设定 Debug=1 和 Log=1 参数。
5. 开始工程。通过显示在内核和 TCP/IP 最大化主窗口中的信息，应该可以看到 Vijeo Citect 是否正在向您的 I/O 设备发送要求初始化二者之间通讯的请求。

如果没有请求被发出，则软件没有被正确配置，您应检查在 Vijeo Citect 启动时是否没有错误。如果启动时有错误，可以从在线帮助中查找这些错误。同时检查您的计算机是否是 I/O 服务器（并与工程中的某个服务器匹配）。为此，运行计算机设置向导，然后对计算机进行单机配置。

如果没有请求，您将看到 Vijeo Citect 和 Winsock 之间的所有通讯，该窗口显示 Vijeo Citect 发出的连接到 I/O 设备的请求和来自 I/O 设备的相应响应。任何错误都具有一个 Winsock 错误代码，可以从 Microsoft 知识库中查找该代码。如果看到**连接成功**消息，Vijeo Citect 应该可以进入在线状态。

另请参见

[调试使用串行通讯的协议驱动程序](#)

调试使用串行通讯的协议驱动程序

要调试使用串行通讯的协议驱动程序，请执行以下操作：

1. 坚持使用尽可能简单的工程 (SAPP)。
2. 为所有协议设置 DebugStr=*。
3. 备份并删除 syslog.dat 和 syslog.bak 文件。系统会在 Vijeo Citect 下次启动时重新创建此文件的干净版本。
4. 开始工程。通过显示在最大化主内核窗口中的信息，应该可以看到 Vijeo Citect 是否正在向您的 I/O 设备发送要求初始化二者之间通讯的请求。

如果没有请求被发出，则软件没有被正确配置，您应检查在 Vijeo Citect 启动时是否没有错误。如果启动时有错误，可以从在线帮助中查找这些错误。同时检查您的计算机是否是 I/O 服务器（并与工程中的某个服务器匹配）。为此，运行 *计算机设置向导*，然后对计算机进行单机配置。

如果有请求发出但没有回应，那么 Vijeo Citect 会尝试通讯。如果 Vijeo Citect 正在发送请求但得不到回应，最常见的原因有：

- **请求 Vijeo Citect 未到达 I/O 设备** - 检查 I/O 设备表中的地址域，确认其正确无误。如果 I/O 设备需要唯一标识符（如节点地址），或需要某些类型的路由路径，然后确认其正确无误。
检查端口表中的参数与和 I/O 设备使用的参数是否相同。如果您有 8 个数据位而 I/O 设备使用 7 个数据位，通讯将中断。
检查电缆工作是否正常。最简单的检查方法是创建一个新工程并使用回送协议。您可以使用此方法验证 Tx 和 Rx 线的完整性，方法是在这些线上放置一个跳线。首先用 PC 上针 2 和 3 之间的跳线进行此测试。然后将电缆电源插上，再次用 Tx 和 Rx 线之间的跳线进行测试。移动跳线直至其到达通讯总线的末端。您可在 Citect 知识库中找到更多有关使用回送协议的信息。
即使回送协议未显示错误，您的电缆仍可能造成通讯丢失。Vijeo Citect 通常会在串行通讯中放置大量恒定负荷，其数量远远大于编程软件放置的恒定负荷，这通常表明 Vijeo Citect 将请求比编程软件严格得多的握手。因此，用于对 I/O 设备编程的电缆适用于编程，但不适用于 Vijeo Citect。检查帮助中协议的接线图。

另一个引起电缆不正确连接的主要原因是管脚 9 到 25 间的转换器。许多此类转换器是专门针对串行鼠标制作的。它们通常只使用 **Tx**、**Rx** 和 **Ground** 信号。如果使用这些转换器中的一个，它们根本不支持握手，并很有可能不适用于您的协议。

如果以上检查未发现问题，使用 COMx 的参数（如上所述）创建日志文件。检查这些日志文件，并确认 Vijeo Citect 实际发送的数据就是要发送的数据。通过使用这些参数生成的日志文件从低于 Vijeo Citect 的层次获取其信息，并准确显示通过 COMx 驱动程序的信息。

- **来自 I/O 设备的响应未到达 Vijeo Citect** - 这很少发生，通常是由于电缆被损坏、带宽不够或连接不正确所引起。按上诉方式检查您的电缆。同时检查所需内容是否都是在 Vijeo Citect 范围内。许多协议要求 Vijeo Citect 在其请求包中发送一个唯一标识符。如果此标识符不正确的，响应不会返回 Vijeo Citect。
- **I/O 设备不理解请求** - 所有 Vijeo Citect 协议都可能会检查 I/O 设备是否正在运行。协议一般会尝试从 I/O 设备（通常是一个状态寄存器或应该位于此处的其它寄存器）中读取数据。但是，许多非标准协议（如 Modbus）并不完全符合该协议的规范。随 Vijeo Citect 一起提供的许多协议具有一些额外参数，可以通过这些参数选择来自 Vijeo Citect。这些可从在线帮助或知识库中找到。与 I/O 设备制造商核洽，确认该 I/O 设备可以响应 Vijeo Citect 发出的请求。如果不确定此请求是针对初始化发送的，使用 DebugStr=* 获取 Vijeo Citect 在其初始化中请求的实际变量地址。
检查使用的协议。Vijeo Citect 可能有很多不同的用于与 I/O 设备通讯的协议。PLC（如 AB PLC5）可以使用不同的串行协议，具体取决于您尝试使用的方法。确保使用的方法正确无误。如果不确定，可尝试其它协议。
- **I/O 设备未正常工作** - 可以使用 I/O 设备制造商提供的某些类型的软件来诊断 I/O 设备的问题。

另请参见

[调试专用板卡驱动程序](#)

调试专用板卡驱动程序

专用板卡驱动程序（如 Allen Bradley KTX 卡、Modicon SA85 卡、Siemens TIWAY 卡等）具有自己的低层驱动程序。每个驱动程序都有调试参数，这使设备行为调试变得简单。检查 Vijeo Citect 知识库和帮助中查找参数。知识库中包含描述这些板卡驱动程序的调试方法的文章。

调试过程与串行连接完全相同：

1. 坚持使用尽可能简单的工程 (SAPP)。
2. 为协议和板卡驱动程序设置任何调试参数。
3. 利用清洁的日志文件启动 Vijeo Citect。
4. 找到所有错误，然后在制造商的文档中查找这些错误。

串口回环测试

串口回环测试用于测试你的串行硬件配置。此测试能用于任何 COM 端口，无论它是本地的，或是多串口的串行板卡（诸如数据采集板）。此测试可以在回环电缆的内部或外部完成。

另请参见

[测试设置](#)

[串行口回环电缆](#)

测试设置

1. 不应该配置其它的协议。执行此测试期间，暂时删除其它板卡和单元。
2. 为每一个要测试的端口配置一个单元。I/O 设备表应具有如下外观：
 - **名称：** <I/O 设备的唯一名称>
 - **号码：** <I/O 设备的唯一网络号码>
 - **地址：** 无
 - **协议：** LOOPBACK
 - **端口名称：** <端口表中的“端口名称”>
3. 支持以下 Citect.ini 选项：
 - [LOOPBACK]

LoopBack - 如果要执行内部回环测试将其设置为 1（确保运行测试后可以将其删除）。当设为 0 时，在每一个端口，需要一个 2 和 3 管脚连在一起的回环连接器。

注意： COMx 驱动程序不支持内部回环测试。外部回环测试是缺省模式。

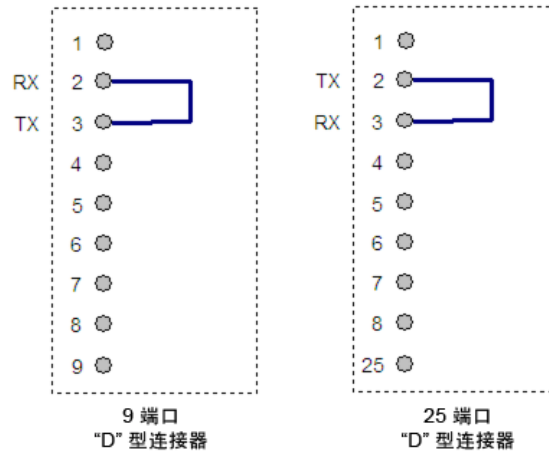
Size - 设置最大帧尺寸。每一个已传输帧的长度是随机的，介于 1 和 -1 之间。默认大小为 512。

4. 启动 Vijeo Citect。每个端口将传输一个随机长度的帧。当完整的帧被接收时，这个过程将重复。
5. 打开内核，键入“页面驱动程序”并按 **Enter**。键入 **V** 以将显示模式设置为“verbose”。此时将显示以下统计数据：
 - 传输的字符数。
 - 传输的帧数。
 - 接收的字符数。
 - 经过的时间（以毫秒计）。
 - 每秒接收的字符数。
 - 开始时间（以毫秒计）。
 - 错误总数。
 - 最后检测到的错误的错误代码。

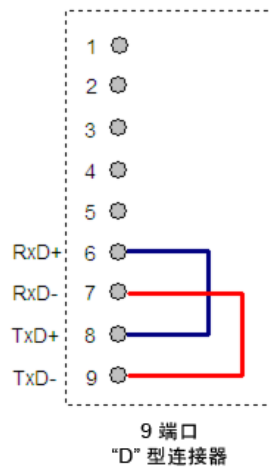
注意：错误总数应该是 0。如果报告的错误数不是零，则您的串行硬件工作不正常。

串行口回环电缆

下图显示使用 RS-232 的回环连接。



下图显示使用 RS-422 或 RS-485 的回环连接。



联络技术支持部门

如果在合理的时间内未能调试成功，请联络技术支持部门，并提供下列信息：

- 您可以通过技术支持部门获得一个解决方案申请。
- syslog.dat, syslog.bak, tcpip.dat 或者 COMx 等日志文件。
- 一份 SAPP 的拷贝，以及 citect.ini 文件。

参见 [获得技术支持](#)。

性能考虑

许多外部因素影响控制和监视系统的性能。计算机的性能、I/O 设备以及它们之间的通信路径是明显的因素。它们传输数据越快，你的系统运行起来就越快。(Vijeo Citect 始终会保持与 I/O 设备硬件能够支持的数据传送速率。) 数据传输速率取决于硬件：一旦安装了此硬件，则影响此特性的能力受到限制。

不过，您可以在一个方面对运行时系统的性能直接施加影响：I/O 设备中数据寄存器的配置。

另请参见

[缓存数据](#)

[将寄存器分组](#)

[在 I/O 设备中重映射变量](#)

缓存数据

在具有众多显示客户机的大型联网系统上，可通过使用内存缓存缩短通讯往复时间。

缓存处于启用状态时，从 I/O 设备读取的所有数据都被临时存放在 I/O 服务器的内存中。如果在缓存时间内（从相同或不同客户机）再次请求相同数据，则 Vijeo Citect I/O 服务器将返回其内存中的值，而不是再次读取该 I/O 设备。当许多客户机需要相同数据时，数据缓存能够在总体上提高响应速度。

建议将缓存时间设置为 300 毫秒。应避免占用很长的缓存时间（不超过 1000 毫秒），因为这样可能对按时向系统提交信息有不利影响。

注意：请不要将数据缓存用于 [磁盘 I/O 设备](#)，因为它们自带缓存。

数据缓存如何工作

数据缓存可以防止短时间内重复读取 I/O 设备数据。当不止一个客户机在短时间内（通常为 300 毫秒）请求 I/O 服务器从 PLC 或相同 I/O 设备读取数据时，不必要的重读就产生了。

正常情况下，根据客户机的请求，I/O 服务器从 I/O 设备中读取状态数据，并把它返回给发出请求的客户机。

如果服务器在将原始数据返回给第一个客户机之前又收到另一客户机的请求，它就通过自动发送原始数据给所有的请求客户机以优化读取操作。（画面通用块读取会显示这个数值）。

如果在服务器返回数据给客户机后客户机又立即发出相同的请求，服务器就会不必要地重读设备。

将数据缓存时间设置为 300 毫秒（或接近），则会防止在此缓存时间段内发生重读。如果服务器已将数据发送给较早的客户机，在 300 毫秒之内，如有其他客户机向相同服务器请求相同数据，服务器上的缓存数据将会立刻响应发送给后续请求。

注意：多个客户机不必是网络中独立的 Vijeo Citect。它们可以是同一计算机中的报警和趋势客户机，因此此优化也将影响单节点系统。

此外，Vijeo Citect 还可以使用预读缓存。当缓存中的数据已陈旧时（接近缓存时间），I/O 服务器将从 I/O 设备中再次请求。这就优化了将再次被使用的数据的读取速度（频繁）。为了给其他读取请求以更高的优先权，如果 I/O 设备的通信通道空闲，I/O 服务器将只请求该数据。

永久保留数据记录

要长期保留缓存设备数据的一份副本，可以将 I/O 服务器的缓存保存到磁盘上。对于每个 [IOServer]SavePeriod，数据将保存到多个持续缓存，每个缓存设备一个。

大多数情况下，将数据保存到磁盘上可允许您在关闭和重启 I/O 服务器时不必再访问每个 I/O 设备以获得当前值。相反，您可以从设备的持久缓存中读取这些值。

注意：如果对远程或预定 I/O 设备启用了完全读缓存，则在激活的 I/O 服务器与该设备断开时，该设备的持续缓存会被保存到磁盘上。无论在 [IOServer]SavePeriod 中设置的值如何，都会发生这样的情况。可通过设置参数 [Dial]ReadThroughCache 启用完全读缓存。

另请参见
[将寄存器分组](#)

将寄存器分组

配置 Vijeo Citect 系统时，必须定义 Vijeo Citect 要在系统运行时读取的每一变量（寄存器地址）。当运行系统工作时，Vijeo Citect 会计算最有效的寄存器读取方法。Vijeo Citect 根据 I/O 设备的类型和寄存器地址优化通信。

Vijeo Citect 从 I/O 设备请求数据时，寄存器的值不会立即返回；将出现系统开销。系统开销（与协议标题、校验、设备反应时间相关联）取决于 I/O 设备的品牌，通常是读取单个寄存器所需时间的几倍。因此，单独读取寄存器效率很低，Vijeo Citect 通常读取连续的寄存器块。因为系统开销仅出现一次（在发出最初请求时），块中的所有寄存器共享此开销，这提高了数据传送的总体效率。

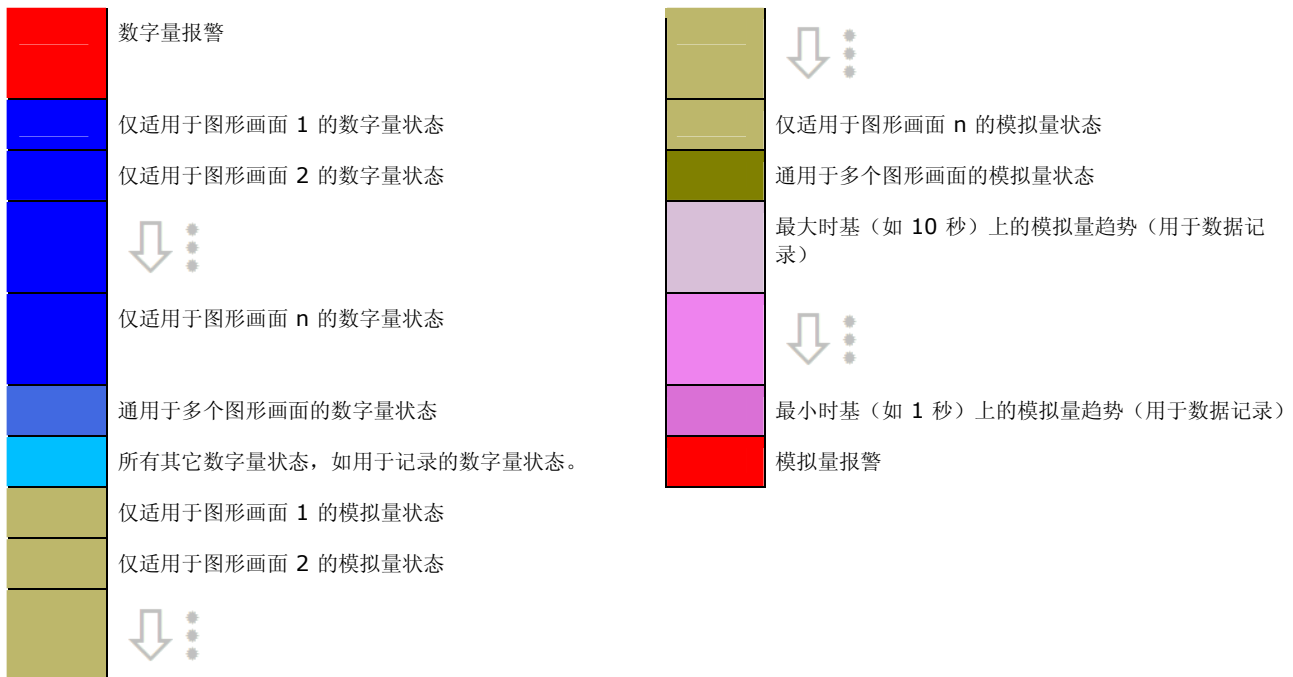
然而，仅在寄存器块的一小部分被实际使用时读取该块效率也很低。如果您的 Vijeo Citect 系统要读取的寄存器遍布 I/O 设备的整个内存，将需要大量通信。Vijeo Citect 必须读取许多连续的块（放弃未使用的寄存器）或单独读取寄存器，这将降低系统的性能级别。可以通过将 Vijeo Citect 要读取的寄存器分组来避免此情况。

连续读取与报警关联的所有寄存器。（如果出现报警条件，Vijeo Citect 可以立即显示报警。）因此，应该将所有指示报警条件的寄存器分组。

与状态显示（对象、趋势等）关联的寄存器仅在需要时被读取（即关联图形画面显示时），并会根据它们显示其中的画面进行适当分组。

用于数据记录的寄存器以您定义的频率被读取。系统会根据它们的读取频率对其进行分组。

下表所示为 Vijeo Citect 系统的理想寄存器分组：



当内存空间受限或现有 PLC 程序可能加入限制时，将寄存器分组到单独的块（即使它们并非连续的块）可提高系统的性能。

设计系统时，可以在每个块的末端留几个备用寄存器以备今后改进之用。

另请参见：

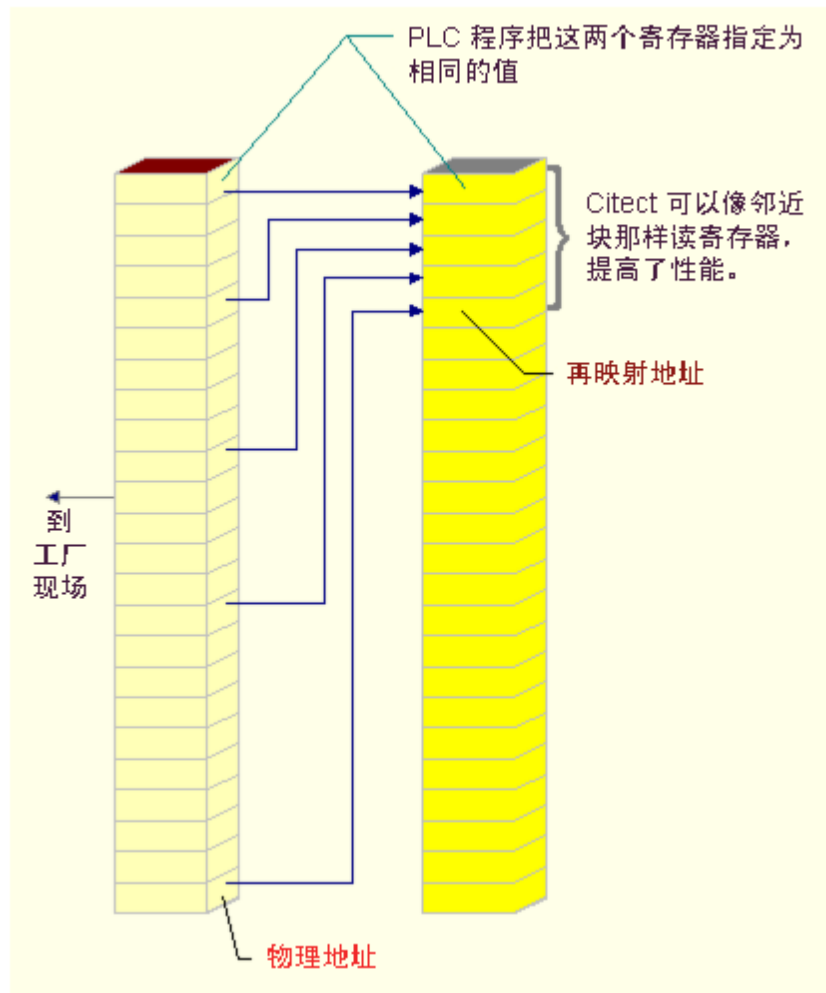
[在 I/O 设备中重映射变量](#)

在 I/O 设备中重映射变量

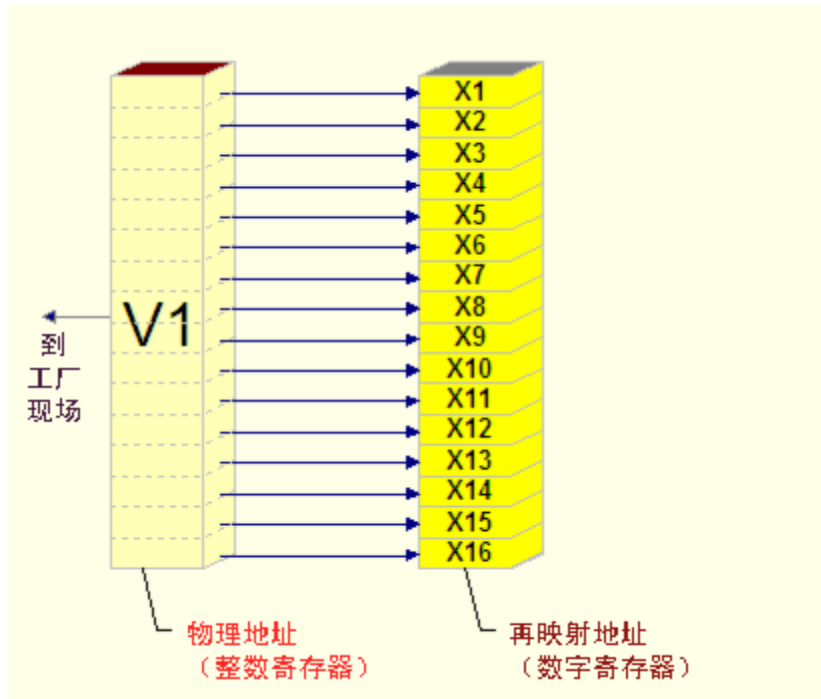
一些 PLC 允许您将一个 I/O 设备变量重映射（或复制）到另一个寄存器地址。Vijeo Citect 允许您重映射，以：

- 分组寄存器，更加有效地提高性能。
- 允许 Vijeo Citect 解释变量类型（如模拟量变量）为不同的变量类型（例如，数字量变量）。例如，如果一个 I/O 设备运行超出了数字量地址的范围，您可以创建一个附加的数字量地址。

要重映射 PLC 中的变量，您必须设计（或修改）PLC 中的逻辑，以关联两个地址。然后，Vijeo Citect 就可以从重映射的地址（而不是物理地址）读取变量，或将变量写入该地址。



您也可以再分配一种类型的变量（例如，整数）到另一种类型的变量（例如，数字量变量）。



要在 Vijeo Citect 中重映射，首先要在工程中创建变量。然后，可以设置重映射，指定地址在期望范围内的变量将被重映射。I/O 服务器按照重映射指令可以在运行时重新引导命令。

注意： 不是所有的 PLC 和/或 Vijeo Citect 驱动程序都支持重映射。除非需要，否则不建议使用。如果您需要了解 PLC 和/或驱动程序是否支持重映射，请联系 Schneider Electric SA 技术支持。

如何在 Vijeo Citect 中重映射一个变量：

1. 打开“工程编辑器”。
2. 选择**通讯 | 重映射**。将显示“重映射”对话框。

选项	描述
Vijeo Citect 变量	<p>第一个在变量标签数据库中定义的重映射（Vijeo Citect）变量（使用“标签”对话框），例如：Motor_1_Run.</p> <p>（最多 79 个字符）。或者，您可以使用指令<单元名称> <地址> 格式（使用特定于您的 I/O 设备的值），例如：IODev X1 .</p> <p>在此输入的地址将被重映射。在运行时，I/O 服务器将通过物理地址读取/写入数据。</p>

选项	描述
长度	重映射变量的数量。Vijeo Citect 将读取充足的物理变量，以重映射这么多数量的变量（最多 10 个字符）。 长度必须小于协议的最大请求长度。协议概述显示协议的最大请求长度。
物理变量	PLC 中的第一个物理变量，例如：ReMapIntV7。（最多 79 个字符）。 此变量不需要在变量标签数据库中定义。您可以使用 <单元名称> <地址> 格式（使用特定于您的 I/O 设备的值）。例如：IODev V7 。
重映射读取	确定是否要执行重映射读取。设置为 TRUE 或 FALSE。 FALSE - 读取正常（不重映射）变量。Vijeo Citect 变量的实际地址将直接从 I/O 设备中读取，而不是从物理变量中。（如果您的 I/O 设备不支持重映射读取，可使用此模式。） TRUE - 读取重映射变量（通过物理变量）。
重映射写入	确定是否要执行重映射写入。设置为 TRUE 或 FALSE。 FALSE - 写入正常（非重映射）变量。Vijeo Citect 变量的实际地址将直接写入到 I/O 设备中，而不是物理变量中。（如果您的 I/O 设备不支持重映射写入，可使用此模式。） TRUE - 写入重映射变量（通过物理变量）。
注释	任何有用的注释（最多为 48 个字符）。

3. 完成此对话框，然后点击**添加**。

注意： 要确定设备是否支持重映射读写，可参见 I/O 设备数据类型帮助。

另请参见
[重映射示例](#)

重映射示例

将 CCM 协议用于 GE 9030 PLC，当读取某些数字注册值时，下面的重映射可以被用于优化通讯。这个示例以 PLC 已正确设置了重映射为基础。

变量标签数据库

数字注册值被定义如下：

变量标签名	Motor_1_Running_Feedback
数据类型	数字量

I/O 设备名	Area1
地址	M1
变量标签名	Motor_1_Overload
数据类型	数字量
I/O 设备名	Area1
地址	M3
变量标签名	Motor_4_Running_Feedback
数据类型	数字量
I/O 设备名	Area1
地址	M4
.	
.	
.	
变量标签名	Motor_4_Running_Feedback
数据类型	数字量
I/O 设备名	Area1
地址	M16

注意地址 M2 没有使用。跳过地址不会影响性能。

重映射数据库

重映射的定义如下：

Vijeo Citect 变量	Area1 M1
长度	16
物理变量	Area1 R10
重映射读取	True

重映射写入

False

物理变量的数据类型为整数，它并不需要在变量标签数据库中定义（但可以定义）。

高级驱动程序信息

本部分提供有关驱动程序的高级信息。

另请参见

[变量（数字）限制](#)

[验证为基于标签的驱动程序分发的工程数据](#)

[写延迟问题](#)

变量（数字）限制

设备通常有指定数据类型的内存区域，象字节，整数或字。一些协议并不支持使用不同的数据类型读或写这些内存区域。当读或写字节，整数和字的单个位时，这种情况是很普遍的。

在这种情况下，读取这些较大数据类型中的单独的位就通过读取指定的数据类型并且让 Vijeo Citect 驱动程序将它划分成单独的位来完成。写较大数据类型的位就比较复杂，因为写较大数据类型的一位同时就会覆盖相同数据类型的其它位。为了防止在写新的位值时改写现有的位，一个“读取-修改-写入”事件可以用来在较大数据类型中写一位。使用这种方式，Vijeo Citect 驱动程序将读这个较大数据类型，修改较大数据类型中的适当位，然后将较大数据类型写回到设备中。

在有必要使用“读取-修改-写入”方法来避免改写较大数据类型的寄存器中的现有位时，如果被写入的设备同样被配置或编程成修改这些相同的寄存器，它可能产生问题。例如，如果 PLC 设备在 Vijeo Citect 驱动程序读取了这些检测器之后但在 Vijeo Citect 写入修改值之前修改其较大数据类型之一的寄存器，则所做更改将被改写。如果 Vijeo Citect 和使用这些数据类型的任何设备被配置成在任何给定时间都只能有唯一一个拥有写访问权，则可避免这种结果。

这一“读取-修改-写入”方法有一个严重的操作关切：如果在 Vijeo Citect 驱动程序读取了较大数据类型之后但在 Vijeo Citect 写入新值之前设备修改了较大数据类型，任何由设备所做的更改将被覆盖。这个问题在控制系统中可能是很严重的，因此要求设备和 Vijeo Citect 必须被配置，以至于这些系统中只有一个写这种数据类型。



警告

不希望的设备操作

如果要使用“读取-修改-写入”方法来更改某个数据类型的位值，在配置系统时应使 Citect 和主设备不具备同时对受影响的

内存范围的写访问权。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

考虑下面的例子：

1. PLC 寄存器的初始状态是 0x02h。
2. Vijeo Citect 驱动程序读这个寄存器的值（做一个备份），准备修改第三位。
3. 然而，在驱动程序将改变写回到 PLC 之前，PLC 代码改变了寄存器位 0 和 4 的值为 0x13h。
4. Vijeo Citect 驱动程序然后改变寄存器备份中位 3 的值为 0x0Ah。当它写到 PLC 中时，它覆盖了寄存器的整个备份（不只是被改变的位）。因为 PLC 代码在 Vijeo Citect 读和写之间修改了位 0 和 4，这些改变被覆盖了。

验证为基于标签的驱动程序分发的工程数据

Vijeo Citect 使用数值索引值来唯一标识工程中的所有变量标签。当从 I/O 服务器为基于标签的驱动程序请求数据时，它们被用作参考点。

编译工程时自动生成这些索引值。当一个分布式工程具有代表了不同计算机上的不同标签地址的索引值时，可能会出现一些状况。基于这一原因，Vijeo Citect 实施了许多自动检查措施，能够验证工程的标签索引值并标记出任何差异。

初始安全检查会在单元级别在客户机计算机上进行，允许将标签索引不匹配的情况隔离到特定的客户机上，然后任何请求才会被发送到 I/O 服务器。这可确保单元地址、单元类型、原始数据类型和标签地址匹配客户机和服务器计算机上的所有索引值。找到的任何差异都将被客户机计算机上的硬件报警标记。

每个画面都将被检查，以确认根据变量数据库的当前版本进行编译。每当一个基于标签的驱动程序加载了变量数据库后都会执行一次检查，以检测它是否匹配当前的标签地址。参数 TagAddressNoCase 允许您调整这些检查的大小写敏感度。

此外，Vijeo Citect 还将检查在尝试编译时工程当前是否在本地计算机上运行，因为这是可能导致不匹配的索引值的情况之一。

如果工程使用基于标签的驱动程序并且当前处在运行状态，Vijeo Citect 将停止编译器并在错误数据库中生成一个错误，指出 Citect32.exe 仍在运行。`.ini` 参数

`[General]CitectRunningCheck` 允许您覆盖此功能，然而建议您将它保留为启用状态，以便标签索引值可按预期指派。

写入延迟效果

Vijeo Citect 以异步的方式执行写 I/O 设备（即当写 I/O 设备时，写操作需要消耗一定时间，同时 Vijeo Citect 继续执行其他操作）。如果其它 Cicode 认为写操作立刻完成了，您就会遇到一些负面影响，如跨两个 Cicode 线程的标签变量的值不一致等。

如果你有这样的 Cicode：

```
PLC_VAR = 1234;
Prompt("Variable is " + PLC_VAR : #####);
```

第一行写到 PLC。

当 Vijeo Citect 执行第一行时，它产生写 1234 这个值到 PLC 变量 PLC_VAR 的请求到 I/O 服务器，然后在写值到 PLC 完成之前 Vijeo Citect 执行 Cicode 的另一行。Vijeo Citect 如此操作是为了当 Cicode 等待缓慢的 I/O 设备时不停止运行。因为写 PLC 没有完成，你可能认为下一条显示 PLC_VAR 的值不会是 1234。不过，此 Cicode 将显示正确的值（1234），因为无论什么时候 Vijeo Citect 写 PLC，它都将首先更新变量的本地备份：任何以后的 Cicode 都将得到正确的值。

因为 Vijeo Citect 可能会保存多个 I/O 设备变量的备份，并且仅更新与当前 Cicode 关联的备份，所以有时此方案不会起作用。其他变量将保持 I/O 设备的旧值直到被更新（读 I/O 设备）。对每一个显示页、Cicode 文件、报警、趋势和报表都是分离的数据区。如果你想要写键盘命令页的 I/O 设备变量，与这一页相关的 I/O 设备变量的备份就将被更新，然而与其它页和所有 Cicode 相关的变量的备份直到下一次读（由参数 [Page]ScanTime 确定）才更新。如果你调用一个假设写已经完成的 Cicode 函数，就将得到最新的值。

在 Cicode 函数中，工作区将写入 I/O 设备变量。所有的 Cicode 函数分享相同的 I/O 设备变量使得写可以象期望的那样运行。

```
FUNCTION
TestFunc(INT nValue)
    PLC_VAR = nValue;
END
```

在异步写 I/O 设备方面延迟继承的另一个负面效应是您可能认为 Vijeo Citect 已经成功的写 I/O 设备了，但事实上写操作可能尚未完成甚至是根本没有写。在这种情况下，仍然可能是硬件（或配置）方面的错误会阻止写操作的成功。如果这种不成功的写操作会产生错误，您的 Cicode 可能无法得到通知或将使用 IsError() 函数，因为在执行了最初的 I/O 设备写命令后，Cicode 已经继续执行。因此，当在允许代码继续执行之前检查写操作是否成功是一件非常重要的事时，您可以在写之后插入函数 TagRead()，然后核对该变量的值。TagRead() 使 Cicode 重新读取 I/O 设备变量，以使用户能检查新的变量。TagRead() 是一个阻塞函数。直到操作完成之前它一直阻塞着 Cicode 任务的调用。

```
PLC_VAR = 1234;
sTestStr = TagRead("PLC_VAR");
IF sTestStr <> 1234 THEN
    Prompt("Write not completed");
END
```

此时，数据将从物理 PLC 中获得，而不是从 I/O 服务器的缓存中，因 I/O 服务器将使任何与 PLC 写相关的缓存数据无效。这将允许你测试一个完整的写。请注意，同时运行的其他 Cicode 任务将不会等待 TagRead 的完成，所以它们可能使用新的或旧的值，依赖于他们

是否使用相同的 PLC 变量备份。用户也可以使用函数 `CodeSetMode(0, 0)` 来停止 Vijeo Citect 写入变量的本地备份。

使用远程设备通过调制解调器通讯

拨号远程 I/O 设备是通过 PSTN（公共交换电话网络）与 Vijeo Citect 相连接的设备，可以通过提前选择和提前配置的调制解调器访问。

一旦连接，Vijeo Citect 就能够读写，拨号远程 I/O 设备与其它 I/O 设备完全相同：本地或远程。

通讯可以是：

- **基于请求：**由 Vijeo Citect 使用 `IODeviceControl()` 启动或由远程 I/O 设备初启动（例如报告报警条件）。
- **定期：**例如，传输一个周期的记录事件。
- **长期：**例如，监控远程水坝的水位。

唯一的限制因素可能是因通讯功能不兼容而导致无法将调制解调器连接到 I/O 设备。许多 I/O 设备具有固定的设置，只能以制造商确定的预设速率通讯。如果调制解调器不能匹配这些设置，将无法建立通讯。

为简化通讯设置，可以将拨号远程 I/O 设备连接到具有相同通讯方式的调制解调器和端口。由于 I/O 设备已连接到相同的调制解调器，Vijeo Citect 能够与所有 I/O 设备逐一通讯，无需挂断和重拨。这减少了必需的电话调用的数目，增加了通讯的效率和速度。

每个 I/O 服务器端和每个 I/O 设备端都至少有一个调制解调器。

另请参见

[I/O 服务器端调制解调器](#)

[I/O 设备端调制解调器](#)

[I/O 设备的多路限制](#)

[配置多路远程 I/O 设备](#)

[可拨号远程 I/O 设备的 I/O 服务器冗余](#)

[可拨号远程 I/O 设备通讯的故障排除](#)

I/O 服务器端调制解调器

要确定在 I/O 服务器端使用的调制解调器数量，需确定每个调制解调器执行的功能。单个调制解调器可以执行以下功能中的任何一个：

拨出	对响应 Vijeo Citect 请求（例如，调度请求、基于事件的请求或操作员请求）的远程 I/O 设备进行拨号。远程 I/O 设备也会返回拨号。
拨入	仅接收来自远程 I/O 设备的拨号，识别呼叫者，然后立即挂断，因此可以接收其它拨号。Vijeo Citect 随后使用回拨或拨出调制解调器返回拨号。

回拨	仅从远程 I/O 设备返回拨号。
拨入和拨出	接收来自远程 I/O 设备的拨号并对远程 I/O 设备执行调度拨号。
拨入和回拨	接收并返回来自远程 I/O 设备的拨号。

调制解调器设置取决于系统要求。做出决定时，应考虑以下指导方针：

- 如果需要一次与多个远程 I/O 设备通讯，则需要每个 I/O 设备配置单独的调制解调器。否则，必须在依次访问 I/O 设备时等待，并且拨号可能被挂断。
- 如果对 I/O 设备进行了调度请求并且还需要从远程 I/O 设备紧急返回拨号，则需要每个功能至少配置一个调制解调器。例如，如果远程 I/O 设备数量很多，并且需要 Vijeo Citect 快速响应，则应在 I/O 服务器端提供其它调制解调器。这样会减少 I/O 设备拨入时（比方说，紧急报警拨入）设备被占用的机会。
- 在远程 I/O 设备数量很多的庞大系统或来自远程 I/O 设备的拨号非常紧急的系统中，最好至少将一个调制解调器专门用于回拨。这样可以对来自远程 I/O 设备的拨入拨号快速响应。这也意味着拨出的调度不会被破坏（如果使用相同的调制解调器返回拨号和调度拨号，则调度拨号将被强制等到回拨拨号完成）。

另请参见

[I/O 设备端调制解调器](#)

[I/O 服务器端的调制解调器配置示例](#)

I/O 设备端调制解调器

如果通讯要求（电话号码、波特率、数据位、停止位和奇偶校验）相同，则可以将多个 I/O 设备连接到单个调制解调器。对于通讯要求不同的每个远程 I/O 设备，必须使用单独的端口和调制解调器。

确定要使用的调制解调器的数量时，应考虑以下问题：

- 访问 I/O 设备后，可以为组中的其它 I/O 设备保留连接。如果 I/O 服务器需要从通讯详细信息相同的另一个 I/O 设备请求数据，则 I/O 服务器会等到当前请求完成，然后使用相同的连接进行第二次请求。
- 可以配置调制解调器初始化电话拨号（调用 Vijeo Citect）和/或接收电话拨号。此操作由 Vijeo Citect 独立完成。

注意：无论调制解调器的位置在哪里，都必须确认其**数据位、奇偶校验、停止位和串行端口速度**设置与设备制造商定义的远程 I/O 设备兼容，否则通讯链接会不正常。

另请参见

[I/O 服务器端调制解调器](#)

I/O 服务器端的调制解调器配置示例

下面的示例示范了如何设置 I/O 服务器端的调制解调器，以适应 I/O 设备的不同组合。

示例 1

您的所有远程 I/O 设备都具有相同的通讯要求（数据位、停止位、奇偶校验和波特率）- 19200 8 E 1。

您不期望从 I/O 设备紧急调用，或您只有几个远程 I/O 设备。这意味着您可在 I/O 服务器端使用一个调制解调器。可以设置这个调制解调器以应答和返回传入的调用，并进行调度和其他 Vijeo Citect 已启动的调用。

要配置调制解调器，请在 Windows 中对其进行定义。假设逻辑调制解调器叫做“标准调制解调器”，请按以下内容对其进行配置：

端口	调制解调器名称	最大速度	数据位	奇偶校验	停止位
COM1	标准调制解调器	19200	8	E	1

您可以随后在 Vijeo Citect 中将其配置成拨出调制解调器和拨入调制解调器：

调制解调器名称	拨出	拨入	回拨
标准调制解调器	TRUE	TRUE	FALSE

示例 2

在这个示例中，您的 I/O 设备共使用两个不同的通讯规格 - 9600 7 O 1 和 19200 8 E 1。

您不期望从 I/O 设备获得重要调用，或您只有几个 I/O 设备。这意味着您可以在 I/O 服务器端只使用一个调制解调器。这个调制解调器必须接收和返回来自所有 I/O 设备的调用，并且必须拨出调用到所有的 I/O 设备。

要配置调制解调器，您必须首先在 Windows 中定义它（通过 Windows 的“控制面板”）。记住，在这里您不仅要定义物理调制解调器。您必须为每一个通讯规格定义一个 Windows（虚拟）调制解调器。

到目前为止，这里提供了两个虚拟调制解调器 - 一个为 9600 7 O 1，一个为 19200 8 E 1。然而，Windows 不允许您定义两个调制解调器作为拨入调制解调器。它仅允许您一个端口定义一个拨入调制解调器。如果您选择第一个，它就不能接收来自第二个的调用，反之亦然。

这意味着您必须设置一个单独的虚拟调制解调器，无论使用哪一个通讯规格，它都可以应答调用。这个调制解调器可以设置为使用通用通讯规格 9600 8 N 1。

所以在 Windows 中，最终将有三个逻辑调制解调器（两个拨出，一个拨入）。假设这些逻辑调制解调器被称作“标准调制解调器”到“标准调制解调器 #3”，您可以将其配置如下：

端口	调制解调器名称	最大速度	数据位	奇偶校验	停止位
COM1	标准调制解调器	9600	7	O	1
COM1	标准调制解调器 #2	19200	8	E	1
COM1	标准调制解调器 #3	9600	8	N	1

您可以随后在 Vijeo Citect 中按下述方式配置调制解调器。

调制解调器名称	拨出	拨入	回拨
标准调制解调器	TRUE	FALSE	FALSE
标准调制解调器 #2	TRUE	FALSE	FALSE
标准调制解调器 #3	FALSE	TRUE	FALSE

示例 3

在这个示例中，有五种不同的通讯规格 - 9600 7 O 1、19200 8 E 1、4800 8 N 1、9600 8 N 1 和 19200 8 N 1。

如果您期望从 I/O 设备接收重要调用，或您有相当多的 I/O 设备，您应当在 I/O 服务器端设置三个调制解调器：

- 一个位于 COM3 端口，专用于接收来自 9600 7 O 1 I/O 设备的调用。
- 一个位于 COM2 端口，用于拨出到 4800 8 N 1、9600 8 N 1 和 19200 8 N 1 I/O 设备。
- 一个位于 COM1 端口，用于拨出到 9600 7 O 1 和 19200 8 E 1 I/O 设备。

两个拨出调制解调器将返回调用，同时启动调用以响应调度的请求，等等。

要配置调制解调器，您必须首先在 Windows 中定义它们（通过 Windows 的“控制面板”）。记住，在这里您不仅要定义物理调制解调器。您必须为每一个通讯规格定义一个单独的 Windows（虚拟）调制解调器。

假设这些逻辑调制解调器被称作“标准调制解调器”到“标准调制解调器 #6”，您可以将其配置如下：

端口	调制解调器名称	最大速度	数据位	奇偶校验	停止位
COM1	标准调制解调器	9600	7	O	1
COM1	标准调制解调器 #2	19200	8	E	1
COM2	标准调制解调器 #3	4800	8	N	1
COM2	标准调制解调器 #4	9600	8	N	1
COM2	标准调制解调器 #5	19200	8	N	1
COM3	标准调制解调器 #6	9600	7	O	1

您可以随后在 Vijeo Citect 中按下述方式配置调制解调器：

调制解调器名称	拨出	拨入	回拨
标准调制解调器	TRUE	FALSE	FALSE
标准调制解调器 #2	TRUE	FALSE	FALSE
标准调制解调器 #3	TRUE	FALSE	FALSE
标准调制解调器 #4	TRUE	FALSE	FALSE
标准调制解调器 #5	TRUE	FALSE	FALSE
标准调制解调器 #6	FALSE	TRUE	FALSE

示例 4

在这个示例中，您的 I/O 设备共使用三个不同的通讯规格：9600 7 O 1、19200 8 E 1 和 9600 8 N 1。但是，在这个示例中，您期望来自 I/O 设备的紧急调用，因此您需要一个调制解调器专用于返回调用。

此时您必须将调制解调器配置如下：

- 一个位于 COM1 端口，用于对所有远程 I/O 设备拨出调用（对于调度的调用等）。
- 一个位于 COM2 端口，用于接收来自远程 I/O 设备的调用。
- 一个位于 COM3 端口，专用于返回这些调用。

要配置调制解调器，您必须首先在 Windows 中定义它们（通过 Windows 的“控制面板”）。记住，在这里您不仅要定义物理调制解调器：您必须为每一个通讯规格定义一个单独的 Windows（虚拟）调制解调器。这意味着您必须配置：

- 三个逻辑调制解调器，它们位于物理拨出调制解调器所连接的端口上。

- 一个逻辑调制解调器，位于物理拨入调制解调器所连接的端口上。
- 三个逻辑调制解调器，它们位于物理回拨调制解调器所连接的端口上。

假设需要的全部七个逻辑调制解调器被称作“标准调制解调器”到“标准调制解调器 #7”，这些调制解调器可被配置如下：

端口	调制解调器名称	最大速度	数据位	奇偶校验	停止位
COM1	标准调制解调器	9600	7	O	1
COM1	标准调制解调器 #2	19200	8	E	1
COM1	标准调制解调器 #3	9600	8	N	1
COM2	标准调制解调器 #4	9600	8	N	1
COM3	标准调制解调器 #5	9600	7	O	1
COM3	标准调制解调器 #6	19200	8	E	1
COM3	标准调制解调器 #7	9600	8	N	1

您可以随后在 Vijeo Citect 中按下述方式配置调制解调器：

调制解调器名称	拨出	拨入	回拨
标准调制解调器	TRUE	FALSE	FALSE
标准调制解调器 #2	TRUE	FALSE	FALSE
标准调制解调器 #3	TRUE	FALSE	FALSE
标准调制解调器 #4	FALSE	TRUE	FALSE
标准调制解调器 #5	FALSE	FALSE	TRUE
标准调制解调器 #6	FALSE	FALSE	TRUE
标准调制解调器 #7	FALSE	FALSE	TRUE

I/O 设备的多路限制

如果您正在使用多路调制解调器，应该使用 I/O 设备去指定呼叫者 ID，而不是调制解调器。这是因为使用调制解调器发放 ID 时，无论调用与哪个 I/O 设备相关，它都发送相同的 ID。这使得很难识别触发调用的 I/O 设备。

通过使用 I/O 设备发放 ID，I/O 服务器将收到每个 I/O 设备的唯一呼叫者 ID。但不是所有 I/O 设备都能发放呼叫者 ID。如果使用多路，您应该使用可以指定呼叫者 ID 的 I/O 设备。

将拨号远程 I/O 设备配置为与 Vijeo Citect 通信：

1. 运行快速通信向导。
2. 通过选择相关 I/O 服务器，然后选择 I/O 设备，并在需要时创建新的实例完成该向导。
3. 在该向导的“预定安排”画面上，选中**将 I/O 设备连接到 PSTN** 复选框。
4. 为 Vijeo Citect 选择适当的日程表以便与远程 I/O 设备进行通信。（对于长期连接 - 无论何时运行 Vijeo Citect，请选择**启动时**。）例如（全部基于 10:00:00 的**同步时间**）：
 - 如果在**重复间隔**字段中输入 **12:00:00**，并在上午 9 点启动工程，则 I/O 服务器将在上午 10 点与 I/O 设备通讯，此后每 12 小时通讯一次（即下午 10 点，然后是第二天上午 10 点，依此类推）。
 - 如果您输入 12:00:00，并在下午 4 点启动您的工程，I/O 服务器将在下午 10 点与 I/O 设备通讯，然后在第二天的上午 10 点再次通讯，等等。Vijeo Citect 将假设通讯是上午 10 点建立的，因此将继续按此假设进行推理，即在上午 10 点后每隔 12 小时重复通讯一次。
 - 如果输入 3 天，并在星期三上午 9 点启动工程，则 I/O 服务器将在上午 10 点与 I/O 设备通讯，此后每 3 天通讯一次（即星期六上午 10 点，然后是下周的星期二上午 10 点，依此类推）。
 - 如果您在**重复间隔**字段中输入 12 月 6 日，并在 11 月启动您的工程，那么 I/O 服务器将在 12 月 6 日上午 10 点与 I/O 设备通信，然后在下一年的 12 月 6 日再次通信。
5. 对长期连接，选择**启动时**。若要断开一个长期连接，请调用 IODeviceControl() 函数（使用类型 8）。
6. 键入 Vijeo Citect 所需的电话号码，以拨号挂接到远程 I/O 设备上的远程调制解调器。包括拨号之前需要获得连接到外部 PSTN 线的预拨号码（例如，**0**（零））- 如果适当。
7. 在下一个向导画面上，如果设备配置为拨号到 Vijeo Citect，请为远程 I/O 设备创建唯一标识呼叫者名称，以便它被 Vijeo Citect 识别。
8. 按照该向导下一画面上的说明进行操作，然后单击**完成**。

另请参见

[配置多路远程 I/O 设备](#)

配置多路远程 I/O 设备

来自同一个远程调制解调器的多路远程 I/O 设备使 Vijeo Citect 能够依次与每一个 I/O 设备通讯，使用相同的电话连接，而不是挂断再拨。

尽管您可以使用“快速通讯精灵”来配置多路远程 I/O 设备，但我们还是建议您手动配置。该精灵可以为每个 I/O 设备创建新的端口。这意味着您不能有多于 255 个 I/O 设备。

1. 运行“快速通讯精灵”来配置第一个设备。
2. 手动配置其他所有的 I/O 设备。

3. 打开 Citect 工程编辑器。
4. 选择**通讯 | I/O 服务器**，然后选择要与 I/O 设备通讯的 I/O 服务器。
5. 选择**通讯 | I/O 设备**。完成此对话框。
6. 要增加系统的效率和能力，您可以给具有相同通讯设置的所有 I/O 设备分配相同的端口名称。

注意：如果您是多路的并且希望能够拨入 I/O 服务器，您应该使用 I/O 设备去指定调用 ID，而不是调制解调器。这是因为使用调制解调器发放 ID 时，无论调用与哪个 I/O 设备相关，它都发送相同的 ID。这使得很难识别触发调用的 I/O 设备。

通过使用 I/O 设备发放 ID，I/O 服务器将收到每个 I/O 设备的唯一呼叫者 ID。但不是所有 I/O 设备都能发放呼叫者 ID。如果使用多路，您应该使用可以指定呼叫者 ID 的 I/O 设备。

如何设置连接到可拨号远程 I/O 设备的调制解调器：

您可以连接多个 I/O 设备到相同的调制解调器。这意味着 Vijeo Citect 可以使用相同的电话连接依次与每一个 I/O 设备通讯，而不需要挂断并且再拨。这减少了必需的电话调用的数目并且增加了通讯的效率和速度。

1. 使用已安装的电话程序（例如超级终端或电话拨号器）将调制解调器连接到 PC 上。
在这里您可以配置调制解调器应答来自于 Vijeo Citect 的调用并/或启动拨号。
2. 如果调制解调器需要拨 Vijeo Citect，配置它启动拨电话到一个已定义好的 Vijeo Citect I/O 服务器拨入型调制解调器（遵循生产厂家的指导）。
3. 根据您的硬件的不同，调制解调器或智能 PLC 可以负责启动对 Vijeo Citect 的调用并识别调用者。无论哪一个设备负责启动调用，都必须设置调用 ID。调用 ID 可以是任何字母数字字符和/或 '_' 字符（下划线）的组合。
一些调制解调器有点切换设置，一些调制解调器在永久内存中有包括自动拨号的初始字符串。参考调制解调器手册获得细节信息。
您可以使用快速通讯精灵或 I/O 设备表去设置 I/O 设备的调用 ID。
如果您正在单个调制解调器上使用多路，您应该使用 I/O 设备去指定调用 ID，而不是调制解调器。这是因为无论调用与哪一个 I/O 设备相联系，使用调制解调器指定 ID 都将发送相同的 ID，从而很难识别触发调用的 I/O 设备。
通过使用 I/O 设备发放 ID，I/O 服务器将收到每个 I/O 设备的唯一呼叫者 ID。但不是所有 I/O 设备都能发放呼叫者 ID。如果使用的是多点式通讯，则应使用能够发放呼叫者 ID 的 I/O 设备。
4. 设置调制解调器的**数据位、奇偶校验、停止位和串行速率**以匹配生产厂家的规范，并与 I/O 设备通讯。
一些调制解调器不允许您通过诸如扩展 AT 命令或点切换的方法去操作通讯。如果是这样，唯一的方法就是使用这个设置值与调制解调器通讯（例如，通过超级终端）。一旦这样做了，调制解调器就记住了最后的串口通讯的值。
5. 连接调制解调器到 I/O 设备。

要在 I/O 服务器上配置调制解调器，必须在 Windows 中设置它，然后在 Vijeo Citect 中设置它。

如果您的 I/O 设备都是相同的，您就不用为每一个调制解调器这样做。然而，如果您的 I/O 设备使用不同的通讯规格（数据位、奇偶校验、停止位和串口速率），那么您的调制解调器必须能够使用所有这些设置进行通讯。要这样设置，您必须在 Windows 和 Vijeo Citect 中为每个规范创建调制解调器。（请参见 [在 I/O 服务器上配置调制解调器的示例](#)）

如何在 Windows 中设置调制解调器

1. 每个连接到 Vijeo Citect I/O 服务器 PC 的调制解调器必须首先使用**开始 | 设置 | 控制面板 | 电话和调制解调器选项**在 Windows 中进行配置。
2. 选择**调制解调器**分页，点击**添加**启动“安装新调制解调器”精灵。
3. 选中标有**不要检测我的调制解调器；我将从列表中选择**的复选框，然后点击**下一步**。
4. 在厂商列表中选择**标准调制解调器类型**。

注意：不要从厂商列表中选择一个商标名调制解调器，即使您正安装的调制解调器的名称包括在列表中。不要点击**从硬盘安装**。

5. 从型号列表中选择**标准 xxxx bps 调制解调器速率**，从而准确匹配要通过此调制解调器进行通讯的 I/O 设备的每秒比特率。检查设备确定设备的通讯率。如果仍然不确定，选择 9600 bps 型号。如果需要，这个值以后可以改变。
6. 不要点击**从硬盘安装**。点击**下一步**。
7. 选择调制解调器连接到的**COMx 端口**。点击**下一步**。
8. 点击**完成**。Windows 将在调制解调器属性表格的调制解调器列表中显示这个调制解调器。
9. 注意，没有给出选项去选择和设置数据位、奇偶校验或停止位等信息。调制解调器精灵将为所有的标准调制解调器类型自动设置缺省值 8-none-1。要改变这些设置以匹配远程 I/O 设备的数据位、奇偶校验、停止位等要求，请在列表选择一个调制解调器，然后点击**属性**按钮。
10. 点击**高级**分页并点击**改变缺省参数选择**。
11. 在下一个对话框中点击**高级**分页，访问调制解调器的数据位、奇偶校验和停止位设置。
12. 使用下拉选项改变数据位、奇偶校验和停止位设置，使它们准确匹配远程 I/O 设备和远程调制解调器使用的设置。不要改变任何高级设置。（缺省是硬件流控制。）
13. 点击**确定**。如果相同速率的调制解调器被安装在相同的端口上作为一个已存在的调制解调器，Windows 将要求确认您是否要不止一次安装相同的设备。点击**是**安装调制解调器的重复副本。
14. 预先配置要在远程拨号 I/O 设备上使用的调制解调器。它将在下一步测试调制解调器的配置设置。
15. Vijeo Citect 没有运行时，通过使用一个终端通讯程序，诸如 Windows 提供的超级终端或电话拨号器，确认本地和远程调制解调器能够正常地相互通讯。

一旦调制解调器的通过程在 Windows 中设置和测试完成，就可以在 Vijeo Citect 中设置它了。

如何在 Vijeo Citect 中设置调制解调器：

在完成下列步骤之前，请确认您已在 Windows 中设置了调制解调器（见上文）。

1. 打开 **Citect 工程编辑器**。
2. 选择**通讯 | I/O 服务器**，然后选择调制解调器要连接的 I/O 服务器。
3. 选择**通讯 | 调制解调器**。将出现“调制解调器属性”对话框。

选项	描述
服务器名称（16 个字符）	调制解调器所连接的 I/O 服务器的名称。
调制解调器名称（64 个字符）	配置的调制解调器的名称（它将出现在 Windows 的“控制面板 电话和调制解调器选项”中）。
注释（48 个字符）	任何有用的注释。
使用调制解调器产生对外呼叫	确定此调制解调器是否用于启动从 I/O 服务器到一个远程拨号 I/O 设备的呼叫。（拨出） 这包括计划性的呼叫、事件驱动的呼叫或者作为对拨入 I/O 设备响应的呼叫。
使用调制解调器应答拨入呼叫	确定此调制解调器是否用于接收来自一个远程拨号 I/O 设备的呼叫。（拨入）

以下字段通过扩展形式实现（按 **F2**）。

使用调制解调器响应 I/O 设备	确定此调制解调器是否用于启动从 I/O 服务器到远程拨号 I/O 设备的呼叫，以作为对接收自 I/O 设备的呼叫的响应。（回拨）
------------------	--

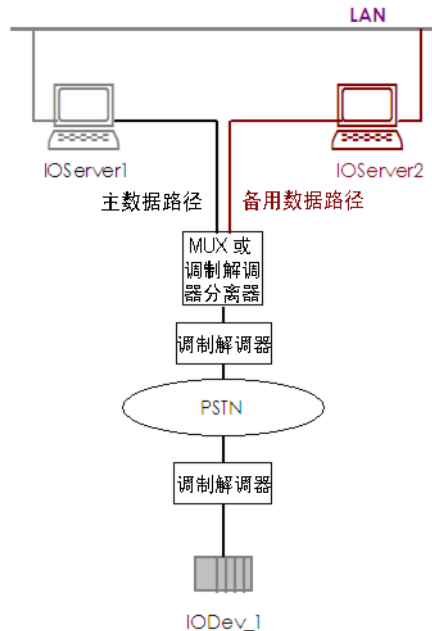
4. 完成此对话框。

注意：Vijeo Citect 允许在 I/O 服务器上最多设置 256 个调制解调器与远程拨号 I/O 设备通讯。在成功建立通讯之前，还必须在 I/O 服务器上使用 Vijeo Citect 正确配置所有目标远程 I/O 设备。

可拨号远程 I/O 设备的 I/O 服务器冗余

可以更改 I/O 服务器在应答呼叫之前等待的振铃次数（使用 **[Dial]RingCount** 参数）。如果要使用冗余 I/O 服务器，将调用主要 I/O 服务器（缺省）。但通过在每个 I/O 服务器中将 **[Dial]RingCount** 不同参数值，可以在主要 I/O 服务器未应答时使用备用 I/O 服务器应答呼叫。

考虑以下设置：



如果在 IO Server1 上设置的振铃数为 3，在 IO Server2 上设置的振铃数为 4，则 IODev_1 将尝试呼叫 IO Server1。如果 IO Server1 在 3 次振铃后未应答，则 IO Server2 将应答呼叫。

另请参见

[可拨号远程 I/O 设备通讯的故障排除](#)

可拨号远程 I/O 设备通讯的故障排除

当使用拨号远程 I/O 设备牵涉到所连接设备的通讯速度、奇偶校验和控制信号时，经常会遇到这种挑战。如果改变速度和奇偶校验不能解决通讯中断问题，则应评估调制解调器的应答代码或命令响应。

下面这些设置可能有助于对解决拨号通讯中断。（因为不是所有的调制解调器在相同的方式上都支持相同的命令，这只是参考。参考调制解调器手册获得详细信息。）

PC 端的调制解调器

```

ATV1 //Enables long-form (verbose) result codes
ATQ0 //Result codes are sent on the RS-232 connection
ATE0 //Commands sent from the computer are not echoed back to the RS-232
connection
AT&C1 //DCD will follow carrier on the line
AT&K0 //Handshaking OFF
ATW0 //Upon connection, only DTE speed is reported
AT%C0 //Compression OFF
AT&D0 //DTR always on

```


如果 PC 端的调制解调器被配置成自动应答呼叫（即使是在 Vijeo Citect 工程关闭时），从 I/O 设备报告的数据可能会丢失。因此，建议在将工程离线之前关闭 PC 调制解调器的自动应答功能。为此，请将下列参数设置为零：

```
ATS0 = 0 // Auto answer OFF
```

但是，也请注意，这有可能影响到使用调制解调器（而非 Vijeo Citect）的应用程序，因为调制解调器无法在 Vijeo Citect 未工作时应答电话。

在 I/O 设备端的调制解调器

```
ATV0 //Enables short-form result codes
ATQ1 //No result codes are sent on the RS-232 connection
ATE0 //Commands that are sent from the computer are not echoed
back to the RS-232 connection
AT&C1 //DCD will follow carrier on the line
AT&K0 //Handshaking OFF
ATW0 //Upon connection, only DTE speed is reported
AT%C0 //Compression OFF
AT&D0 //DTR always on
ATS0 //Set to greater than 0 (sets the number of rings required before the modem
answers an incoming call).
```

永久连接的选择方式（向后兼容）

如果您正在设置连接到拨号远程 I/O 设备调制解调器，建议您遵循 [与拨号远程 I/O 设备通讯](#) 中介绍的过程。此方法可用于实现向后兼容。

调度通讯

Vijeo Citect 允许用户通过 I/O 设备调度通讯（忽略连接类型：调制解调器、无线电链接等）。例如，如果在一个网络或线路中连接了多个 I/O 设备，则可以调度读数，以便重要 I/O 设备的读取次数比重要性较低的 I/O 设备多。此外，为了节约带宽，水坝水位监视器的供水厂商可以从 Vijeo Citect 每小时调度一次水位读数。

另请参见

[指定调度](#)

[写入调度的 I/O 设备](#)

[从调度的 I/O 设备进行读取](#)

指定时间表

要配置与 I/O 设备的预定通信，必须将其标记为“预定的”设备。可使用[快速通信向导](#)完成此操作。如果您的 I/O 设备不能够预定通信，在向导中就不会显示预定选项。

注意：预定通信不适用于专为处理例外报告协议而设计的驱动程序。要与预定之外的 I/O 设备通信，请使用 `IIODeviceControl` 函数。

要预定与 I/O 设备的通信，请执行以下操作：

1. 选择工程编辑器（或按工程编辑器图标）。
2. 选择**通信 | 快速向导**，或打开 Citect 浏览器。
3. 双击当前工程的通信文件夹中的**快速 I/O 设备设置**图标。
4. 依次按各屏幕上的说明操作，选择相关 I/O 设备等。当预定屏幕显示时，选中**将 I/O 设备连接到 PSTN**框（即使您的 I/O 设备不是通过调制解调器连接的），但将要拨打的电话号码和呼叫者 ID 域留空。
5. 填写时间表域以激活所需预定。例如（全部基于在 10:00:00 点同步）。

如果**重复间隔**字段中输入 12: 00: 00，并在上午 9 点启动工程，则 I/O 服务器将在上午 10 点与 I/O 设备通讯，此后每 12 小时通讯一次（即下午 10 点，然后是第二天上午 10 点，依此类推）。

- 如果您输入 12:00:00，并在下午 4 点启动您的工程，I/O 服务器将在下午 10 点与 I/O 设备通讯，然后在第二天的上午 10 点再次通讯，等等。它将假设通讯是上午 10 点建立的，因此将继续按此假设进行推理，即在上午 10 点后每隔 12 小时重复通讯一次。
- 如果输入 3 天，并在星期三上午 9 点启动工程，则 I/O 服务器将在上午 10 点与 I/O 设备通讯，此后每 3 天通讯一次（即星期六上午 10 点，然后是下周的星期二上午 10 点，依此类推）。
- 如果在**重复间隔**中输入 12 月 6 日，并在 11 月期间启动工程，则 I/O 服务器将在 12 月 6 日上午 10 点与 I/O 设备通信，然后是第二年 12 月 6 日再次通信，依此类推。

6. 对长期连接，选择**启动时**。要断开长期连接，必须调用类型为 8 的 `IIODeviceControl()` 函数。
7. 如果您的 I/O 设备没有经过调制解调器连接，您必须进入端口表（从通信菜单选择端口表）并将端口号改为实际的 COM 端口号。

另请参见

[写入调度的 I/O 设备](#)

写入调度的 I/O 设备

无论 I/O 设备什么时候激活通讯（按照它的预定），您都可以直接写。然而，如果你试图在不通讯的时候写，写请求就将排队等候直到可以。例如，你可以预定每小时写一次。如果有人在这一小时中在控制客户端改变了这个标签的值，这个改变值到一小时期满才会写到 I/O 设备。

因为写请求只有当通讯时才能够写到 I/O 设备，所以您应当确认在关闭之前所有的未决写都已经完成。

注意：不要使用一个预定 I/O 设备控制硬件，因为硬件的确切状态是不可知的。尽管你可以从缓存中读状态，它也可能在缓存创建之后就改变了。



不希望的设备操作

请勿在 Vijeo Citect 管理的系统中使用已调度的 I/O 设备来控制硬件。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

另请参见

[从调度的 I/O 设备进行读取](#)

从调度的 I/O 设备进行读取

当 I/O 服务器与 I/O 设备通讯时，它将立即写入任何排队等候的写入请求，然后读取所有 I/O 设备的标签。这些参数值将被存储在缓存中，使您仍可在通讯间访问它们。

注意：由于 I/O 服务器将在通讯启动时读取所有标签，因此，即使实际没有使用某些标签读取，许可点数也会增加。

在延长连接期间保持数据最新

通常，通信在读写请求完成时即告终止。然而有时，您可能想要延长通信（例如，通过类型 7 调用 `IODeviceControl()` 函数）。

在这种情况下（如果完全读缓存被禁止），当客户计算机从 I/O 服务器请求设备数据时，它会从其缓存中获取数据，而不是从 I/O 设备。即使 I/O 服务器与设备保持连接时也可能发生这种情况。

要获取 I/O 设备的最新数据，可以使用 `Cicode` 强制要求进行定期读取，或者通过将 `IODeviceControl()` 函数设置为类型 11 以更改缓存超时。

例如：

```
INT hTask;  
// Initiate communications and read tags.  
// Sleep time will depend on how fast your  
// modems connect.  
FUNCTION  
DialDevice(STRING sDevice)  
    INT bConnected = 0;  
    INT nRetry = 5;
```

```
hTask = TaskHnd("");
IODeviceControl(sDevice, 7, 0);
Sleep(20);
WHILE bConnected <> 1 AND nRetry > 0 DO
    bConnected = IODeviceInfo(sDevice, 18);
    nRetry = nRetry - 1;
    Sleep(10);
END
IF bConnected = 1 THEN
    WHILE TRUE DO
        Sleep(2);
        IODeviceControl(sDevice, 16, 0);
    END
END
END
// Kill the read task and terminate the connection.
FUNCTION
HangupDevice(String sDevice)
    TaskKill(hTask);
    IODeviceControl(sDevice, 8, 0);
END
```

还可通过启用完全读缓存来迫使 I/O 服务器直接从 I/O 设备读取数据。设置 `[Dial]ReadThroughCache`，当 I/O 服务器连接到设备时，它将直接从设备中提供数据给请求的客户机。在这一时间中缓存并不更新数据，而是在服务器断线之前刷新为最新的设备数据。

注意：如果使用调制解调器，可能需要调整或停用调制解调器中的休止状态定时器，阻止它们在没有数据读取时断线。休止状态定时器由 S30 寄存器控制。如果您的调制解调器不支持这个寄存器，请参考调制解调器手册。

避免不必要的多次 I/O 设备数据读取

为了避免不必要的 I/O 设备读取，可以使用数据缓存临时将从设备读取的数据存储在 I/O 服务器的内存中。这意味着，如果 I/O 服务器在短时间内多次收到对设备数据的请求，它不是再次联系该 I/O 设备读取相同的数据，而是可以从缓存中获取该数据。

Chapter: 16 标签过程变量

您必须向 Vijeo Citect 在运行系统中使用的每一个 I/O 设备 变量指定一个变量标签。要定义变量标签，需要在变量标签数据库中声明它们。变量标签成为一个标记符，用于参考 I/O 设备寄存器的地址。使用标签有以下几项好处：

- 使用变量时不必每次都记住地址。可以使用应为逻辑性和描述性的标签名，因此不会混淆。
- I/O 设备的地址只被定义一次。如果更改此地址，只需更新变量标签定义即可，而不是配置中的每一种情况。
- 您可以在相同的声明中决定原始数据的合适范围。

您必须将自己的变量定义为特定数据类型。I/O 设备支持的最普通的变量是数字量和整型。Vijeo Citect 也支持实型、字符串、字节、bcd、长型和 longbcd 数据类型。

在定义了变量标签后，您可以使用它们执行以下操作：

- 在图形画面上显示对象。
- 存储趋势和分析数据。（参见 [趋势数据](#)。）
- 监视报警。（参见 [配置和处理报警](#)。）
- 控制设备和过程。（参见 [定义命令和控件](#)。）
- 将数据存储在内存中（参见 [配置本地变量](#)。）

I/O 设备支持的最普通变量是数字量和整型变量，尽管某些 I/O 设备支持其它的数字变量和字符串。

另请参见

[标签名称语法](#)

[使用结构化标签名称](#)

[配置本地变量](#)

[配置变量标签](#)

[格式化数字变量](#)

[使用阵列](#)

标签命名

Vijeo Citect 针对变量标签的名称设置了两项限制：

- 标签只能使用特定语法。参见 [标签名称语法](#)。
- 标签不能与工程或任何包含工程中的 Cicode 函数同名。如果标签与某 Cicode 函数同名并被置于图形画面上，编译时会出错。

此外，使用标签命名规则可使工程的设计、配置、授权和维护更为便捷和迅速。有关命名规则的建议，请参见 [使用结构化标签名称](#)。

标签名称语法

Vijeo Citect 标签（变量标签、报警标签和趋势标签）必须包含以下语法：

```
[<alpha> | '_' ] * [<alpha> | <digit> | '\' | '_' ]
```

也就是说，标签名称必须以字母字符（A-Z 或 a-z）或下划线字符（_）开头。后面的字符必须是字母字符（A-Z 或 a-z）、数字字符（0-9）、反斜杠字符（\）或下划线字符（_）。使用任何其它字符都将导致编译器出错。

例如，“_MyTag123”和“my\New\Tag”均为有效的字符名称，而“\NewTag\”则无效。

仅当 INI 参数 **[General]TagStartDigit** 被设置为 1（默认值为 0）时，以数字字符开头的标签名称（如“12TagName”）才有效。

注意：报警标签的名称必须遵循此语法，但报警名称则不需要。

使用结构化标签名称

建议在系统中使用以下命令约定，以便在使用某些功能（如精灵和超级精灵）时获得最大收益。（如果使用的命令系统与以下约定不同，则可以通过修改要使用的精灵来使用 Vijeo Citect 提供的精灵和超级精灵。）

每个标签名称最多可以包含 79 个字符。要建立约定，必须将标签名称中的字符划分为几个部分，用于描述标签的性质，例如，标签所在的区域、变量的类型和任何特定属性。建议对 Vijeo Citect 命名约定使用四个基本部分：

```
Area_Type_Occurrence_Attribute
```

区域

区域部分标识工厂区域、编号或名称。如果在一个特定的区域中使用前缀识别标签，则可以在这个区域很容易地复制所有的 Vijeo Citect 函数。例如，如果您有 3 个锅炉，每个锅炉有相同的控件，则可以为第一个锅炉配置一个标签，然后将该标签复制到锅炉 2 和 3。只需将标签名中的区域段改变为锅炉 2 和 3 的区域。标签的区域段保留不变，例如：

锅炉 1	锅炉 2	锅炉 3
B1_TIC_101_PV	B2_TIC_101_PV	B3_TIC_101_PV

如果不需要这个设备，则可以通过减少标签字符数来忽视“区域”段。

类型

类型段确定参数类型，过程设备或控制硬件。建议使用 ISA 标准命名系统。

变量标签	意义
B1_TIC_101_PV	温度指示控制器
B1_FIC_101_PV	流量指示控制器
B1_PUMP_101_PV	泵
B1_VALVE_101_PV	阀

事件

事件段确定回路数。

变量标签	意义
B1_TIC_101_PV	温度指示控制器 101
B1_TIC_102_PV	温度指示控制器 102
B1_PUMP_101_PV	泵 101
B1_PUMP_102_PV	泵 102

特征

属性段确定与回路有关的特征或特定参数。

变量标签	意义
B1_TIC_101_PV	过程变量
B1_TIC_101_SP	设置点
B1_TIC_101_OP	输出
B1_TIC_101_P	增益或比例带宽
B1_TIC_101_I	积分
B1_PUMP_101_CMD	启动泵的命令信号

B1_PUMP_101_M	自动/手动模式
B1_TIC_101_V	值(运行/停止)

推荐属性

Vijeo Citect 提供的精灵和超级精灵使用下面的属性协定。如果您遵循此协定，则可以使用精灵，而无需对其进行修改。

助记符	离散控制/监控	数据类型	范围
_CMD	启动设备的命令信号	数字量	0 = 关, 1 = 开
_M	控制模式	数字量	0=手动, 1=自动
_V	参数值	数字量	0=关闭, 1=开启
_FAIL	设备故障	数字量	1=良好, 0=故障
FAULT	设备故障	数字量	1=良好, 0=故障

助记符	过程报警	数据类型	范围
_ALM	通用报警	数字量	0=激活, 1=非激活
_HHALM	上限报警	数字量	0=激活, 1=非激活
_HALM	上限报警	数字量	0=激活, 1=非激活
_LALM	下限报警	数字量	0=激活, 1=非激活
_LLAM	下限报警	数字量	0=激活, 1=非激活
_DALM	偏差报警	数字量	0=激活, 1=非激活
_DLALM	偏差下限报警	数字量	0=激活, 1=非激活
_DHALM	偏差上限报警	数字量	0=激活, 1=非激活
_HHTRIP	上限报警差错点	模拟量	

助记符	过程报警	数据类型	范围
_HTRIP	上限报警差错点	模拟量	
_LTRIP	下限报警差错点	模拟量	
_LLTRIP	下限报警差错点	模拟量	
_DTRIP	偏差差错点	模拟量	
_LDTRIP	下限偏差差错点	模拟量	
_HDTRIP	上限偏差差错点	模拟量	
_HHyst	高高限报警过度	模拟量	
_Hhyst	高限报警过度	模拟量	
_Lhyst	低限报警过度	模拟量	
_LLhyst	低低限报警过度	模拟量	
_LDhyst	低限偏差报警过度	模拟量	
_HDhyst	高限偏差过度	模拟量	

助记符	逻辑控制/监控	数据类型	范围
_PV	过程变量	模拟量	
_SP	设置点	模拟量	
_RSP	远程设置点	模拟量	
_OP	输出	模拟量	
_OPM	输出模式	数字量	0=手动, 1=自动
_SPM	设置点模式	数字量	0=本地, 1=远程
_P	增益 (比例带宽)	模拟量	
_I	积分 (重新设置)	模拟量	
_D	微分 (比率/超前)	模拟量	
_KP	增益放大器	模拟量	

助记符	逻辑控制/监控	数据类型	范围
_KI	积分放大器	模拟量	
_KD	微分放大器	模拟量	
_SPTK	设置点追踪模式	数字量	0=关闭, 1=追踪
_OPTK	输出追踪模式	数字量	0=关闭, 1=追踪
_SPB	设置点偏差	模拟量	
_SPR	设置点比率	模拟量	
_DEV	偏差		
_TOT	累加器参数值	模拟量	
_COUNT	计数器参数值	模拟量	
_CRESET	计数器重设命令	数字量	0=计数, 1=重设
_CLIMIT	计数器事先调整限制	模拟量	
_TIME	定时器参数值	模拟量	
_TRESET	定时器重设命令	数字量	0=计时, 1=重设
_EXP	定时器过期	数字量	
_TLIMIT	定时器限制	模拟量	
_CALC1	计算结果 1	模拟量	
_LINZ1	线性化信号 1	模拟量	
_Q	数据品质标志	数字量	1=良好, 0=损坏

注意： 要缩短标签名称，可以省略下划线，但可能会对可读性造成影响，例如：
B1TIC101PV instead of B1_TIC_101_PV。

配置本地变量

当您启动运行时系统时，本地变量允许您在内存中存储数据。本地变量在系统每次启动时创建，因此当您关闭系统时本地变量不保留它们的值。本地变量可以是 Vijeo Citect 所支持的任何数据类型，包括除字符串以外的所有标准 Vijeo Citect 类型的二维数组。

当您需要每个过程具有单独的数据副本时，本地变量非常有用。对于在工程中配置的每个本地变量，每个过程都有其自己的副本，而且本地变量中的值仅对于写入这些值的过程可用。

如何配置本地变量：

1. 在“工程编辑器”中，选择**标签 | 本地变量**。将显示“本地变量”对话框。
2. 在**名称**域中，输入此本地变量的名称（最多 79 个字符）。变量名称不能包含 '-'、'/'、'%' 或 <空格> 字符。
3. 在**数据类型**域（16 个字符）中，选择下列支持的数据类型之一：

数据类型	变量	大小	允许值
BCD	以十进制为基础的 二进制代码	2 字节	0 到 9,999
BYTE	字节	1 字节	0 到 255
DIGITAL	数字量	1 位或 1 个字 节	0 或 1
INT	整型	2 字节	-32,768 到 32,767
UINT	无符号整型	2 字节	0 到 65,535
LONG	长整型	4 字节	-2,147,483,648 到 2,147,483,647
LONGBCD	长以十进制为基础 的二进制代码	4 字节	0 到 99,999,999
REAL	浮点	4 字节	-3.4E38 到 3.4E38
STRING	字符串	256 字节（最 大）	ASCII（空结尾）

数值和数字变量的缺省值为 0，字符串变量的缺省值为 ""（空字符串）。如果不指定数据类型，本地变量将被视作 16 位整数。

4. 在**数组大小**域（8 个字符）中，输入用于存储本地变量的数组（元素数目）的大小。数组将是**数据类型**域中指定的数据类型。数组可以是一维或二维。元素的最大数量是 32766，对于一个二维数组，这意味着第一个维数的大小乘以第二个维数的大小应该小于或等于 32766。当指定一个多维数组时，应使用逗号分隔维数，例如 "20,30"。

5. 在**零点刻度**域（11 个字符）中，输入表示数据零点的本地变量的值。零点刻度值用作趋势图和条形图的下限，零点刻度值以下的值将导致运行时系统中出现“超出范围”提醒信息。
6. 在**满刻度**域（11 个字符）中，输入表示满刻度数据点的本地变量的值。满刻度值用作趋势图和条形图的上限，满刻度值以上的值将导致运行时系统中出现“超出范围”提醒信息。
7. 在**工程单位**域中，输入此值表示的工程单位（例如 %、deg、mm/sec 等）。最多 8 个字符。这个属性是可选的。如果没有指定工程单位，则不使用工程单位。
8. 在**格式**域中，输入当一个变量值被显示在图形页上、写到文件中或传递到函数中（期望一个字符串）时，该值的显示格式。这个属性是可选的。如果没有指定格式，缺省格式是####.#。最多 11 个字符。
9. 在**注释**域中，输入有用的注释（最多 48 个字符）。该属性是可选的，不在运行时使用。
10. 点击**添加**。

另请参见

[配置变量标签](#)

配置变量标签

如何配置变量标签：

1. 启动 Citect 资源管理器。
2. 点击**变量标签**，或选择**标签 | 变量标签**。将显示“变量标签”表格。
3. 输入变量标签的属性。
4. 点击**增加**加入新的记录，或点击**替换**修改一个记录。

您至少应完成**变量标签名称**、**I/O 设备名称**、**数据类型**和**地址域**。

您可以将任何现有的变量标签粘贴到工程中的表格内。

如何选择一个现有的变量标签：

1. 打开“工程编辑器”。
2. 选择**编辑 | 粘贴标签**。

如何配置一个数字量标签：

1. 打开“工程编辑器”。
2. 点击**变量标签**，或选择**标签 | 变量标签**。
3. 完成出现的**变量标签**对话框中的属性，使用 **DIGITAL** 作为**数据类型**。
4. 点击**增加**加入新的记录，如果修改了记录，则可点击**替换**。

您必须至少完成**变量标签名称**、**I/O 设备名称**、**数据类型**和**地址域**。保留下列属性为空：

- 原始零点刻度、原始满刻度
- 工程零点刻度、工程满刻度
- 工程单位、格式

如何配置一个模拟量标签：

1. 启动“工程编辑器”。
2. 点击**变量标签**，或选择**标签 | 变量标签**。将显示“变量标签”表格。
3. 输入属性，使用 **INT**（或 Real、BCD、Long、LongBCD）作为**数据类型**。
4. 点击**增加**加入新的记录，或点击**替换**修改一个记录。

您必须至少完成**变量标签名称**、**I/O 设备名称**、**数据类型**和**地址域**。

另请参见

[变量标签属性](#)

[格式化数字变量](#)

变量标签属性

您可以针对 [配置变量标签](#) 使用此对话框。变量标签有下面的属性：

变量标签名

对于标签，您可以使用任何名称（79 个字符），只要它符合 [标签名称语法](#) 并且不与该工程或任何其他包括的工程中的 **Cicode** 函数的名称相同即可。如果具有许多标签，请使用命名约定（参见 [使用结构化标签名称](#)）。这样就可以更方便地查找和调试标签。

如果您使用的是分布式服务器，那么此名称对于集群必须是唯一的（例如，不能在多个集群中使用同一个变量标签名称）。

集群名称

对其应用标签的集群的名称（16 个字符）。从可用集群（在 [集群定义](#) 中定义）的列表中选择集群名称。

数据类型

I/O 设备变量的类型（16 个字符）。I/O 设备支持用于与 Vijeo Citect 交换数据的多种数据类型。因为缺乏工业标准，大多数 I/O 设备生产厂商采用各自的命名协定为它们的 I/O 设备变量命名。然而，所有的变量都符合下面的数据类型中的一种：

数据类型	变量	大小	允许的参数值
BCD	以十进制为基础的二进制代码	2 字节	0 到 9,999
BYTE	字节	1 字节	0 到 255
DIGITAL	数字量	1 位或 1 字节	0 或 1
INT	整型	2 字节	-32,768 到 32,767
UINT	无符号整型	2 字节	0 到 65,535

LONG	长整型	4 字节	-2,147,483,648 到 2,147,483,647
ULONG	无符号长整数 (仅用于在屏幕上显示。算术运算不受支持。)	4 字节	0 到 4,294,967,295
LONGBCD	长以十进制为基础的二进制代码	4 字节	0 到 99,999,999
REAL	浮点	4 字节	-3.4E38 到 3.4E38
STRING	字符串	256 字节 (最大)	ASCII (空结尾)

注意：如果没有指定标签的范围，那么当您写入一个超出类型范围的值时，将产生一个超出范围提醒信息。

您必须指定正确的符合正在配置的 I/O 设备变量数据类型的数据类型。每一个数据类型有唯一的地址格式。当定义变量地址时必须使用这个格式（象地址属性）。

需要确认只使用对于您的 I/O 设备有效的数据类型。如果不指定数据类型，变量将被视作 16 位整数。

Vijeo Citect 支持 I/O 设备寄存器串联。例如，您可以定义一个实数数据类型（在 Vijeo Citect 中）作为两个相邻的整数数据类型（在 I/O 设备中）。Vijeo Citect 在两个整数的边界上读取并返回一个实数。

如果使用寄存器串联，变量地址必须是奇边界或偶边界。不能混淆边界。例如，V1、V3、V5 是有效的地址。

使用此功能时小心：I/O 设备必须保持第二个寄存器的完整。（如果 I/O 设备写入第二个整数，实数的值将损坏。）某些 I/O 设备的结构可能不支持此功能，因而可能发生不可预知的行为。



警告

不希望的设备操作

在串联寄存器时请勿混合使用奇、偶变量地址作为边界。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

字符串变量

数字变量是最普遍的，但许多 I/O 设备也支持 ASCII 字符串。可以使用字符串存储文本数据（例如，从条形码读取器上）。

I/O 设备中的所有字符串必须以 NULL 结尾。Vijeo Citect 使用 NULL 字符检查字符串结尾，如果没有 NULL 字符，Vijeo Citect 将读取（并且显示）字符串结尾后内存中的任意额外字符。

当您使用磁盘 I/O 设备时，也可以为处方存储或操作人员显示信息指定一个字符串数据类型。

不是所有的 I/O 设备都支持字符串。然而，如果您的 I/O 设备支持整数数据类型，Vijeo Citect 可以使用这些整数寄存器存储 I/O 设备中的 ASCII 字符串。Vijeo Citect 字符串只能存储在相邻块（连续寄存器）中，并被存储成为数组。

要显示 I/O 设备的数据类型，从**帮助**中双击 I/O 设备书，从列表中选择 I/O 设备，然后选择**数据类型**主题。

I/O 设备名

变量存储所在的 I/O 设备的名称（31 个字符）。如果正在使用 I/O 设备冗余，必须在这里定义主 I/O 设备名，而不是备用设备。

地址

变量被存储的 I/O 设备中的寄存器地址（254 个字符）。地址的格式和前缀取决于为 I/O 设备配置的协议，可通过检查工程编辑器中的“I/O 设备”表单上的**协议**字段来确定。

原始零点刻度/原始满刻度

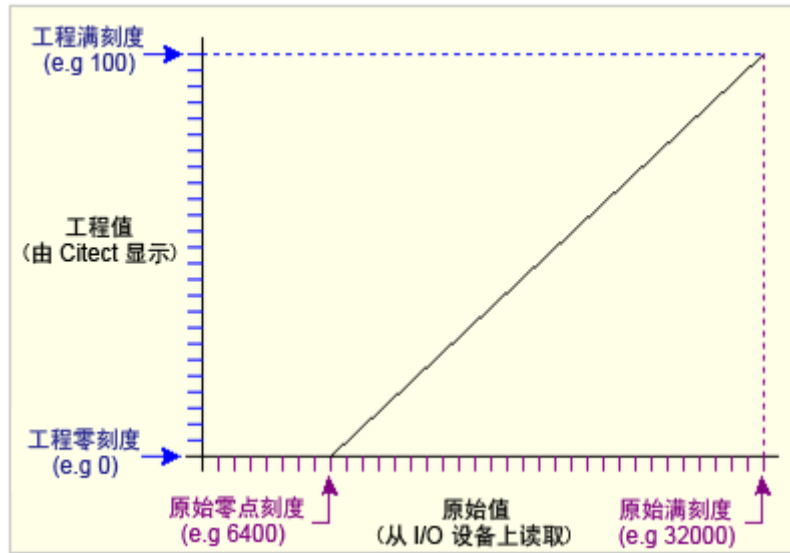
代表数据零点和满刻度点的未处理的（原始的）值（11 个字符）。原始值是 Vijeo Citect 从 I/O 设备中读取的值。

工程零点刻度/工程满刻度

Vijeo Citect 从原始值计算的刻度值（11 个字符）。原始零刻度被刻度到工程零刻度，原始满刻度被刻度到工程满刻度。这些属性表现为工程单位，并且用于趋势和棒装图的上下限。

大多数 I/O 设备返回一个整型值显示模拟量输入的值。为了返回一个有用的值，I/O 设备转换一个输入信号（通常 4-20 毫安）为一个原始刻度值，通常（不总是）范围在 6400 到 32000。

要将此值显示为有意义的值，您可以指定一个比例计算。Vijeo Citect 然后将据此调整所有值，如下图所示。



变量被调整刻度的值（工程值），不是它的原始值，被使用在系统中。

刻度属性是可选的。如果没有定义刻度，工程零点刻度缺省是原始零点刻度，工程满刻度缺省是原始满刻度；也就是没有刻度。

在运行时系统中，定义低于原始零点刻度或高于原始满刻度的值会产生“超出范围”提醒信息。

工程单位

（8个字符。）值代表的工程单位（例如，%、deg、mm/sec等）。此属性为可选属性。如果没有指定工程单位，则不使用工程单位。

格式

（11个字符）。当它被显示在图形画面上、写到文件中或传递到函数中（期望一个字符串）时，变量值的显示格式。此属性为可选属性。如果没有指定格式，缺省格式是####.#。

死区

（11个字符）。死区是变量标签工程范围的百分比，要使更新能通过系统发送，标签必须以此范围为依据进行更改。缺省值是0。

例如，如果一个变量标签的工程零点刻度是0，工程满刻度是10000，请将死区设为1%。如果标签的当前值是5600，那么PLC中标签的值必须更改为5700以上或5500以下，然后更新将通过系统发送。

注释

任何有用的注释（48个字符）。

链接

“变量标签”表单的状态行中的**链接**字段将读取**是或否**，并且指示变量标签是否链接到外部数据源。当使用除 Vijeo Citect 以外的软件编程 I/O 设备时，一个外部数据源用于存储标签数据。当外部标签数据改变时，链接的变量标签更新，意味着不需要在 Vijeo Citect 中再输入信息。

格式化数字变量

数字变量（数字）的值可以显示在图形画面上，或以多种格式（例如，24、0024、24.000 或 24.0%）写入到文件中。

格式定义符

格式定义符是用于数字变量定义的键盘字符。可以使用下面的定义符：

定义符	描述	功能
#	无用字符	显示的字符数
0	零	填充
-	负	对齐
.	句点	十进位计数法
EU		工程单位
S		指数计数法

指定数字量数目

井号 (#) 指定 Vijeo Citect 显示多少数字量。所有数字变量显示在动点右侧，例如：

格式：####

在这个例子中，Vijeo Citect 在动点的右侧显示至少 4 个数字（或空格）。如果数字包含不只 4 位，第一位位于动点处。下图以上面的格式显示了几个数字。


```
+ 5
+ 75
+1275
+5731275
  ↑
  动态点 (AN)
```

以零填充

当某数字包含的数字量比您的格式指定的数字量少（上例中的 5 或 75），Vijeo Citect 仅显示有意义的数字量。您可以使用填充字符零 (0) 作为格式字符串的第二个字符，以用零填充数字，例如：

Format:#0##

此格式字符串显示：

```
+0005
+0075
+ 1275
+5731275
  ↑
  动态点 (AN)
```

改变对齐方式

缺省情况下，数字变量（在其字段中）是右对齐的。可通过将减号(-)作为格式字符串中的第二个字符来改变这一缺省对齐方式，例如：

格式：#-###

此格式字符串显示：

+5
+75
+1275
+5731275
↑
动态点 (AN)

指定十进位计数法

要指定十进位计数法，请使用句点(.)，例如：

格式：###.##

在这个例子中，Vijeo Citect 显示小数点前的 3 个数字，小数点后的 2 个数字。如果变量是 12.3，Vijeo Citect 将显示 12.30。

所有数字自动进位，也就是说，12.306 将显示为 12.31。

指定工程单位

可以在定义变量标签时指定工程单位（如 %、deg、rpm、M、mm/sec 等）。为了在数字格式中包含这些单位，在合适的位置键入 EU。

格式：####.##EU

注意：如果没有为变量指定工程单位，将只显示（或记录）数字。

指定指数计数法

要指定指数计数法，需提供指数字符 (s)，例如：

格式：#s###

在这个例子中，Vijeo Citect 以指数计数法格式显示数字，例如：1.234e+012。

综合格式定义符

你可以合并使用格式定义符，例如：

格式：#0-###.##EU

这个格式字符串在小数点前显示 6 位，在小数点后显示 2 位。左对齐，填充并且显示工程单位（如果定义）。

使用短格式符号

可以使用短格式符号作为无用 (#) 符号的备用符号。对于短符号，可以使用数字指定数字变量的格式，例如：

格式：3.2

此格式字符串在小数点前显示三位，在小数点后显示两位。此格式字符串相当于 ###.## 格式规范。也可以包含带有短格式符号的工程单位，例如：

格式：6.0EU

此格式字符串在小数点前显示六位，在小数点后无位数显示。数字随工程单位一同显示（如果已指定）。此格式字符串相当于 #####EU 格式规范。

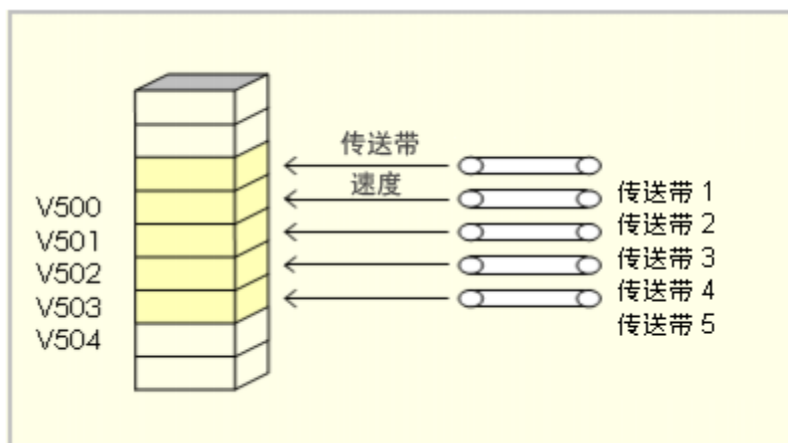
不能对短格式符号使用填充或左对齐。

注意： 如果数字变量的参数值非常大，则会以指数计数法显示（例如 1.2345E012）。如果没有为变量指定格式，则使用缺省格式 ####.#（或 4.1）。

另请参见
[使用数组](#)

使用数组

数组是存储在 I/O 设备连续内存寄存器中的变量（所有相同数据类型）集合。使用单独的设备（或过程）执行类似功能时，数字数组非常有用。在连续内存寄存器中，可以对要存储变量（与每个过程相关）的 I/O 设备进行编程，例如：



此处，五个连续变量（V500、V501、V502、V503 和 V504）用于存储五个传送带（1 到 5）的速度。可以配置一个标签作为数组，而不是配置五个单独的变量标签（每个传送带一个）。

要为数组定义一个变量标签，请定义第一个地址并将数组大小（连续地址的数目）添加到寄存器地址，例如：

变量标签名称	Conveyor_Speed
地址	V500[5]

此处，五个寄存器地址由变量标签 Conveyor_Speed 引用。

引用阵列元素

阵列的每个元素都通过索引进行引用。您可以通过指定标签名和索引从阵列中获得每一个变量：

```
<Tag Name>[ Index ]
```

例如，要引用上面例子中阵列的第三个变量（传送带 3），使用下面的语法：

变量标签	Conveyor_Speed[2]
------	-------------------

阵列第一个元素的索引是 0（零）。在此列中，Conveyor_Speed[0] 是阵列的第一个元素（传送带 1），Conveyor_Speed[4] 是最后一个元素（传送带 5）。

使用阵列时，请注意以下各项：

- 不要定义大阵列，因为每次请求一个阵列元素时，Vijeo Citect 将从 I/O 设备中读取整个阵列。对于大阵列，系统性能可能降低。
- 阵列的大小必须小于协议的最大请求长度。I/O 设备描述帮助主题显示了协议的最大请求长度。
- 您应以 16 位为界限，声明所有数字量阵列。Vijeo Citect 将每个数字量阵列舍入为最近的 16 位界限。因此，阵列中的所有元素都将改变。例如，如果您声明一个阵列 Test 在 35 位开始，Vijeo Citect 将在 32 位开始阵列。阵列索引也从 32 位开始，Test[0] 是 32 位，而不是 35 位。

字符串阵列

如果您使用 Vijeo Citect 字符串数据类型，则必须定义由整型数据类型组成的阵列。一个 int 寄存器可存储两个字符串字符。

要计算阵列的大小（对于字符串数据类型），使用下面的公式：

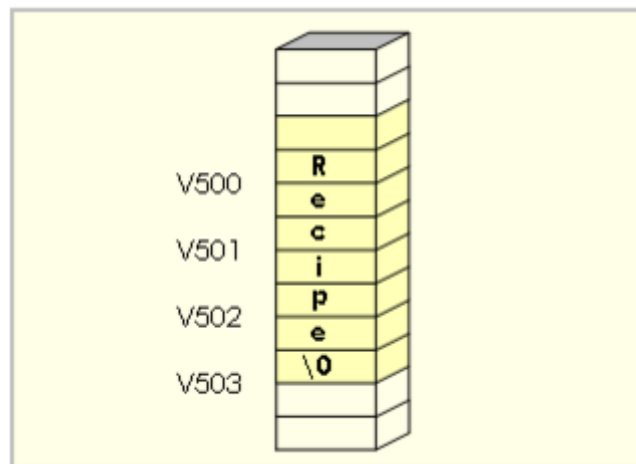
$$\text{数组大小} = \frac{(\text{字符个数} + 1)}{2}$$

数组的最后一个元素始终用于存储空字符 '\0'。该字符标记字符串的结束。

要存储“Recipe”，您必须定义 4 个元素的阵列，例如：

变量标签名	Recipe_Tag
地址	V500[4]

每个寄存器存储两个字符，也就是说：



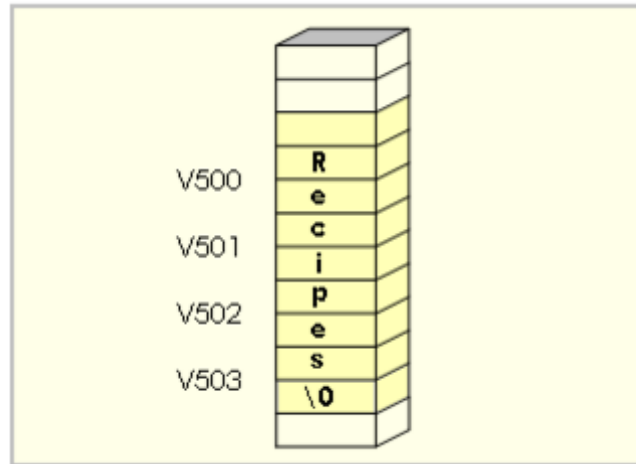
您可以通过定义标签引用整个字符串：

```
<Tag Name>
```

例如：

变量标签	Recipe_Tag
------	------------

要存储“Recipes”，您应该定义 4 个元素的阵列。字符被存储如下：



注意：如果您的 I/O 设备支持字符串数据类型，您必须采用由正使用的 I/O 设备确定的格式指定地址。

另请参见
[使用结构化标签名称](#)

Chapter: 17 链接、导入和导出标签

由于经常独立于 Vijeo Citect 对 I/O 设备进行编程，因此您可以导入或链接到外部数据源中的所有标签。这意味着当您为 I/O 设备进行编程时，只需要定义一次标签信息。而不必在 Vijeo Citect 中重复输入相同信息。由于所需的信息已经保存在外部数据源中，因此只需要导入它或与之链接即可。

Vijeo Citect 变量数据库中有两类标签 - 链接标签和非链接标签。在从 Citect 浏览器中执行标签导入时，只有非链接标签被更新。链接标签保留不变。在执行刷新或自动刷新时，只有链接标签被更新。非链接标签保留不变。在执行同步（实时更新）时，I/O 设备的所有标签都被更新和链接。

Vijeo Citect 还允许将标签导出到您指定了目标位置和格式的外部文件。然后，可以将此文件导入第三方 I/O 设备编程数据库，或只是将其作为一个备份。

注意：只有在通过 OFS 导入时才支持标签名称及其字段的同时更新。在任何其他情况下，如果尝试同时更新标签名称和字段，则只有名称被更新。这种情况下，只能使用单独的操作分别更新标签名称和字段。

另请参见

[链接标签](#)

[导入标签](#)

[导出标签](#)

[联合支持矩阵](#)

链接标签

链接是 I/O 设备特有的操作。当您在 Vijeo Citect 中（使用快速通信向导）添加一个 I/O 设备记录时，可以选择将它链接到外部数据源。外部数据源是对实际 I/O 设备进行编程时保存所有标签数据的地方。如果链接到外部数据源，Vijeo Citect 会自动为 I/O 设备中的每个标签创建标签变量记录。

这些标签变量记录动态链接到外部数据源的标签上。当外部标签改变时，Vijeo Citect 将更新它们。例如，您可以对 I/O 设备进行编程，配置工程，然后增加新标签或编辑已有标签。在这种情况下，Vijeo Citect 会自动更新您的更改。

这种更新发生在您试图读取 Vijeo Citect 中的改变标签时（例如，您编译工程，使用“标签变量”对话框显示标签，修改或粘贴标签，或执行手动刷新等）。例如，如果您使用第三方 I/O 设备编程包改变了标签的地址，外部数据源也会随之改变。然后，当您显示标签变量记录或在 Vijeo Citect 中编译工程时，该改变将从外部数据源复制到 Vijeo Citect 的标签变量数据库。

您可以知道标签是否链接，因为“标签变量”表单底部的状态行将显示**链接:是**。如果标签处于链接状态而且您更改了某个字段，则下次刷新链接时将不会覆盖您的更改。不过，一般来说，将会禁用其值取自外部数据源的任何字段。

注意：为外部标签定义的某些属性将不与 Vijeo Citect 相关。而且，有些也不是 Vijeo Citect 可以读取的格式。每个 I/O 设备在 Vijeo Citect 中都有一个关联的格式文件。就是这个文件确定了什么信息被复制到 Vijeo Citect 的标签变量数据库中，以及这些信息如何转换。在大多数情况下，您不必编辑此文件。

对于链接到 Unity 标签的 Citect 标签，请注意以下事项：

1. 无论在哪一方增加新标签，都将导致另一方增加新标签。
2. 无论在哪一方删除现有标签，都将导致另一方删除相应的标签。
3. 无论在哪一方修改现有标签，都将导致另一方修改相应的标签。
4. Unity 或 Citect 中的删除操作将优先于其他操作（如修改）。
5. 如果同一标签在 Unity 数据库与 Citect 数据库之间存在某种竞争，则同时从两个数据库进行修改，Unity 数据库项将优先于 Citect 数据库项。
6. 在下列情况下，将触发链接 I/O 设备实时更新同步：
 1. Unity 配置更新。
 2. Citect 配置更新。
 3. 手动刷新。

有关 Unity 与 Vijeo Citect 之间支持级别的信息，请参见 [链接的标签](#)。

断开与外部数据源的链接

可以从 I/O 设备视窗中或通过快速通讯向导断开与外部数据源的链接。如果断开连接，可以选择制作所有标签的本地副本，也可以将它们统统删除。

删除 I/O 设备

如果删除标记为链接的 I/O 设备记录，您可以选择为所有已链接标签创建本地副本或只将其删除。

要链接到外部数据源中的标签，请执行以下操作：

1. 启动“工程编辑器”。
2. 选择**通讯 | I/O 设备**。
3. 滚动浏览到相关 I/O 设备，然后选择**已链接 | True**。
4. 根据需要填写其它域。

要使用快速通信向导链接到外部数据源中的标签，请执行以下操作：

1. 启动工程编辑器并选择**通讯 | 快速向导**。（此外，可以打开 Citect 工程管理器，然后点击当前工程的**通讯**文件夹中的**快速 I/O 设备设置**。）
2. 逐个完成向导屏幕，选择相关 I/O 设备等。当**链接到外部数据库**屏幕显示时，选中**将 I/O 设备链接到外部标签数据库**复选框，然后填写剩余的详细信息。

要手动刷新链接的标签，请执行以下操作：

1. 打开 Citect 工程管理器。
2. 选择工具 | **刷新链接标签**，显示**刷新链接标签**复选框。

另请参见

[刷新链接标签属性](#)

[导入标签](#)

刷新链接标签属性

使用此对话框可以刷新 [链接标签](#)。此对话框包含以下字段：

选择链接的 I/O 设备

工程中（和包含工程）每个链接的 I/O 设备都将显示在这里。要刷新某个 I/O 设备的标签，点击该 I/O 设备，然后点击**刷新**。这将使用所选 I/O 设备的最新标签参数值来更新您的工程。

如果您使用 Vijeo Citect 修改任何 I/O 设备标签，在刷新时，您的修改将不会被覆盖。

导入标签

从外部数据源导入标签，可以减少数据输入时间。当你编程 I/O 设备时，不用再输入所有的标签信息，一旦当你配置工程时，你可以编程 I/O 设备，然后将标签导入到 Vijeo Citect，在此，可以将它们作为普通标签使用。（Vijeo Citect 会自动为 I/O 设备中的每个标签创建标签变量记录。）

注意： Vijeo Citect 仅支持使用字母数字字符、反斜线和下划线字符（请参见 [标签名称语法](#)）。其他任何字符都将在导入时转化成下划线。这在编译时会导致“重复标签”错误。

与连接一样，导入标签是特定于 I/O 设备的操作：导入特定 I/O 设备的所有标签。然而，与连接不一样的是，导入标签记录不以任何方式连接到外部数据源的标签上。因此，导入通常是一次性操作。要更新导入的数据，你必须再次导入它们。

注意： 标签将导入到在其中定义选定 I/O 设备的工程中。它可以是在浏览器中选择的工程，也可以在它的一个包括工程中。

对于大多数外部数据源，导入过程包括两步。第一步是将数据从 I/O 设备导出为 Vijeo Citect 能够读取的格式，然后再将数据库导入到 Vijeo Citect 中。不过，使用 Unity Pro 保存的标签数据可以由 Vijeo Citect 直接读取。这意味着不需要导出。

某些导入字符在 Config 目录的格式文件中定义（带扩展名 .fmt）。每种数据库类型都有一个格式文件，而每个文件都指定了如何将外部数据转化成 Vijeo Citect 变量数据。通常，导入的标签如果不存在，则会将它添加到 Vijeo Citect 变量数据库；如果已经存在，则进行更

新。不过，如果某个字段已经在格式文件（对应于导入数据库类型）的 [EditableFields] 部分中列出，则：

- 如果该标签已经存在，该标签的 Vijeo Citect 字段不会用外部值进行更新。
- 如果该标签尚未存在，该标签的 Vijeo Citect 字段将用外部值进行更新。

例如，如果某个标签有 10 个字段要导入，而其中的 5 个字段已在格式文件的 [EditableFields] 部分中列出，则如果该标签已经存在于 Vijeo Citect，该标签将用外部值（[EditableFields] 中列出的 5 个字段）进行更新。如果该标签尚未存在，则会用外部值（全部 10 个字段）来创建该标签。

以这种方式标记为可编辑的字段可由 Vijeo Citect 进行更改。为了防止内部更改丢失，在外部数据库中对这些字段的更改都将被忽略。

注意：如果外部数据库支持 LiveUpdate，且已经为 I/O 设备选择了 **LiveUpdate 链接**，则现有的 Vijeo Citect 标签将与外部数据库同步，而且标签将用导入的字段进行更新，无论它是否被标记为可编辑字段。被配置为 **已链接** 或 **自动刷新链接** 的 I/O 设备将按正常导入操作，而不会更新被标记为可编辑的字段。

在将标签导入 Vijeo Citect 时，有两种选项可用来处理现有的标签：

- 导入之前删除与 I/O 设备关联的所有标签。
- 导入时更新已存在的标签。在 Vijeo Citect 和在外数据源中找到的标签在 Vijeo Citect 中进行更新。存在于 Vijeo Citect 中但不存在于外部数据源中的标签将保持不动。附加所有新标签。

如果导入已经连接的数据源，则会复制该数据源中的所有标签。即，每个标签将有两个副本：一个本地的，一个连接的。

请参见 [导入的标签](#) 获取有关 Unity 与 Vijeo Citect 之间的支持级别的信息。

为外部标签定义的某些属性将与 Vijeo Citect 无关。而且，有些也不是 Vijeo Citect 可以读取的格式。要定义将哪些信息复制到 Vijeo Citect 的标签变量库以及如何转换这些信息，您必须编辑 I/O 设备的格式文件。

从外部数据源导入标签变量：

1. 打开 Citect 浏览器。
2. 选择工具 | 导入标签。
3. 根据需要完成 **导入标签变量** 对话框。

另请参见

[导入标签变量对话框属性](#)

导入标签变量对话框属性

使用此对话框可从外部数据源 [导入标签](#)。

注意：如果 I/O 设备连接到外部数据源，“数据库类型”、“外部数据库”、“连接字符串”和“标签前缀”字段将显示为灰。

“导入标签变量”对话框有以下字段：

I/O 设备

将为其导入标签的 I/O 设备。使用菜单选择已经使用 Vijeo Citect 定义了的 I/O 设备。

注意：标签将导入到在其中定义选定 I/O 设备的工程中。它可以是在浏览器中选择的工程，也可以在它的一个包括工程中。

数据库类型

外部数据源参考的数据格式。

外部数据库（128 个字符）

参考外部数据库的来源。单击“浏览”按钮可导航到文件或指定配置选项。外部数据库可以是：

- 直接的路径和文件（例如 C:\Data\Tags.csv）
- IP 地址和节点（例如 127.0.0.1\HMI_Scada）
- URL（例如 <http://www.abicom.com.au/main/scada>）
- 计算机名（例如 \\coms\data\scada）

标签数据可以直接从 Unity SpeedLink 或 OPC 工厂服务器 数据源。单击浏览按钮可指定配置选项。请参见 [OPC 工厂服务器标签导入](#) 或 [Unity 链接标签导入](#)。

连接字符串（128 个字符）

数据源的连接详细信息。这类似于 ODBC 连接字符串。例如：

```
UserID = XXX; Password = YYY
```

或

```
ServerNode=111.2.3.44; Branch=XXX
```

仅某些驱动程序需要连接字符串：

- Unity SpeedLink（静态/动态）
- OPC

选择这些驱动程序中的一个然后浏览外部数据源将在需要之处自动填充“连接字符串”字段。

给导入的标签添加前缀

选中此框可在 *Variable*.DBF 中的所有导入标签的名称前面插入前缀。

标签前缀

在标签变量数据库中所有导入标签的名称前插入的前缀（最多 8 个字符）。

导入前删除所有的 I/O 设备标签

选中此框将在导入之前（从标签变量数据库中）删除所有 I/O 设备的标签。

如果不选中此复选框，存在于 Vijeo Citect 中和外部数据源中的标签在 Vijeo Citect 中更新。存在于 Vijeo Citect 中但不存在于外部数据源中的标签保持不动。将附加新标签。

清除数据源中不存在的已删除的标签

选中此框将删除已经从外部数据库中移除的标签。换句话说，如果一个标签仍然是您工程中 I/O 设备的一部分，但不在外部数据库中，它将从工程中删除。

Unity 链接标签导入

“联合链接配置”对话框指定用于生成警报和趋势标签的参数。它包含以下字段：

I/O 设备和协议驱动程序 目标 I/O 设备的名称及其相应协议。Unity SpeedLink 使用 MODBUS、MODNET、MBPLUS、UNITE 或 OPC（不支持 OPC1）仅将标签导入到 I/O 设备。

联合数据库类型 可从“导入变量标签”对话框中选择导入/导出数据库驱动程序类型。

PLC 系列 您要从其导入的 PLC 的系列名称。可以是 Premium 或 Quantum。此选项仅对 Unity SpeedLink 静态协议有效。

联合文件 您要从其导入标签的联合工程文件的路径。如何数据库类型为 Unity SpeedLink 动态，则为 *.stu 文件。如何数据库类型为 Unity SpeedLink 静态，则为 *.xsy 文件。

启用联合配置文件 如果用户配置文件限制联合数据库，且需要用户名和密码才能进行访问，则启用此选项。请参见 Unity Pro 在线帮助的“访问安全管理”部分，以获取更多信息。

联合用户名和密码 从其导入的联合数据库的用户名和密码（如果需要）。

启用 DCOM 如果数据库主机为 DCOM 服务器，则启用此选项。

DCOM 服务器名 如果启用，客户端用于宿主从其解压缩标签的数据库的 DCOM 服务器。

DCOM 用户名和密码 如果启用，DCOM 服务器的用户名和密码。

启用标签生成配置 启用此选项可自动从变量标签数据库生成标签。使用 XML 模板文件中指定的规则创建标签。

配置模板另存为指定生成标签规则的 [TagGen XML 模板](#) 文件的文件名。点击**配置**，选择并配置 XML 文件。请参见 [附加标签生成配置](#)。

日志文件路径导入或导出过程时生成的日志文件的路径名。

日志记录级别导入和导出日志中所需的详细级别。

日志池大小指定给定设备的最大日志文件数量。

验证点击“验证”按钮，可检查联合链接设置是否正确。此按钮可启用画面上的“确定”按钮。

附加标签生成配置

此对话框指定 XML 模板的路径、在该模板中指定的参数，并显示模板说明（从模板中使用的 desc 属性中提取）。

1. 点击**浏览**以选择要使用的 XML 文件。
2. 在参数列表中双击某个参数以更改其值。
3. 点击**确定**以将此模板保存到工程中。该模板将被复制到工程中，并在验证时使用 taggen_import_ioDeviceName.xml 或 taggen_link_ioDeviceName.xml 文件名进行保存。

注意：返回此对话框时，路径名将指向工程中的模板的位置，而不是原始模板文件的路径。

OPC 数据访问服务器标签浏览器

OPC 数据访问服务器标签浏览器对话框在 Vijeo Citect 中生成 OPC 数据访问服务器数据库驱动程序要求的字符串。

OPC 数据访问服务器标签浏览器对话框具有以下域：

计算机名

OPC 服务器（可以是一个 IP 地址）或网络路径服务器的位置。将此域留空可选择本地计算机。

数据库树形显示

显示网络上所有正运行 OPC 服务器应用程序的服务器，并且显示每个服务器的分等级数据节点结构。

从可利用的数据库中选择一组标签去输入。OPC 数据可以是树状（分级的）或平面结构。Vijeo Citect OPC 驱动程序支持访问这两种类型的存储。如果数据位于树状结构中，则可以从根节点向下访问第一层的数据。将来可能支持更深的分支层。

注意：对于由合适的数据库管理员分配的读/写访问权，身份验证值必须是有效的。

OPC 工厂服务器 (OFS) 标签导入

Vijeo Citect 现在可以从 OPC 工厂服务器中导入标签。单击“浏览”按钮可指定数据库的位置。在显示的“OPC 数据访问服务器参数”对话框中，您可以浏览安装在 PC 上的 OFS 服务器并选择将要导入 Vijeo Citect 的标签。

注意，在从 OFS 导入标签时有某些限制和事情应加以注意。有关详细信息，请参见 [OFS 标签导入 限制](#)。

注意： 导入标签时有两种模式可用在 OPC 工厂服务器上：设备模式和 OFS 模拟器模式。如果使用设备模式，导入标签所花的时间将显著增长。这是因为对于 Vijeo Citect 所请求的每一个标签，OPC 工厂服务器必须在设备中验证该项。强烈建议使用 OFS 模拟器模式以避免这些性能问题。

计算机名

OPC 工厂服务器（可以是一个 IP 地址或 UNC 路径）的位置。将此域留空可选择本地计算机。

选择要导入的服务器和分支

显示网络上所有正运行 OPC 工厂服务器应用程序的服务器，并且显示每个服务器的分等级数据节点结构。

从可利用的数据库中选择一组标签去输入。OPC 数据可以是树状（分级的）或平面结构。Vijeo Citect OPC 驱动程序支持访问这两种存储类型。如果数据位于树状结构中，则可以从根节点向下访问第一层的数据。将来可能支持更深的分支层。

注意： 对于由合适的数据库管理员分配的读/写访问权，身份验证值必须是有效的。

标签生成配置启用

启用此项可自动从标签数据库中生成标签。使用 XML 模板文件中指定的规则创建标签。

已配置的模板另存为

“标签生成 XML 模板”文件（指定用于生成标签的规则）的文件名。点击**配置**，选择并配置 XML 文件。请参见 [附加标签生成配置](#)。

日志文件路径

导入或导出过程时生成的日志文件的路径名。

日志级别

导入和导出日志中所需的详细级别。

日志池大小

指定用于给定设备的日志文件的最大数量。

OPC 工厂服务器 (OFS) 标签导入 限制

从 OFS 数据库中导入标签时请注意下列限制。

- 对于每个导入的标签，Vijeo Citect 标签名称除 "." 字符将被 "_" 字符取代之外，其余部分与 OFS 上定义的标签名称相同。
- 如果启用了用户帐户控制 (UAC)，则在 Windows Vista 中使用 OFS 标签导入功能从 OFS 数据库中导入标签可能会不成功。UAC 禁止对外部数据库的访问。如果要通过 Windows Vista 中的 OFS 导入，请关闭 UAC，或者在 OFS 和 Unity Pro Windows 快捷方式上右键单击并选择“作为管理员运行”。
- 要生成报警和趋势标签，可在基本数据类型 (EDT) 的自定义和注释字段中设置一个唯一的标识符。例如，使用字符串 "VJA" 来标识用户要用来生成对应数字报警标签的标签。此功能在导出数据类型 (DDT) 和导出功能块 (DFB) 标签上只能使用“注释”字段。
- OFS 内置标签不能导入。只能在 Vijeo Citect 中手动添加它们。
- 在 Vijeo Citect 标签形式中，注释字段的长度限制为 48 个字符。
- Vijeo Citect 不支持 DDT 或 DFB 结构类型。OFS 导入功能将枚举此类标签的所有元素并将其作为基本标签导入。
- 仅当 OFS 与 .STU 和 .XVM 文件关联时才支持 OFS 标签导入功能。
- 下表显示了当 OFS 与 *.stu 或 *.xvm 文件关联后标签导入功能和 TagGen 所支持的数据类型。

数据类型	OFS 与 *.stu 文件关联			OFS 与 *.xvm 文件关联	
	标签导入	标签生成	自定义字段。	标签导入	标签生成
基本数据类型 (EDT)					
BOOL	是	是	是	是	是
EBOOL	是	是	是	是	是
BYTE	是	是	是	是	是
INT	是	是	是	是	是
DINT	是	是	是	是	是
WORD	是	是	是	是	是
DWORD	是	是	是	是	是
UINT	是	是	是	是	是
UDINT	是	是	是	是	是
REAL	是	是	是	是	是
STRING	是	是	是	是	是
DATE	是	是	是	是	是
DT	是	是	是	是	是
TIME	是	是	是	是	是

数据类型	OFS 与 *.stu 文件关联			OFS 与 *.xvm 文件关联	
	标签导入	标签生成 注释字段	自定义 字段。	标签导入	标签生成 仅限注释字 段
TOD	是	是	是	是	是
导出数据类型 (DDT)					
DDT 中的 EDT	是	是 ²	否	是	是 ³
DDT 中的阵列	是	是 ²	否	是	否
数组					
EDT 的阵列	是	是	否	是	否
DDT 的阵列	否	否	否	是	是 ¹
功能块类型 (FBT)					
基本功能块 (EFB)	是	是 ²	否	是	是 ³
导出功能块 (DFB)	是	是 ²	否	是	是 ³
同步					
实时更新	否	否	否	否	否
自动刷新标签	否	否	否	否	否

1. 当 OFS 与 *.xvm 文件关联时，只有 DDT 中的 BOOL 类型标签可用来生成报警和趋势标签。
2. 当 OFS 与 stu 文件关联时，标签实例中定义的注释字符串将被采用，而在 DDT/DFB 定义字段中定义的注释字符串将被忽略。
3. 当 OFS 与 xvm 文件关联时，在 DDT/DFB 定义字段中定义的注释字符串将被采用，而在标签实例中定义的注释字符串将被忽略。

TagGen XML 模板

TagGen 是一种 XML 文件，此文件使用专用标签和属性指定输入和输出数据库的域，并定义通过现有数据库域创建标签的过滤器和转换规则。

XML 文件的基本结构如下：

```
<?xml version="1.0"?>
<template>
<param name="parameter">
</param>
<input name="variable" file="variable.dbf">
</input>
```



```
<output name="trend" file="trend.dbf" filter="">
</output>
</template>
```

此概要模板未指定任何函数，它提取来自 `variable.dbf` 数据库的输入并将结果输出到 `trend.dbf` 数据库。`<template>` 标签是模板文档的根标签，必须一直显示。

`<template>` 标签包含三个段：

- `<param>`

这是一个可选段，您可以使用该段指定可供模板其它段引用的字符串常量。例如：

```
<param name="parameters">
<string name="MyIODevice">DISK_PLC</string>
</param>
```

在此范例中，变量 `MyIODevice` 被提供了值 `DISK_PLC`。

请注意，`<param>` 标签具有 `name` 属性。只要名称可以唯一标识 XML 文件内的段，就可以使用该名称。在引用文件内其它段中的变量时使用此段名称。参见 [引用变量](#)。

模板被选中时，此处定义的字符串显示在 [附加标签生成配置](#) 对话框中。可以在模板中定义尚未使用的字符串，以指定版本和修订信息，进而对版本修改进行跟踪。

- `<input>`

`<input>` 段指定数据库的名称和用于处理输入标签的域。XML 文件中必须有至少一个输入段。例如：

```
<input name="variable" file="variable.dbf">
<field name="name"></field>
<field name="type"></field>
<field name="unit" load="true">{parameter.IODevice}</field>
<field name="custom"></field>
<array name="taginfo">{ToProperty('{custom}', '=', ';')}</
array>
<string name="alarminfo">{variable.taginfo[VJA]}</string>
<string name="trendinfo">{variable.taginfo[VJT]}</string>
</input>
```

此示例标识 `variable.dbf` 数据库中的四个域：*名称*、*类型*、*单位*和*自定义*。这些域会在稍后用在文件中。输入段仅加载其中的 *单位*域与参数段中 `IODevice` 变量的值匹配的行。这可通过避免处理无关行来改善性能。

一个名为 `taginfo` 的阵列被创建，其中包含以自定义域的值为基础的键值对。如果自定义域的值是“`VJA=Alarm;VJT=Trend`”，那么会创建包含值 `taginfo[VJA] = "Alarm"` 和 `taginfo[VJT] = "Trend"` 的阵列。

两个字符串变量基于 `taginfo` 阵列的值被创建。

- `<output>`

`<output>` 段定义输出数据库，要针对输入域进行的处理以及要生成的各输出域。XML 文件中可以包含许多输出段。如果需要，输出段可以指定与输入段相同的数据库文件。例如：

```
<output name="digalm" file="digalm.dbf"
  filter="'{variable.alarminfo}=VJA' AND
  '{variable.type}=DIGITAL'">
<field name="tag">{variable.name}_ALARM</field>
<field name="name">{variable.name}_ALARM</field>
<field name="var_a" key="true">{variable.name}</field>
<field name="taggenlink" load="true">{parameter.IODevice}</
  field>
</output>
```

在此范例中，域被写入到 `digalm.dbf` 数据库中。如果输入行与数字量报警相关，只有当前行中会出现输出处理。也就是说，如果输入段中的 `alarminfo` 变量具有的值是“VJA”，输入数据库中的类型域是 `DIGITAL`。标签和名称域会被提供值 `name_ALARM`。

有关模板中使用的专用标签的描述，请参见 [XML 模板标签](#)。有关更全面的示例，请参见 [示例 XML 模板](#)。

引用变量

变量可用于临时存储信息。变量可在 XML 文件的 `<param>` 部分中进行全局定义，也可以在 `<input>` 或 `<output>` 部分进行局部定义。

变量可使用 `<string>` 标签进行定义，可参见 [XML 模板标签](#)。例如：

```
<string name="MyIODevice">DISK_PLC</string>
```

要引用变量，可使用以下语法：

```
{SectionName.VariableName}
```

对于此示例：

```
{parameters.MyIODevice}
```

如果引用的变量是在同一部分中进行定义的，则 *部分名称* 可以省略。

变量在 XML 文件中进行连续评估。首先分配 `<param>` 部分中的变量，然后分配 `<input>` 部分中的变量，最后按顺序分配每个 `<output>` 部分中的变量。也就是说：

- `<param>` 部分中的变量可在同一模板中，由 `<input>` 部分和所有 `<output>` 部分引用。
- `<input>` 部分中的变量可以由所有 `<output>` 部分引用。
- 每个 `<output>` 部分中的变量仅可由后续的 `<output>` 部分引用。

引用阵列

通过成员名称引用阵列成员，可使用：

```
{SectionName.ArrayName[MemberName]}
```

通过匹配模式引用阵列成员，可使用：

```
{SectionName.ArrayName(PatternString)}
```

有关模式匹配通配符的信息，请参见 [模式匹配](#)。

XML 模板标签

用在 TagGen XML 文件中的专用标签在下面列出。

`<template>`

这是在 XML 文件里的根标签。在每一个 XML 文件里的，仅应有一个 `<template>` 标签。

可包含：

```
<param>, <input>, <output>
```

属性：

描述

模板部分的描述。

`<param>`

此可选标签定义用于 XML 文件的全局参数。用户在这能指定可用于文件的其余部分的字符串常量。

可包含：

```
<string>
```

属性：

名称

参数部分的名称。

描述

参数部分的描述。

<input>

这定义 XML 文件的源数据库、将要导入的字段和可能需要的任何输入过滤器。必须仅有一个输入部分被指定。

可包含:

<field>, <string>, <array>, <calculator>

属性:

名称

输入部分的名称。

文件

输入数据库文件名称。

描述

输入部分的描述。

<output>

这定义 XML 文件的输出数据库，将要生成的标签和任何必要的从输入变量数据到生成标签数据的转换。模板至少必须有一个已定义的输出部分。

可包含:

<field>, <string>, <array>, <calculator>

属性:

名称

输出部分的名称

文件

输出数据库文件名称

过滤器

输出部分过滤器，语法:

`filter="variable-FilterString"`

<FilterString> 是一个能够用括号将通配符或通配符字符串与逻辑运算符

(AND、OR 和 NOT) 和分类逻辑项结合的字符串表达式。例如，如果字符串以字母“L”开头，“L*”将匹配，如果字符串以字母“L”或“H”开头，“L* OR H*”将匹配。有关通配符运算符列表，请参见 [模式匹配](#)。

描述

输出部分的描述。

<field>

此标签指定这个数据库字段将在 <input> 和 <output> 部分处理。

当在输入部分使用时将为每一个从输入数据库加载的记录指定命名的字段。文件只读。

当在输出部分使用时将为每一个在输出数据库中产生的标签记录指定写入的命名的区域。

属性:

名称

区分大小写的数据库字段名称。

关键字

仅对输出部分有效，此逻辑标志表明这是否是关键字字段。这对于输出部分的特别处理是有用的。例如，用户可以通过匹配关键字字段使新旧记录设置相同步。

加载

对输入和输出部分都有效，此逻辑标志表明这是否是加载过滤器字段。当从输入和输出数据库装加载数据时，使用加载过滤器字段来过滤记录。

描述

字段元素的描述。

注意：关键字和加载的字符串表达式能包含常规字符串、变量引用、通配运算符或者是它们的混合。有关通配运算符列表，请参见 [模式匹配](#)。

<string>

此标签定义一个可用在 <param>、<input> 和 <output> 部分的字符串常量。

属性:

名称

变量字符串的名称。

描述

字符串元素的描述。

注意：为了统一快速链接，当字符串“IODevice”在 <param> 部分中定义时，指的是工程的 IO 设备。不应被赋予其他的值。

<array>

此标签在 <input> 和 <output> 部分定义一个动态字符串数组。

字符串的设置取决于正在被处理的记录字段和由 ToProperty() 与 Split() 命令产生的字符串。有关详细信息，请参见 [内置函数](#)。

属性:

名称

变量字符串数组的名称。

描述

数组元素的描述。

<calculator>

此标签在 <input> 和 <output> 部分里定义表达式变量。

此标签的内容是用来指定简单数据转换操作的数学表达式。

表达式中的运算对象仅能是数字或有数字值的变量。表达式能使用括号改变计算的先后次序。

例如，在 <input> 部分里的下列计算器通过增加它的当前值来计算所有的输入记录。

```
<calculator name="c1" initial="0">{c1} + 1</calculator>
```

属性：

名称

计算器的名称。

初始值

计算器的初始值，默认为 0。

描述

计算器元素的描述。

模式匹配

可以在字符串中使用下列通配符

字符	描述
*	匹配输入数据流中的任何字符串
?	匹配数据流中的任意单个字符
%d	匹配数据流中的任意十进制整数 (nnn..., 其中 n 为 0-9 中的数字)
%e	匹配数据流中的任意八进制数 (0onnn..., 其中 n 为 0-7 中的数字)
%h	匹配数据流中的任意十六进制数 (0xnnn..., 其中 n 为 0-9、A-F 或 a-f)
%s	匹配数据流中的任意字符串 (与 * 相同)
{	记号字符串的起始标记。输入模式中用 { } 括起的字符 (包括普通字符和特殊字符) 均代表“记

字符	描述
	号字符串”。数据流中匹配此记号字符串的字符即是一个标记，可以由输出数据字符串引用并直接作为组写到输出数据库中。
}	记号字符串的结束标记。
\	标记后面的字符作为文字对待。例如，如果希望在输入数据流中有一个文字字符 *，可以使用 * 来表示。如果需要使用文字反斜线符号 \，可以使用 \\。

内置函数

有三个内置函数可用在 XML 模板中操作字符串。调用内置函数的语法如下：

```
{<FunctionName>(<arg1>, <arg2>, ...)}
```

这三个内置函数为：

SubString(string, start, count)

此函数用于从给定来源提取以字符 *start* 开头且在 *count* 个字符后终止的子字符串。

Split(string, delimiter)

此函数用于将源字符串用分隔符分割为几部分，并将它们储存在基于零的字符串数组中。

例如：

```
<array name="tagid">
  {split('VJA;VJT', ';')}
</array>
```

生成 tagid[0] = VJA, tagid[1] = VJT。

ToProperty(string, separator, delimiter)

此函数将源字符串用分隔符 (*delimiter*) 分割为几部分，然后进一步用分隔符 (*separator*) 分割为键值对。例如：

```
<array name="taginfo">
  {ToProperty('VJA=Alarm;VJT=Trend', '=', ';')}
</array>
```

生成 taginfo[VJA] = Alarm, taginfo[VJT] = Trend。

采样 XML 模板

所提供的采样模板是为了创建模拟量或数字量报警标签和趋势标签。下面显示的模板指定了一些规则，用于针对变量标签的导入和/或同步从工具 SpeedLink 设备/数据库生成 Vijeo Citect 报警和趋势标签。

注意： 为了满足特殊要求，需要修改此模板。有关详细信息，请参阅 [TagGen XML 模板](#)。

对于自定义字段（导入类型为“DIGITAL”的 Vijeo Citect）中包含文本“VJA”的每个工具数据库变量标签，此文件将在 Vijeo Citect (digalm.dbf) 内生成名为 <variablename>_ALARM 的链接数字量报警标签。

对于自定义字段（导入类型不是“DIGITAL”也不是“STRING”的 Vijeo Citect）中包含文本“VJA”的每个工具数据库变量标签，此文件将在 Vijeo Citect (anaalm.dbf) 内生成名为 <variablename>_ALARM 的链接模拟量报警标签。

对于自定义字段（导入 Vijeo Citect）中包含文本“VJA”的每个工具数据库变量标签，此文件将在 Vijeo Citect (trend.dbf) 内生成名为 <variablename>_TREND 的链接趋势标签。此范例文件不针对任何其它趋势标签属性输入参数值。

```
<?xml version="1.0"?>
<template desc="Default SpeedLink Unity Pro TagGen template">

  <param name="parameter">
    <string name="IODevice" desc="SpeedLink required parameter, it will be set to the
      given I/O Device name by the system."></string>
  </param>

  <input name="variable" file="variable.dbf" desc="Load variable tags for the
    specified I/
    O Device">
    <field name="name"></field>
    <field name="type"></field>
    <field name="unit" load="true">{parameter.IIODevice}</field>
    <field name="custom"></field>

    <array name="taginfo">{ToProperty('{custom}', '=', ';')}</array>

    <string name="alarminfo">{variable.taginfo[VJA]}</string>
    <string name="trendinfo">{variable.taginfo[VJT]}</string>
  </input>

  <output name="digalm" file="digalm.dbf" filter="{variable.alarminfo}=VJA' AND
    '{variable.type}=DIGITAL'" desc="Generate digital alarm tags from input digital
    variable
    tags">
    <field name="tag">{variable.name}_ALARM</field>
    <field name="name">{variable.name}_ALARM</field>
    <field name="var_a" key="true">{variable.name}</field>
    <field name="taggenlink" load="true">{parameter.IIODevice}</field>
  </output>

  <output name="anaalm" file="anaalm.dbf" filter="{variable.alarminfo}=VJA' AND
```



```

' {variable.type}=NOT DIGITAL' AND ' {variable.type}=NOT STRING' " desc="Generate
analog
alarms from input analog variable tags">
<field name="tag">{variable.name}_ALARM</field>
<field name="name">{variable.name}_ALARM</field>
<field name="var" key="true">{variable.name}</field>
<field name="taggenlink" load="true">{parameter.IODevice}</field>
</output>

<output name="trend" file="trend.dbf" filter="' {variable.trendinfo}=VJT' "
desc="Generate trend tags from input variable tags">
<field name="name">{variable.name}_TREND</field>
<field name="expr" key="true">{variable.name}</field>
<field name="taggenlink" load="true">{parameter.IODevice}</field>
</output>

</template>

```

简单通配符范例

下例显示空字符串变量（称为“通配符”）如何用作通配符模式从变量数据库装载一组数据。处理 xml 文件时，字符串变量被替换为“*”通配符字符。下例将装载数据，其中变量标签的名称包含单词“Tank”。

```

<input name="variable" file="variable.dbf">
<string name="wildcard"></string>
<!-- Loading set of data where string contains the word "Tank" -->
<field name="name" load="true">{wildcard}Tank{wildcard}</field>
<field name="unit"></field>
<field name="custom"></field>
</input>

```

采样模拟量报警范例

下例指定了一些规则，用于根据链接变量标签的工程刻度生成 Vijeo Citect 模拟量报警标签和设置双高限制属性。

对于自定义字段（导入类型不是“DIGITAL”也不是“STRING”的 Vijeo Citect）中包含文本“VJA”的每个工具数据库变量标签，此文件将在 Vijeo Citect (anaalm.dbf) 内生成名为 <variablename>_ALARM 的链接模拟量报警标签，其中包含以下属性：

- 双高限制，基于变量标签的工程满刻度和工程零刻度以及用户指定的“除数”片断。

在本例中，只对整型变量的模拟量报警设置了限制。

```

<?xml version="1.0" encoding="utf-8"?>
<template>

<param name="parameter">
<string name="IODevice"></string>
<string name="Divisor"></string>

```

```

</param>

<input name="variable" file="variable.dbf">
<field name="name"></field>
<field name="type" ></field>
<field name="unit" load="true">{parameter.IODevice}</field>
<field name="eng_zero"></string>
<field name="eng_full"></string>
<field name="custom"></field>
<!-- customstring field may contain "VJA;VJT;VJC" -->
<array name="taginfo">{ToProperty('{custom}', '=', ';')}</array>
<string name="alarminfo">{variable.taginfo[VJA]}</string>
</input>

<!-- output section -->
<output name="anaalm" file="anaalm.dbf" filter="{variable.alarminfo}=VJA' AND
' {variable.type}=INT' ">
<field name="tag">{variable.name}_ALARM</field>
<field name="name">{variable.name}_ALARM</field>
<field name="var" key="true">{variable.name}</field>
<field name="taggenlink" load="true">{parameter.IODevice}</field>
<calculator name="highh" >({variable.eng_full}-{variable.eng_zero})/
{parameter.Divisor} </calculator>
<field name="highhigh">{highh}</field>
</output>

</template>

```

导出标签

“导出”功能允许您将 I/O 设备数据导出到外部数据源，指定选择的目标和格式（例如 RSLOGIX 驱动程序）。可以随后将这个文件导入到第三方 I/O 设备编程包数据库中。或者可以将其作为备份使用。

注意： 不支持使用 OPC 驱动程序执行导出。

要导出的文件必须已经存在，否则导出操作不能进行。您可以选择在导出前删除内容，或留下它并创建复制标签。

注意： 不是所有的数据库类型都支持导出标签功能。如果有到任何外部数据源的现有链接，链接的标签也将被导出。因为每一个外部数据源的类型结构不同，一些标签数据可能无法被导出。这由 I/O 设备的格式文件决定。

有关联合和 Vijeo Citect 之间的支持级别的信息，请参见 [导出的标签](#)。

注意： 超过 32 个字符的标签名称无法导出到 Unity。

如何导出标签变量到外部数据库：

1. 打开 Citect 工程管理器。
2. 选择工具 | 导出标签。
3. 根据需要完成导出标签变量对话框。

另请参见

[导出标签变量属性](#)

[外部数据源](#)

[格式化文件](#)

导出标签变量属性

使用此对话框 [导出标签](#) 到外部数据库。“导出标签变量”对话框具有下列域：

外部数据库

对标签变量将被导出到的外部数据源的参考（最多 128 个字符）。可以是：

- 直接的路径和文件（例如 C:\Data\Tags.csv）
- IP 地址和节点（例如 127.0.0.1\HMI_Scada）
- URL（例如 <http://www.abicom.com.au/main/scada>）
- 计算机名（例如 \\coms\data\scada）

注意：在导出执行前数据源必须存在。

数据库类型

外部数据源参考的数据格式。

连接字符串

输入连接字符串，提供数据源的连接详细信息。这类似于 ODBC 连接字符串。例如：

```
UserID = XXX; Password = YYY
```

不是所有的数据源都需要连接字符串。

I/O 设备

您正在导出标签的 I/O 设备。使用菜单选择已经使用 Vijeo Citect 定义了的 I/O 设备。

从标签中删除前缀

如果希望从导出标签名的前面删除已知前缀，请选择此框。

标签前缀

要从导出标签名前面删除的前缀（最多 8 个字符）。

删除现有标签

如果要在导出之前删除任何外部数据库中的标签，请选择此框。

注意：对于联合驱动程序，即使您选择了此复选框，现有标签也不会被删除。

外部数据源

当设置导入、导出或链接时，您需要提供数据源和数据格式。

您的数据源可以是：

- 直接的路径和文件（例如 C:\Data\Tags.csv）
- IP 地址和节点（例如 127.0.0.1\HMI_Scada）
- URL（例如 http://www.abicom.com.au/main/scada）
- 计算机名（例如 \\coms\data\scada）

数据库类型域指定了外部数据源的格式。当 Vijeo Citect 试图从此数据源读取时，它将使用数据库类型指定的机制。支持的数据库类型是：

- OPC（OPC1 不支持）
- CSV（逗号分隔值）
- 概念版本 2.1 ASCII 文件

如何配置外部数据源作为一个文件

示例使用了一个 CSV 文件，在这里是一个 RSLOGIX 数据库驱动程序

在“导入/导出或链接”对话框中，输入以下详细信息：

- **外部数据库** C:\Data\Tags.csv
- **数据库类型** RSLOGIX 驱动程序
- **连接字符串**（留为空白）

如何配置使用专门驱动程序的外部数据源

注意：此示例使用提供的 OPC 驱动程序。

在“导入/导出或链接”对话框中，输入以下详细信息：

- **外部数据库：** OPC 服务器进程的名称，例如 FactorySoft.InProc
- **数据库类型：** OPC
- **连接字符串：** 参数是 **ServerNode** 或 **Branch**，但这两个参数都是可选的。它们的使用取决于 OPC 服务器的位置和需求数据的范围。**ServerNode** 可以是 IP 地址或到服务器的网络路径。例如：
 - ServerNode=127.0.0.1
 - ServerNode=\\Server
 - ServerNode=www.server.com

对于 **Branch**，OPC 数据可以是树状（分级）或平面结构。OPC 驱动程序支持访问这两种存储类型。如果数据在树状结构中，它能够通过输入分枝名从根节点向下获得第一个分枝。例如：

- Branch=device1
将来可能支持更深的分枝级别。

另请参见
[格式化文件](#)

格式文件

格式文件定义了导入/导出/链接的规则。此文件将列从外部数据源格式对应到内部 Vijeo Citect 数据库格式。换言之，就是该文件决定导入/导出/链接哪些信息，以及在操作过程中如何修改这些信息。

格式文件还提供关于让 Vijeo Citect 使用正确的驱动程序访问外部数据源的信息。

有些格式文件随附 Vijeo Citect 提供；但有时，您可能需要使用 Microsoft Notepad 等编辑器来写入或修改格式文件。

下面是一个格式文件如何规范工作的范例。（有关详细信息，请参见 [格式文件布局](#)。）

1. 外部数据源中的列 3 需要复制到 Vijeo Citect 的标签数据库的“Name”列中。（Vijeo Citect 名称只能使用字母数字字符（a - z、A - Z，以及 0 - 9）和下划线“_”，但在格式文件中通常不用考虑这些；Vijeo Citect 会自动执行转换）。然而，如果外部数据源的列 3 碰巧为空，就根本不需要复制此记录（一定会被拒绝）。
2. 外部数据源的列 1 需要直接复制到 Vijeo Citect 的标签数据库的“Addr”列中，因为它们二者代表意义相同。
3. 外部数据源的列 4、5、6 和 7 都需要复制到 Variable.DBF 中的“Comment”列（外部数据源在几个域之间划分注释也比较常见）。这些域需要按照原有顺序复制，因此如果外部数据源列 4 中的数据为“Loop”，列 5 中是“1”，列 6 中是“Process”，列 7 中是“variable”，这些域按原有顺序进行复制，则 Variable.DBF 的“Comment”列将为“Loop 1 Process variable”。此过程称为“串联”。（对于“Comment”列，Vijeo Citect 会自动在来自外部数据源的每个域之间添加一个空格。）
4. 外部数据源列 1 中的数据决定了 Vijeo Citect 需要在“Type”列中写入什么。然后，数据不能直接复制，因为对 Vijeo Citect 没有意义。而是需要一个转换（或过滤）过程。转换需要自己的一套规则，例如：
5. 如果外部数据源的列 1 为“BT%d:%d.%d”（其中 %d 代表“任意十进制数”），Vijeo Citect 需要在“Type”列中写入字符串“DIGITAL”。
6. 如果外部数据源的列 1 为“F%d:%d/%d”（其中 %d 代表“任意十进制数”），Vijeo Citect 需要在“Type”列中写入字符串“DIGITAL”。
7. 如果外部数据源的列 1 为“O:%e”（其中 %e 代表“任意八进制数”，即所有位都为 0 - 7），Vijeo Citect 必须保留“Type”为空。它仍将接受记录（倘若所有其它列都通过了过滤测试），但不在“Type”列中写入内容。假设为 Vijeo Citect 目前没有（或不需）适合的对应类型。
8. 如果外部数据源的列 1 为“PD%d:%d.%d”，Vijeo Citect 需要在“Type”列中写入字符串“REAL”。
9. 如果外部数据源的列 1 为“ST%d:%d”，Vijeo Citect 需要在“Type”列中写入字符串“STRING”。

10. 如果没有规则限制外部数据源列 1 中的内容，Vijeo Citect 则必须拒绝整条记录，而不将其复制到 Vijeo Citect 数据库。

另请参见
[格式文件布局](#)

格式文件布局

格式文件分为不同的段：

- [\[General\] 段](#)
- [\[Columns\] 段](#)
- [\[ImportFilterMap\] 和 \[ExportFilterMap\] 段](#)

每个段在单独的一行中包含段标题（用方括号括起的段名称（例如“[my_section]”））。后面是段的主体，通常是框体的单行声明：

```
"something = something_else -> something_else_again"
```

“=”和“->”两侧有无空格均可，但整个声明必须在一行。格式文件中的大多数声明都遵循这个模式，但在很多情况下，可以没有“->”（转换器），或只有一个转换器后面不跟任何内容。

下面的段是所有格式文件都必须有的：

```
[General]  
[Columns]  
[ImportFilterMap]  
[ExportFilterMap]
```

其它段是否需要，需要视 Vijeo Citect 与外部数据源之间转换的复杂性而定。具体由 [ImportFilterMap] 和 [ExportFilterMap] 的内容来决定。

注释也可以加在到段中或段之间。要这样加注释，请使用分号“;”作为注释行的第一个字符。则该行的剩余内容将被视为注释，将被 Vijeo Citect 忽略。例如：

```
; I am putting the [General] section here  
[General]
```

[General] 段

General 段包含 4 行：

```
[General]
Name=nameDescription= descriptionDriverName= driver nameDriverInst="a special
string"
```

*名称*和*描述*当前未被 Vijeo Citect 使用。Vijeo Citect 使用 *驱动程序名称* 加载用于访问外部数据源的正确驱动程序。此驱动程序可能是 Vijeo Citect 安装程序的一部分或是自定义驱动程序（包括一个您自行写入的驱动程序），用于访问特定数据源（可以是协议、硬件类型、服务器或文件类型）。驱动程序必须是兼容 OLE DB 的驱动程序。

可以通过 *特殊字符串* 将额外的信息传递到驱动程序。此信息将被添加到连接字符串（在 Citect 浏览器和工程编辑器）。因此，连接字符串可用于可能经常改变的信息，此 *特殊字符串* 可用于相对固定的信息（如 .CSV 文件的逗号 “,” 分隔符）。该字符串的主要用途是作为输入文件的分隔符。要指定逗号 “,” 已被输入文件用作分隔符，应使用以下语法：

```
DriverInst="delimiter=,"
```

[Columns] 段

Columns 段定义了外部数据源的列格式。它的结构如下：

```
[Columns]
External column name 1 = column width -> data type
External column name 2 = column width -> data type..
External column name n = column width -> data type
```

Vijeo Citect 为 *External Column name n* 放置数据唯一的限制是它在段内必须是唯一的。为了方便，你可以使用外部数据源的名字（诸如 "Description"、"PLC_id"、"Iotype"）或只是叫 "Column1"、"Column2" 等。这些条目出现的顺序必须与这些字段在外部数据源中的顺序相同。[ImportFilterMap] 和 [ExportFilterMap] 段中的外部数据源列使用的名称也必须符合这个顺序。

列宽 是字段中的字符数，*数据类型* 是列的数据类型。当前可接受的数据类型是 "STRING"。

[ImportFilterMap] 和 [ExportFilterMap] 段

[ImportFilterMap] 和 [ExportFilterMap] 段具有相同的语法和功能，除了 [ImportFilterMap] 描述在导入时如何从外部数据源转换数据，而 [ExportFilterMap] 描述相反的转变。

在格式文件中，它们是最复杂的段。（剩下的文本将集中在 [ImportFilterMap]，因为 [ExportFilterMap] 遵循相同的逻辑。）

[ImportFilterMap] 的结构如下：

```
[ImportFilterMap]
  Import Rule 1 = External column name 1 -> Citect Column i
  Import Rule 2 = External column name m -> Citect Column j..
  Import Rule nn = External column name n -> Citect Column k
```

Import Rule nn 中的值可以是任何字符串，但在段内必须是唯一的。因此，为了方便，您可能希望使用 "ImportRule1"、"ImportRule2"、"Mapping1"、"Filter1" 等，或与转换有关的任何描述，诸如 "Description_to_comment"。

用于 *External column name n* 的名字必须是与 [Columns] 段中的 *External column name n* 相同的名字。

用于 *Citect Column k* 的名称必须是与 Vijeo Citect 内部标签数据库中的其中一列相同的名字，如 "Name"、"Type"、"Addr"、"Comment" 等。

因此，外部列和 Vijeo Citect 列的值提供了有关在导入期间如何将数据从外部列转换到 Citect 列的信息。

例如：

```
ImportRule1 = Description -> Comment
```

这显示了外部数据源的 "Description" 字段中的数据与需要进入 Vijeo Citect 数据库的 "Comment" 字段中的数据之间的关系。

您用于 *Import Rule nn* 的名称可能与格式文件中另一可选段的名称相同：此时，该附加段为 Vijeo Citect 提供更多信息。在最简单的情况下，如果在格式文件中没有这个名称的段，规则只是简单地声明 *External column name n* 中的数据将不做修改或过滤而直接复制到 *Citect Column k*。

因此，如果外部数据源中的 "Description" 列包含 "Truck Position 1"，上面的输入显示在 [ImportFilterMap] 段，而且不存在名称为 [ImportRule1] 的段，则在导入之后，Vijeo Citect 数据库中的 "Comment" 列将包含字符串 "Truck Position 1"。

串联

要将多个域串联到 Vijeo Citect 数据库中的一个域，请向 [ImportFilterMap] 部分添加单独的条目。每个部分必须包含相关外部列的名称和 Vijeo Citect 中目标列的名称。条目必须以域要串联的次序显示。

因此，如果外部数据源包含一个名为 "IOdev" 包含值 "M" 的域，另一个是名为 "IOaddr" 包含值 "61" 的域，您要将它们连起来以便将值 "M61" 导入到 Vijeo Citect "Addr" 域中，则用下列方式实现：

```
[ImportFilterMap]
Addr1= IOdev -> Addr
```



```
Addr2= IOaddr -> Addr
```

此处，必须要确认没有名为 [Addr1] 或 [Addr2] 的格式文件，除非您需要某种过滤或转换。

另请参见
[域转换](#)

域转换

要修改映射数据或应用过滤（拒绝特定的记录），必须从映射段中使用相关行的名称创建新的段。例如，如果有下面的映射段：

```
[ImportFilterMap]
Test1_to_type = Test1 -> Type
```

而且要在将 Test1 列导入之前转换其中的数据，在该文件的某个位置必须有一个名为 [Test1_to_type] 的段：

```
Test1_to_type]
```

后面接着需要的转换规则。

注意：名称必须来源于导入映射段，而不是导出映射段。如果使用了导出映射段的名称，则转换将适用于导出而不是导入。

此转换/过滤段的基本格式如下：

```
[Relevant mapping name]
Filtering Rule 1 = External Pattern 1 -> Citect String 1
..
Filtering Rule m = External Pattern m
..
Filtering Rule n = External Pattern n -> Citect String n
```

用于 *过滤规则* n 的名称对 Vijeo Citect 没有本质意义（除非它将该名称用作查找条目的关键词）。唯一的限制是必须在段内唯一，所以您可以使用任意方便的名字。

外部模式 n 中的值是组合字符，Vijeo Citect 会在外部数据源列中查找它。结合方式如下：

格式文件中的字符	外部数据源中匹配的字符串
<特定的文本>	<特定的文本>
*	任意字符串。
?	任意单个字符。
%d	任意十进制整数 (nnn... 其中 n 为 0-9)。
%e	任意八进制数字 (0onnn... 其中 n 为 0-7)。
%h	任意十六进制数字 (0xn... 其中 n 为 0-9、A-F 或 a-f)。
%s	任意字符串。
{	“记号字符串”的起始标记。输入模式中用 { } 括起的任何字符 (包括普通和特殊字符) 均代表记号字符串。在数据流中匹配记号字符串的字符被输出数据字符串使用并且直接作为组写到输出数据库中。
}	记号字符串结束标记。
\	标记后面的字符作为文字对待。例如, 如果在输入数据流中希望有一个文字字符 *, 你可以使用 * 来表示。如果需要使用文字反斜线符号 \, 可以使用 \\。

任何其它的字符必须逐字匹配同样的字符。

如果外部列中有 External Pattern n, Citect String n 会写入到 Vijeo Citect 的相关列中 (按照对应关系)。

另外, 在输出数据字符串中可以显示两个特殊的字符:

输出字符串中的字符	意义
\$	模式 \$n (其中 n 是任意整数) 在输出数据流中由第 n 个“记号”替换; 记号是输入模式中由 { } 括起的匹配序列。(如果 \$ 没有接一个记号数字, 将产生提醒信息。)
!REJECT!	序列必须在输出数据模式中显示。整个记录会被拒绝。因为记录已经匹配上输入模式, 所以不再检查其它规则。
\	标记后面的字符作为文字对待。如果需要一个文字 \$ (使用 \\$), 或有其它位直接接着, 将使用此项。例如, 如果字符串“3August2001”直接连接记号, 则使用“\$1\3August2001”。
\ (在行结尾)	在输出行的结尾插入一个空格“ ”。如果没有这个规定, 系统将无法区分输入行的结尾 (可能跟着字符, 比如空格, Windows 将忽略) 和在输出行结尾需要的空格。

其它字符将逐字写入到输出数据库中。

会逐一查看此段中的每个过滤条件，找一个匹配项。如果规则不匹配，就逐一尝试，直到找到匹配项。如果找不到匹配项，将拒绝整个记录；进而任何域中的数据都不会复制到 Vijeo Citect 中。

例如，要将外部数据源中的字符串“FLOAT”转换为 Vijeo Citect 中的“DIGITAL”，可以使用以下条目：

```
[ImportFilterMap]
Test1_to_type = Test1 -> Type
..
[Test1_to_type]
Rule1 = FLOAT -> DIGITAL
..
```

下面是一个较复杂的范例，我们假定外部数据源有一个名为“Tag”的列，它相当于 Vijeo Citect 中的“Name”域。此外，还假定外部数据库没有 Vijeo Citect 的“Type”域的直接对应项，而 Vijeo Citect 需要填入此域。就需要使用“Tag”域来决定哪些内容进入到 Vijeo Citect 数据库的“Type”域。

如果外部数据源中的“Tag”列有值“1:060/07”，我们决定应该将字符串“DIGITAL”写入到 Vijeo Citect 的“Type”域。实际上，如果该域的“1:”后面连接了八进制值，后跟“/”，后跟八进制值，我们希望字符串显示在我们的“Type”域中。如何在格式文件中表达这些呢？

首先，有两组关系需要考虑，一个是将外部数据源中的“Tag”域连接到 Vijeo Citect 中的“Name”域；另一个是将其连接到 Vijeo Citect 中的“Type”域。所以我们需要在 [ImportFilterMap] 段有两个映射（条目）：

```
[ImportFilterMap]
Tag_to_Name = Tag -> Name
Tag_to_Type = Tag -> Type
..
```

因为我们希望将“Tag”域中的数据直接拷贝到“Name”域中，我们通过在格式文件中不包括 [Tag_to_Name] 段来实现。

但是因为我们不是从“Tag”域直接复制到“Type”域，而只是使用数据来决定哪些内容进入“Type”，所以就需要 [Tag_to_Type] 段。

重新调用需要的输出：如果“Tag”域的“1:”后面连接了八进制值，后跟“/”，后跟八进制值，我们希望字符串显示在我们的“Type”域中。

在格式文件中表达如下：

```
[Tag_to_Type]
Rule1 = I:%e/%e -> DIGITAL
..
```

这样将匹配“I:060/07”或“I:0453/02343445602”（而且还会将字符串“DIGITAL”写入到 Vijeo Citect 的 Type 域），但不会匹配“I:060/98”或“I:054”。

下面是几个如何使用通配符的范例，格式文件中的模式“HE%sLD”或“HE*LD”应该匹配外部数据源中的“HELLO WORLD”或“HE IS VERY BOLD”。模式“HE??????LD”应该匹配“HELLO WORLD”而不是“HE IS BALD”，因为每一个问号“?”必须匹配一个字符。

Vijeo Citect 还将处理多通配符模式，比如“%s/%s:%s”。

下面是一个比“Hello World”更有帮助的范例，假设我们直接复制数据而不修改，但我们要确认不复制空域。模式“?%s”或“?*”将匹配任意至少有一个字符的字符串，但不匹配空白。

有时，输出只需要输入流中的一部分，或输入需要划分成不同的输出列。这种情况下，“记号”十分有用。

在这个导出情况的范例中，Vijeo Citect 数据库中的“Addr”域需要在外部数据库的两个域中划分：“IOdev”（它的值总是“D”或“M”）；和“IOaddr”（它的值是一个不超过 3 位数的十进制整数）。Vijeo Citect 数据库“Addr”域中的值是“D62”、“M546”等字符串。

这个情况可以通过串联来解决，也就是说，使用一个映射去写 Iodev 域，然后其它三个映射将每一位单独复制到外部数据库的 IOaddr 域。但这可能是很复杂的并且在一些情况下无法实现。

最好就是使用记号去解决这个情况：

```
[ExportFilterMap]
..
Addr2IOdev = Addr -> IOdev
Addr2IOaddr = Addr -> IOaddr
..
[Addr2IOdev]
D = D* -> D
M = M* -> M
AnythingElse = * ->
..
[Addr2IOaddr]
Rule = ?{%d} -> $1
```

在 [Addr2IOaddr] 段中，{%d} 是一个记号字符串，因为它是规则中使用的第一个（并且是唯一的）记号，使用 \$1 在输出流中引用它。因此，如果 Vijeo Citect 数据库的“Addr”域包含“D483”，“D”会写入到外部数据池的“IOdev”域中，“483”（记号）写入到“IOaddr”域中。

这里是一个如何使用多记号的范例。假设我们需要：将所有句点字符 (.) 转换成冒号 (:); 去掉前两个字符 (空字符); 然后从希望的数据中去掉所有不需要的字符; 也就是, 将 “..BJ6452.78.....” 转换成 “BJ6542:78”。这可以通过使用下面的规则来实现:

```
Rule = ??{*%d}.{%d}* -> $1:$2
```

这时, 我们可以引入格式文件的另一个功能。如果您使用下面的规则:

```
[Relevant mapping name]
Filtering Rule = External pattern
```

例如, 如果不包含 “-> Citect 字符串”, Vijeo Citect 将此解释为 “检查该字符串是否匹配外部模式, 如果是, 则保持不变进行复制”。

如果使用这个规则:

```
Filtering Rule = External pattern ->
```

即, 如果没有 “Citect String”, 将代表: “如果字符串模式匹配, 则接受记录, 但复制一个 NULL 字符串到 Vijeo Citect 数据库中。”

再来看上面的范例, 我们增加一个限制, 在外部数据源的标签域没有数据的任何记录 (也就是说, 空白或 NULL 字符串) 不应导入到 Vijeo Citect 中。我们要添加 [Tag_to_Name] 段, 且将只有一个规则: 我们接受除空白之外的一切内容。

```
[Tag_to_Name]
RejectBlanks = ?*
..
```

重新调用 Vijeo Citect, 按照过滤规则依次检查每个模式, 直到在外部数据源中找到匹配字符串的模式。请记住, 通过安排正确的规则, 在一些情况下, 通过使用串联, 可以进行大范围的转换和过滤。

例如, 如果某些字符串类型需要转换, 但所有其它字符串类型需要原样复制到 Vijeo Citect, 则可以让具有一组规则的段在顶部, 后跟最终规则, 以便其它所有字符串类型保持原样。

```
[Tag_to_Name]
Rule n = ..... -> .....
..
```

```
LetEverythingElseThru = %s
```

一个不含任何其它字符的 %s 或 *, 可与所有字符串匹配, 包括空白。

下面是一个如何拒绝特殊字符串或模式的范例, 我们假定需要拒绝所有以“DFILE”开始的标签 (另一个实际范例)。我们可以使用以下条目:

```
[Tag_to_Name]
Rule1 = DFILE* -> !REJECT!
..
LetEverythingElseThru = %s
```

很明显, 因为与输入模式不匹配的任何标签都将被拒绝, 所以有 !REJECT! 规则而后面没有关于您希望接受的模式的其它规则是没有任何意义的。当使用 !REJECT! 规则时, 决定显示规则后的逻辑将变得尤为重要。通常, 您可以在映射中把拒绝规则设为第一个规则。在映射中将 !REJECT! 规则作为最后一个规则没有任何意义。

在一些由其它系统产生的文本文件中, 如果不需要开头的几行, 但需要剩下的数据, 这时 !REJECT! 规则是很有用的。

另请参见

[让Vijeo Citect识别格式文件](#)

识别格式文件

需将格式文件保存到安装期间选定的 Vijeo Citect 用户和数据文件夹中的 config 文件夹内。

如果 Vijeo Citect 要执行导入、导出或链接操作, 它需要知道格式文件的名称和将访问外部数据源的驱动程序名称。还需要将显示在输入或输出对话框的**数据库类型**域中的字符串文本。所有这些信息都存储在相同目录下名为 tagdriv.ini 的另一个文件中。

tagdriv.ini 的格式很简单, 并以 odbc.ini 格式为基础。它在安装时就已包含格式文件和驱动程序 (随 Vijeo Citect 附带) 所需要的信息。您只需要为您所使用的驱动程序和新格式文件复制相同的布局即可。

The tagdriv.ini 文件包含数个段。第一个段是具有以下通用布局的 [External data sources] 段:

```
[External data sources]
Section name 1 = the name you want to appear in the import/export/ link menu for
entry 1
Section name 2 = the name you want to appear in the import/export/ link menu for
entry 2..
Section name nn = the name you want to appear in the import/export menu for entry
nn
```

该段中的每一条目引用执行导入、导出或链接操作所需的格式文件和驱动程序的组合。

“=”号左边的文本必须引用显示在 tagdriv.ini 中的其它段的名称。

对于输入、输出或刷新变量标签，“=”号右边的文本与显示在“**数据库类型**”下面菜单中的内容完全相同。

其它段引用 [External data sources] 段中描述的输入或输出类型，并提供有关格式文件和驱动程序对的详细信息。这些段的通用布局如下：

```
[Section name matching an entry in [External data sources] ]
driverid = Driver ID
datastring = The name of the format file
Currently the Driver ID is always CiTrans.
```

因此，如果我们假设您安装的 Vijeo Citect 版本包含 5 个格式文件，那么 tagdriv.ini 中将有 5 个段，如下面的范例所示：

```
; This file contains the driver name, driver prog id, and format file mappings
; The format file must reside in the BIN directory

[External data sources]
CSV = CSV Driver
Concept ver 2.1 Ascii = Concept Ver 2.1 ASCII file
OPC = OPC
UnityProDynamic = Unity SpeedLink Dynamic
UnityProStatic = Unity SpeedLink Static

[CSV]
driverid = CiTrans
datastring = csv.fmt

[Concept ver 2.1 Ascii]
driverid = CiTrans
datastring = concept ver 2_1.fmt

[OPC]
driverid = CiTrans
datastring = OPC.fmt

[UnityProDynamic]
driverid = CiTrans
datastring = UnityProDynamic.fmt
LiveUpdate = TRUE

[UnityProStatic]
driverid = CiTrans
datastring = UnityProStatic.fmt
LiveUpdate = FALSE
```

这会将 5 个条目添加到数据库类型下拉菜单中：CSV、ASCII、OPC、Unity Pro Dynamic 和 Unity Pro Static。

菜单中的 5 个条目与 [External data sources] 段中“=”号右边的字符串相匹配。

如果您添加其它格式文件，还需要向 tagdriv.ini 中添加匹配条目。例如，如果您为 Simatic 数据源添加新格式文件，则必须将一个与之类似的行添加到 [External data sources] 段：

```
SIMATIC = Siemens SIMATIC Driver
```

还必须向文件的底部添加下列段：

```
[SIMATIC]
driverid = CiTrans
datastring = SIMATIC.fmt
```

保存文件，重新启动 Citect 浏览器，“Siemens SIMATIC 驱动程序”将显示在菜单中。

如果选择此条目，[SIMATIC] 段下数据字符串条目中的格式文件将被用于输入、输出或链接；即 simatic.fmt。

联合支持矩阵

以下部分显示联合和 Vijeo Citect 间支持的级别：

- [导入的标签](#)
- [导出的标签](#)
- [链接的标签](#)

导入的标签

对于 Unity SpeedLink 动态和静态：

联合 PLC	联合数据类型	Citect 协议	支持状态
QUANTUM	BOOL	Unite	不支持
	EBOOL	Modnet	支持（除 BYTE 类型）
	BYTE		
	WORD	Modbus	支持（除 BYTE 类型）
	DWORD		
INT			
DINT	MBPlus	支持（除 BYTE 类型）	
UINT			
UDINT			
REAL			
TIME *			
DATE *	OPC	支持（除 DT 类	
TOD *			

联合 PLC	联合数据类型	Citect 协议	支持状态	
	DT * STRING		型)	
PREMIUM	BOOL EBOOL BYTE WORD DWORD INT DINT UINT UDINT REAL TIME * DATE * TOD * DT * STRING	Unite	支持	
		Modnet	支持 (除 BYTE 类型)	
		Modbus	支持 (除 BYTE 类型)	
		MBPlus	支持 (除 BYTE 类型)	
		OPC	支持 (除 DT 类型)	
		M340	Unite	不支持
		Modnet	支持 (除 BYTE 类型)	
		Modbus	支持 (除 BYTE 类型)	
		MBPlus	不支持	
		OPC	支持 (除 DT 类型)	

注意：将标签从 Unity Pro 导入到 Citect 工程时，Citect 建议使用 Unity SpeedLink 动态驱动程序。

注意：除了 DATE、TOD、DT 和 STRING 数据类型之外的所有其他数据类型都支持阵列。Unity 数据类型标有 * 号，Citect 没有与 Unity 类型直接匹配的数据类型。使用 Citect 支持的数据类型和 Cicode 转换和模拟这些数据类型。

此外，也不支持下列联合数据类型：

- 结构数据类型。
- 不是基于零的阵列。
- 多维阵列。
- 所有协议都不支持 Unity BOOL 数据类型的阵列，除了 OPC 和将从导入中排除的类型。

导出的标签

对于 SpeedLink 静态:

联合 PLC	联合数据类型	Citect 协议	支持状态		
QUANTUM	BOOL EBOOL BYTE WORD DWORD INT DINT UINT UDINT REAL TIME * DATE * TOD * DT * STRING	Unite	不支持		
		Modnet	支持 (除 BYTE 类型)		
		Modbus	支持 (除 BYTE 类型)		
		MBPlus	支持 (除 BYTE 类型)		
		OPC	不支持		
		PREMIUM	BOOL EBOOL BYTE WORD DWORD INT DINT UINT UDINT REAL TIME * DATE * TOD * DT * STRING	Unite	支持
				Modnet	支持 (除 BYTE 类型)
				Modbus	支持 (除 BYTE 类型)
				MBPlus	支持 (除 BYTE 类型)
				OPC	不支持
M340	BOOL EBOOL BYTE WORD DWORD INT DINT UINT UDINT REAL TIME * DATE * TOD * DT * STRING			Unite	不支持
				Modnet	支持 (除 BYTE 类型)
				Modbus	支持 (除 BYTE 类型)
				MBPlus	不支持
				OPC	不支持

注意：除了 DATE、TOD、DT 和 STRING 数据类型之外的所有其他数据类型都支持阵列。Unity 数据类型标有 * 号，Citect 没有与 Unity 类型直接匹配的数据类型。使用 Citect 支持的数据类型和 Cicode 转换和模拟这些数据类型。

此外，也不支持下列联合数据类型：

- 结构数据类型。
- 不是基于零的阵列。
- 多维阵列。
- 除了 OPC，所有协议都不支持 Unity BOOL 数据类型的阵列。

链接的标签

链接的 I/O 设备实时更新

对于 Unity SpeedLink 动态：

联合 PLC	联合数据类型	Citect 协议	支持状态
QUANTUM	BOOL	Unite	不支持
	EBOOL		
	BYTE	Modnet	支持（除 BYTE 类型）
	WORD		
	DWORD		
	INT	Modbus	支持（除 BYTE 类型）
	DINT		
	UINT		
	UDINT	MBPlus	支持（除 BYTE 类型）
	REAL		
TIME *			
DATE *			
TOD *	OPC	支持（除 DT 类型）	
DT *			
STRING			
PREMIUM	BOOL	Unite	支持
	EBOOL		
	BYTE	Modnet	支持（除 BYTE 类型）
	WORD		
	DWORD		
	INT	Modbus	支持（除 BYTE 类型）
	DINT		
	UINT		
	UDINT	MBPlus	支持（除 BYTE 类型）
	REAL		
TIME *			
DATE *			
TOD *	OPC	支持（除 DT 类型）	
DT *			
STRING			

联合 PLC	联合数据类型	Citect 协议	支持状态
M340	BOOL	Unite	不支持
	EBOOL		
	BYTE	Modnet	支持（除 BYTE 类型）
	WORD		
	DWORD		
	INT	Modbus	支持（除 BYTE 类型）
	DINT		
	UINT		
	UDINT		
	REAL	MBPlus	不支持
	TIME *		
	DATE *	OPC	支持（除 DT 类型）
	TOD *		
	DT *		
STRING			

注意：除了 DATE、TOD、DT 和 STRING 数据类型之外的所有其他数据类型都支持数组。Unity 数据类型标有 * 号，Citect 没有与 Unity 类型直接匹配的数据类型。使用 Citect 支持的数据类型和 Cicode 转换和模拟这些数据类型。

此外，也不支持下列联合数据类型：

- 结构数据类型。
- 不是基于零的阵列。
- 多维数组。
- 所有协议都不支持 Unity BOOL 数据类型的阵列，除了 OPC。

Chapter: 18 定义和绘制图形画面

Runtime 系统通常由一系列显示在计算机屏幕上的图形画面组成，并提供“透视图的窗口”。您可以设计画面以向操作员提供对工厂某个区域（或整个工厂）的控制。通过使用各种被称为对象（objects）的图形项，您的图形画面还可以显示工厂的状态。

另请参见

[创建新的图形画面](#)

[使用画面模板](#)

[使用浏览顺序](#)

[指定启动画面](#)

[调整画面大小](#)

[定义画面属性](#)

[设置缺省画面设置](#)

[了解绘图环境](#)

[在工程中使用查找和替换功能](#)

创建新图形画面

可以在监视器上逐个显示图形画面，或者一次显示几个画面。可以按由操作员命令控制或自动控制的任何顺序来显示这些画面。

创建新画面：

1. 在“图形编辑器”上，点击**新建**，或者选择**文件 | 新建**。
2. 点击**画面**。
3. 选择创建画面所依据的**模板**。
4. 选择画面的**风格**。
5. 根据需要选中或清除**链接**和**标题栏**选项。
6. 选择画面的**屏幕分辨率**。
7. 点击**确定**。

注意：如果使用图形编辑器创建新画面，必须对画面记录进行编辑（使用工程编辑器）以定义一个浏览次序。

另请参见

[新建对话框](#)

新建对话框

使用此对话框可通过选择某个按钮，创建画面、模板、符号、精灵或超级精灵。

使用页面

要打开一个已经存在的页面：

1. 单击**打开**，或选择**文件|打开**。
2. 选择**类型：页面**。
3. 选择存放页面的**工程**。
4. 选择该**页面**。
5. 单击**确定**。

注意：要从工程中删除页面，请选定该页面的名称，然后点击**删除**。

保存当前页面：

1. 单击**保存**，或选择**文件|保存**。
2. 选中该页面所在的**工程**。（该页面名称的前八个字符必须同其它页面名称相区别。
3. 单击**确定**。

将当前页面用新的名称保存：

1. 选择**文件|另存为**。
2. 选中该页面所在的**工程**。
3. 在**页面**选项中键入该页面的名称（最多 64 个字符）。该页面名称的前八个字符必须同其它页面名称相区别。
4. 单击**确定**。

保存所有当前已经打开的页面

- 选择**文件|全部保存**。（每个页面名称的前八个字符必须是互不相同的。）

另请参见

[使用模板（新页面/模板）对话框](#)
[打开/另存为对话框](#)

要确定某个图形页面的位置：

1. 选择**文件|查找**。
2. 从**工程名称**列表中选中某个工程。
3. 对**页面**列表中的页面进行浏览。
4. 点击**确定**来查看所选的页面。

另请参见

[在工程中使用查找和替换功能](#)

用打印机打印某个图形页面：

- 选择**文件|打印**。（该操作中将不会出现确认对话框。）

要关闭一个已经存在的页面：

- 选择**文件|关闭**。

用户模板（新画面/模板）对话框

利用该对话框，用户可以在某个已经存在的模板基础上创建一个新的画面或模板。

模板

用户可以在一个列表中来选择一个基本的模板并在此基础上建立新的画面或者模板。首先利用滚动条来确定模板缩略图的位置，再选择相应的模板并点击确定**确定**（或者双击该模板的缩略图）。

注意： 要编辑模板，选择该模板并点击**编辑**。

风格

画面的风格。Vijeo Citect 模板分为几种风格，并且提供了多种画面分辨率。创建新工程时，用户可以选择最适合自己以及实际应用的风格。要了解关于每个类型的详细信息，请参考名为“展现 Vijeo Citect”的小册子。该小册子是附带在 Vijeo Citect 系统中的。

链接

选中该方框以保持同原始模板的链接。如果原始模板发生改变，则与该模板链接的画面或模板会自动被更新。

注意： 用户可以在任何时刻通过从“编辑”菜单中选择**切断链接**命令来切断同模板的链接，但是当某个链接被切断后，用户将无法重新建立某个页画面或者模板同其原始模板之间的链接。

标题栏

标题栏中将显示画面或者模板的标题。

分辨率

画面或者模板的屏幕分辨率：

显示器类型	宽度（像素）	高度（像素）
缺省值	计算机当前使用的屏幕宽度	计算机当前使用的屏幕高度
VGA	640	480
SVGA	800	600
XGA	1024	768
SXGA	1280	1024

显示器类型	宽度 (像素)	高度 (像素)
用户	用户定义的大小	用户定义的大小

打开/另存为对话框

可以通过此对话框打开或保存画面、模板、符号、精灵或超级精灵。（要选择实体的类型，请点击相应的标签。）

画面/符号/模板/精灵

图形画面、模板、符号、精灵或超级精灵的名称。

打开画面、模板、符号、精灵或超级精灵时，从较大的窗口中选择其名称。

保存画面、模板、符号、精灵或超级精灵时，向较小的输入框中键入名称（如果希望覆盖现有画面、模板、符号，精灵或超级精灵，则可以从较大的窗口中选择名称）。

注意：如果您使用的是分布式服务器，那么此名称对于集群必须是唯一的（例如，不能在多个集群中使用同一个画面名称）。

项目

要将图形画面、模板、符号、精灵、或超级精灵保存其中的工程。

库

（仅限符号、精灵和超级精灵。）保存符号、精灵或超级精灵的库。要创建新库，点击**新建**。

风格

（仅限模板。）模板的风格。要创建新风格，点击**新建**。

启用预览

显示画面、模板、符号、精灵或超级精灵的缩略图。

标题栏

（仅限模板。）指定是否为标题栏提供空间。如果您使用标题栏，那么将会略微减小屏幕上显示的空间。

分辨率

（仅限模板。）模板的屏幕分辨率：

屏幕类型	宽度 (以像素计)	高度 (以像素计)
------	-----------	-----------

屏幕类型	宽度（以像素计）	高度（以像素计）
缺省值	计算机当前使用的屏幕宽度	计算机当前使用的屏幕高度
VGA	640	480
SVGA	800	600
XGA	1024	768
SXGA	1280	1024
User	用户定义的大小	用户定义的大小

要删除画面、模板、符号、精灵或超级精灵，选中相应的对象并点击**删除**。

使用页面模板

可以在运行系统中使用多种不同的页面类型。

- Mimic 页面显示工厂的状态，操作员可使用其中的按钮和其它控件控制工厂内的过程。
- 报警页面显示报警信息。
- 趋势页面显示工厂过去和现在的活动的直观表示。

注意：必须为报警创建缺省页面（包括报警汇总和硬件报警页面）。

要使用户能够快速创建图形页面，系统为所有常用页面类型提供了模板。模板有助于创建具有一致“外观”的工程页面。（在工程中保持一致会缩短操作员了解如何使用系统所需的时间）。

另请参见

[选择页面风格](#)

[链接模板](#)

[创建自己的模板](#)

选择画面风格

模板分为几种风格，并且提供了多种画面分辨率。创建新工程时，用户可以选择最适合自己口味和实际应用的风格。

另请参见

[链接模板](#)

链接模板

使用模板创建新画面时，可以保持该模板的链接。如果原始模板发生改变，则与该模板链接的画面会自动被更新。

当画面与模板链接时，无法通过常用的双击方法从该画面访问组成该模板的对象。要显示这些对象的属性，请按住**控制 (CTRL)** 键的同时双击要查看其属性的对象。或者选择**工具 | 定位对象**，选定该组，然后点击**确定**。不过，这些属性中的多数都是只读属性。

注意：可以使用**编辑 | 删除关联**随时切断与模板的链接，但在切断链接后无法重新将画面或模板与其原始模板链接。

另请参见
[创建自己的模板](#)

创建自己的模板

如果工程包含几个类似的画面（例如，菜单画面、通用过程或通用设备），用户可以创建自己的模板（包含所有通用对象）作为这些画面的基础。然后可以基于该模创建画面，并在每个画面上添加单个对象。

如果要删除或更改通用对象的位置，或者要添加新的通用对象，用户无须更改每个画面，只需更改模板即可。所有基于该模板的画面都会自动被更新。

注意：创建模板时，应将其保存在工程目录中。当备份工程时，它也会随之备份。不要修改提供的标准模板。当编辑完模板后，必须使用**更新页面**命令（在**工具**菜单中）更新基于该模板的每个画面。注意，模板的属性不会被自动更新。

创建新模板：

1. 单击**新建**，或选择**文件 | 新建**。
2. 单击**模板**。
3. 选择一个**模板**并在此基础上创建新模板。
4. 为模板选择一种**风格**。
5. 根据需要选中或清除**链接**和**标题栏**选项。
6. 为模板选择**屏幕分辨率**。
7. 点击**确定**。

打开现有模板：

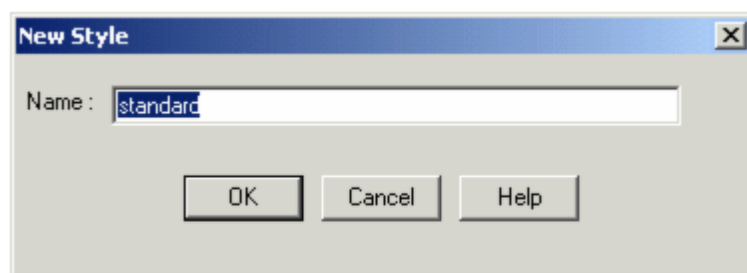
1. 单击**打开**，或选择**文件 | 打开**。
2. 选择**类型：模板**。
3. 选择存放模板的工程。
4. 选择**模板**。
5. 点击**确定**。

注意：要从工程中删除模板，请选定该模板的名称，然后点击**删除**。

保存当前模板：

1. 单击**保存**，或选择**文件 | 保存**。
2. 选择要保存模板的工程。
3. 单击**确定**。

注意：要为模板创建新风格，请点击**新建**。使用“新风格”对话框创建新的模板风格。



另请参见
[新风格对话框](#)

新建风格对话框

使用此对话框，您可以创建新的模板风格。在**名称**文本框中，输入新风格的名称。

另请参见
[创建自己的模板](#)

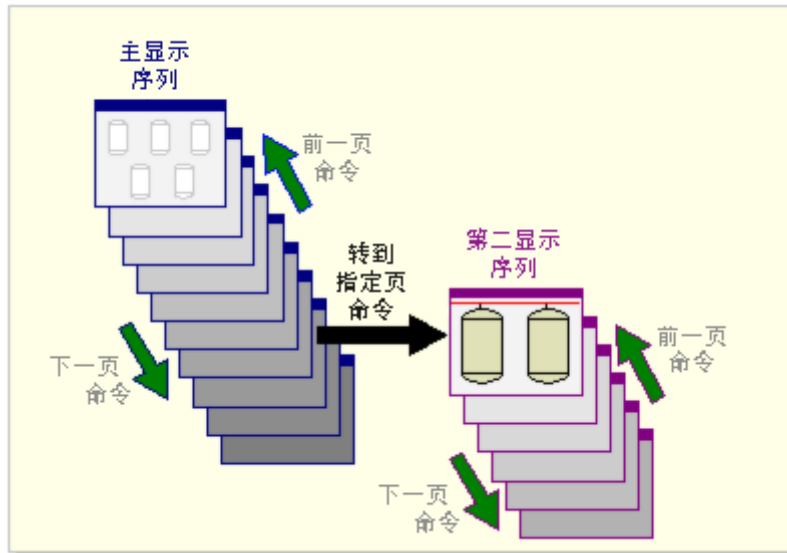
使用浏览顺序

用户可以将相关的画面按照一定的浏览次序连接起来。浏览顺序为系统中的画面建立了一种线性的浏览顺序。

在定义图形画面时，用户可以指定在浏览顺序中的何处对画面进行显示。在一个浏览顺序中，操作人员通过选择**前一页**或者**后一页**命令（或者在每个画面上定义一组类似的按钮）来选择对前面一个画面或者后面一个画面进行显示。

当用户第一次对某个画面进行保存时，该画面将被自动地添加到浏览顺序中去。

用户还可以通过定义一个可以将某个画面在另一个（第二个）顺序中显示的 **Page GoTo** 命令来使用多浏览顺序。这种情况下，后一页以及前一页命令将显示第二个顺序中的后一页或者前一页，如下图所示：



用户不一定要采用一个浏览顺序。用户可以定义几个 Page GoTo 命令来在一种分级结构中对指定的画面进行显示。

另请参见
[指定启动画面](#)

指定启动画面

每一个系统都必须有一个启动页面。当启动 Runtime 系统时，启动页面首先显示在屏幕上。您可能会希望自己来设计一个启动页面来显示启动信息（如公司徽标）。

如果要为工程中使用特殊启动页面，绘制页面并以名称 **启动** 保存该页面。（默认情况下，名为“启动”的页面会在 Runtime 系统启动时显示。）

注意：您并非必须指定启动页面。如果不指定启动页面，将显示默认的启动页面。这个默认的启动页面上包含有能够选择显示用户的图形页面的命令按钮。可以通过计算机设置向导更改默认启动页面的名称。

另请参见
[调整页面大小](#)

调整页面大小

按照默认设置，图形编辑器中的新页面将会添满整个显示区域。您可以根据需要重新调整其大小。可以：

- 在创建页面时指定其大小。
- 在页面创建后随时更改其大小。

- 将所有页面的默认设置指定为不同的尺寸。

另请参见

[页面（屏幕）分辨率](#)

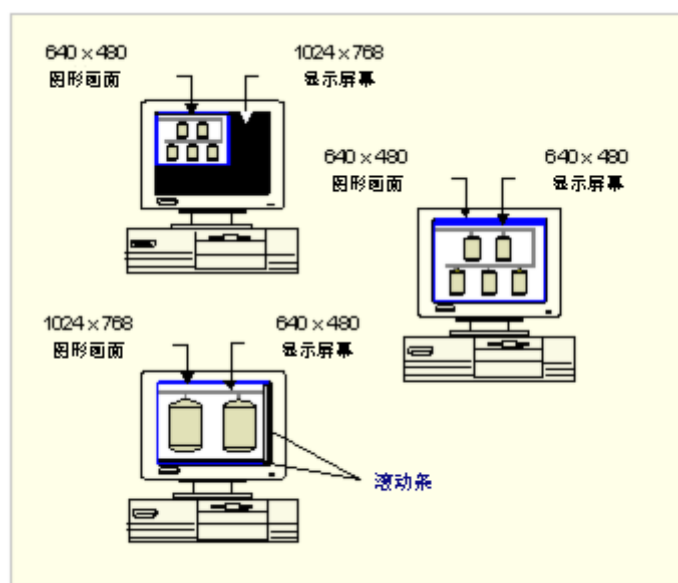
[运行时的页面大小](#)

画面（屏幕）分辨率

在绘制图形画面时，确定用来绘制画面的计算机的分辨率，以及要在 Runtime 系统中显示画面的屏幕的分辨率。

如果用来显示某个画面的显示器的分辨率高于该画面的分辨率，该画面显示时将比显示区域小。例如，用户在一台 VGA（640x480）的显示器上对画面进行绘制，并将其在一台 XGA（1024x768）的显示器上进行显示，该画面将被显示在显示器的左上角，并且将只占据显示器一半多一点的区域。

相反，如果当某个画面在一个分辨率比该画面分辨率低的显示器上进行显示时，该画面将大于显示区域。例如，如果在 XGA 屏幕上绘制画面，然后在 VGA 屏幕上显示该画面，那么它所占的区域将大于整个屏幕；只能使用滚动条滚动到画面的未显示区域。



另请参见

[运行时的画面大小](#)

运行期间页面的大小

按照默认设置，运行时显示的页面：

- 将显示同保存该页面时相同的页面大小，除非其父页面（调用该页面的页面）的大小被重新调整。

- 以恢复状态显示（可以点击**最小化 (Maximize)**）。

可以拖动窗口框来调整页面的大小。

另请参见
[调整页面大小](#)

定义画面属性

可以定义图形画面的属性。

要设置画面，请执行以下操作：

1. 在图形编辑器中打开该画面。
2. 选择文件 | 属性。
3. 使用“画面属性”对话框定义图形画面的属性。

另请参见
[画面属性 - 常规](#)
[画面属性 - 外观](#)
[画面属性 - 键盘命令](#)
[画面属性 - 事件](#)
[画面属性 - 环境](#)
[设置缺省画面设置](#)

页面属性—通用

图形页面具有如下属性：

窗口标题

运行期间显示在页面上的标题。一个窗口标题最多 64 个字符。

描述

输入页面描述与其各种函数诸如此类，最多 250 个字符。此描述仅用于注释不影响系统执行，运行时也不显示。

前一页

页面将会跳转到运行期间浏览序列（**browse sequence**）（最大 64 个字符）中位于当前页面前面的一幅页面。从菜单选择现有页面或直接键入页面名称。

此属性是可选的。如果用户没有指定前一页，那么在运行期间显示当前页面时前一页（**Previous Page**）命令是无法执行的。

后一页

页面将会跳转到运行期间浏览序列（**browse sequence**）（最大 64 个字符）中位于当前页面后面的一幅页面。从菜单选择现有页面或直接键入页面名称。

这个属性是可选的。如果用户没有指定一个后面的页面，那么在运行期间显示当前页面时后一页 (Next Page) 命令是无法执行的。

[安全] 所有区域 ([Security] All areas)

如果用户希望该页面属于所有区域（该页面可以被任何具有可以查看至少一个区域 (area)；权限的操作员所查看——参考 [增加用户记录 \(Adding userrecords\)](#)）请选中复选框。

[安全] 区域 ([Security] Area)

输入该页面所属于的区域。只有拥有访问区域权限的使用者能查看此页。点击菜单并选择某个区域，或者直接键入某个区域序号。如果用户没有指定某个区域，则所有的用户可以查看该页面。

[页面扫描时间] 默认 ([Page scan time] Default)

页面扫描时间 (Page Scan Time) 定义为在运行期间对图形页面进行刷新的频率。页面扫描时间同时也决定了页面显示期 (While Page Shown) 事件（即在运行期间显示页面时所执行的命令）执行的速度。

选中此复选框来采用默认的页面扫描时间（同采用参数[Page]ScanTime 设定的一样），否则，不选此复选框，并在下面的域中键入（或选择）另一个数值。例如，如果用户键入的页面扫描时间为 200 毫秒，将会试图每 200 毫秒对页面进行一次刷新，并且任何页面显示期 (While Page Shown) 事件将会每 200 毫秒刷新一次。

注意： 用户可以使用计算机设置向导 (Computer Setup Wizard) 来对默认页面扫描时间进行设置。

[日志] 日志设备

页面的键盘命令信息将会被记载到该设备的日志中去。点击此域右边的菜单来选择一个设备，或直接输入一个设备名。

注意： 必须把 *MsgLog* 字段包含在要发送的信息的记录设备的格式中。

[集群环境] 从调用程序继承

选中此复选框来从页面或打开此页的 *cicode* 任务（例如使用 *PageDisplay* 或 *WinNewAtCicode* 函数）使当前页继承默认集群环境。如果此项已经被选中，使用者不能选中某一特定的默认集群。参见 [关于集群环境](#)来了解更多信息。

[集群环境] ([Cluster context]) 集群 (Cluster)

指定此页的缺省集群环境。

单击**应用**然后单击**确定**。要为该页面进一步定义属性，请选择相关分页标签。

另请参见
[定义页面属性](#)

页面属性—外观

使用**外观**标签上的控件定义图形页面外观。

图形页面具有下列的外观属性：

[模板] 风格 ([Template] style)

系统在运行时图形页面的风格（外观）。利用已经存在的页面默认可以设置一种默认的风格（应用于所有的新建页面）并用页面属性设置模板。

多数用户更愿意采用标准风格。用户可以在图形和模板下的包含工程 (Include Project) 中查看预定义风格 (pre-defined styles)。

[模板] 分辨率 ([Template] resolution)

页面的默认屏幕分辨率：

显示器类型	宽度（像素）	高度（像素）
VGA	640	480
SVGA	800	600
XGA	1024	768
SXGA	1280	1024
User	****	****

[模板] 名称 ([Template] name)

页面所基于的模板名称。从菜单中选择模板名称。

注意：如果要查找一个自己建立的模板，用户必须检查在上面输入了正确的**风格和屏幕分辨率**。

[模板] 显示标题栏 ([Template] Show title bar)

决定是否显示 Windows 的标题栏（在页面的上方）。标题栏包括窗体的名称、最小化、最小化和关闭按钮（在标题栏的最右端），以及控制菜单按钮（在标题栏的最左端）。

如果要查找一个自己建立的模板，用户必须确保输入了正确的要以全屏的方式来显示页面（不显示标题栏），页面的大小必须同屏幕的大小相同（或者要大些）。如果页面的尺寸

小于屏幕的大小，即使在全屏的模式下，标题栏将仍然被显示出来。标准的模板风格对于两种页面大小都适用。

[查看区域] 宽度 ([View area] width)

在运行时操作员所能够查看到的区域的宽度（像素）。点击向上或者向下箭头按钮来增加或者减少区域的宽度，或者直接键入另外的数值。

[查看区域] 高度 ([View area] height)

在运行时所能够查看到的区域的高度（像素）。点击向上或者向下箭头按钮来增加或者减少区域的高度，或者直接键入另外的数值。

背景颜色

图形页面的背景中所显示的颜色。

在本对话框的右侧的预览区域中将会显示该页面的图像。单击**应用**，然后单击**确定**。要为该页面进一步定义属性，请单击相关分页标签。

另请参见

[定义页面属性](#)

画面属性 - 键盘命令

利用键盘命令属性，用户可以为页面定义键盘命令。一个键盘命令是一组特别的键序列，当它在运行时被操作员敲击时执行的命令。

你也能定义每次此键序列被输入时记录的信息。

如果操作人员的权限不满足下面的**安全 (Security)**中所指定的条件，将不能在运行时键入该页面的键盘命令。

键盘命令有以下属性：

键顺序 (Key sequence) (32 个字符)。

在此输入操作员能够输入以执行命令的键序列。

你可以输入任意多的键序列。如需添加一个键序列，点击**添加**，并键入键序列或从下拉列表里选取。如需编辑一个已有的键序列，点击相关的行，并点击**编辑 (Edit)**。使用者也可以点击**删除 (Delete)**来删除键序列。

键序列命令

当选取的键序列被输入时马上执行的命令 (系列指令)。键序列命令最大长度为 254 个字符。

如需插入标签或函数，单击右边的**向导 (Wizard)**快捷方式按钮。此按钮显示两个选项：**插入标签**和**插入函数**。

[安全] 等同页面区域

选取此复选框来给某个键盘命令分配同页面相同的区域（参考 [页面属性—常规](#)）。只有能够访问此区域的(有所需的权限的)用户能够发出此命令或记录信息。如果想把此键盘命令分配给另一个区域，请勿选择此选项：输入下面的另一个区域。

[安全] 命令区域

输入此键盘命令所属的区域，最多 16 个字符。只有能够访问此区域的（有所需的权限的）用户能够发出此命令或记录信息。例如，如果用户输入区域 1，只有对区域 1 拥有权限的操作人员才能够使用该命令。

从菜单点击某个区域或键入区域数字。

[安全] 无权限限制

打勾此选项来禁止权限限制；否则，保留其为空白，并在下面输入另外的权限。

不分配权限限制的结果根据您是否在安全设置里使用区域而有所不同：

- **无区域 (No Areas)**：所有的操作人员都对该页面拥有完全的控制权。
- **区域**：操作人员仅需要拥有对该区域的操作权限就可以对整个页面进行控制（参考 [增加使用者记录](#)）。

[安全] 权限等级

输入用户为能够发出此命令或记录信息而所必须具备的权限级别（最多 16 个字符）。例如，如果用户键入权限等级 1（Privilege Level 1），则操作人员必须拥有权限等级 1 才能够使用该命令。你也可以将此限制与区域限制组合起来使用(参见上面)。例如，如果你赋予键盘命令至区域 5，和权限级别 2，用户必须被设置成具有权限 2 和区域 5（请参见 [增加使用者记录](#)）。

从菜单选择某个权限或键入区域数字。

[日志] 日志信息

操作员在运行时间执行所选的行为时发向记录设备的 *MsgLog* 字段的一个文本信息（最大 32 个字符）。该信息可以是常规文本，超级精灵语法，或者是两者的结合。

如果包括字段数据使其作为记录信息的一部分,必须在配置设备时插入字段名使其成为设备格式的一部分。例如,在**设备**视窗的**格式**字段中,可以输入 *MsgLog,20 FullName,15*。这将会容纳如 **P2 started by John Smith** 这样的信息记录。

接收消息的设备是在页面属性的常规 (General) 属性中进行指定的。

单击**应用**，然后单击**确定**。单击**清除属性**可清除属性的详细内容，然后禁止属性。要为该页面进一步定义属性，请单击相关分页标签。

另请参见
[定义页面属性](#)

页面属性 - 事件

使用**事件** 标签来给图形页面赋予命令。当页面打开或关闭时或者说无论页面是否打开，这些命令都能执行。用户还可以定义不同的消息，这些消息将在执行命令的同时被写入日志。

页面事件有以下属性：

事件

命令可以附加于三种事件。用户可以选择多于一种类型的事件。每种类型的事件都可以附带有其专门的命令（也就是说用户可以在页面开启时执行一个任务，在页面关闭时执行另一个命令）。

[事件] 进入页面时

如果用户希望在页面第一次被显示时执行某个命令则选择该选项。例如，用户可以执行某个命令将一个数据库中的配方数据抽调到某个 **Cicode** 变量中，并且在该页面上进行显示。

注意： 如果命令使用 ActiveX 控件，则使用“页面显示时”事件。“进入页面时”事件可能不会正确地初始化 ActiveX 控件。

[事件] 退出页面时

如果用户希望在操作人员退出页面时执行某个命令选择该选项。例如，此命令可以用来关闭一个在页面开启时打开的数据库。

请勿调用下列函数：`PageGoto()`、`PageNext()`、`PagePrev()`、`PageDisplay()`，或者 `PageLast()`。

注意： 如果用户关闭系统，当前已经打开的页面的“退出页面时”函数将不会执行。如果某个特定的页面退出代码必须被执行，应当在调用 `Shutdown()` 函数之前调用该代码。

[事件] 页面显示期

如果用户希望某个命令在页面开启期间被连续地执行则选择该选项。例如，**页面显示期** 命令可以被用来对该页面执行背景计算。

[事件] 显示页面时

当页面首次显示，并且其上面的所有对象（包括 ActiveX 控件）已经被初始化时，如果使用者要执行某个命令，选择该选项。如果要在命令中引用 ActiveX 控件，则应使用此事件，而不是“进入页面时”。

注意： 如果命令使用 ActiveX 控件，则使用“页面显示时”事件。“进入页面时”事件可能不会正确地初始化 ActiveX 控件。

另请参见 [PageInfo Cicode 函数](#)，类型 25。

命令

一旦所选的事件发生，这些命令（一组命令）（最多 253 字符）将会立刻被执行。

如需插入标签或函数，单击右边的**向导**快捷方式按钮。此按钮显示两个选择，**插入标签**和**插入函数**。

单击**应用**，然后单击**确定**。单击**清除属性**可清除属性的详细内容，然后禁止属性。要为该页面进一步定义属性，请单击相关分页标签。

另请参见
[定义页面属性](#)

页面属性—环境

使用**环境**标签给图形页面定义环境信息。可以给图形页面、模板、或者超级精灵增加、删除，或者编辑画面环境变量。

例如，当点击（一个**调整 (Tune) >>**按钮）来显示调整参数时可以设计所有的回路画面。用户可以采用环境变量来定义放大的窗体的尺寸。**DspGetEnv()** 函数可以被用作编写该按钮的 **Cicode** 代码的一部分来使得窗体放大。这意味着用来使窗体放大的 **Cicode** 函数可以是通用的。每次可以使用同样的 **Cicode** 函数。

注意：如果用户在某个模板中添加了环境变量，这些环境变量将被包括在所有在该模板基础上创建的页面中。然而，如果该模板的环境变量发生改变，那些页面上相应的变量将不会发生改变。要改变某个超级精灵的环境变量，请参考 [超级精灵环境变量](#)。

变量 (Variables)

环境变量应被添加到页面中。当页面在运行时被打开,创建这些变量。它们的值可以采用 **DspGetEnv()** 函数进行读取。当页面被关闭时，环境变量内存就被释放。在上面的这个例子中，用户应当添加一个变量来定义页面的宽度，再添加另一个来定义高度。

要添加一个环境变量，点击**添加 (Add)**。要对一个已经存在的环境变量进行编辑，选中它，并点击**编辑 (Edit)**。（当点击**添加**或者**编辑**按钮时，出现一个包含两个字段的小对话框，一个是属性，另一个是该属性的值）。要移除 (Remove) 某个环境变量，选中它，并点击**移除 (Remove)**。

另请参见
[定义页面属性](#)

设置缺省页面设置

您能定义所有新图形页面的设置。

要对所有新建的页面属性进行设置：

1. 选择**文件 | 缺省**。
2. 完成**页面缺省**对话框, 然后单击**确定**。

另请参见
[页面缺省值](#)

画面默认值

您可以使用“画面缺省值”对话框定义所有图形画面使用的屏幕分辨率和风格。
可以在创建或编辑画面时覆盖它们的这些默认值。

[模板] 分辨率 ([Template] Resolution)

标准图形画面（如报警画面和标准趋势画面）的缺省屏幕分辨率：

屏幕类型	宽度（以像素计）	高度（以像素计）
VGA	640	480
SVGA	800	600
XGA	1024	768
SXGA	1280	1024
User	****	****

[模板] 风格 ([Template] Style)

Runtime 系统中的图形画面的风格（外观）。您选择的风格是您所添加到该工程中的新画面的缺省风格。可以使用“画面属性”更改现有画面和模板的风格。

多数用户更愿意采用标准风格。您可以在图形、模板下的包含工程中查看所有预定义的风格。

[模板] 显示标题栏 ([Template] Show title bar)

确定是否在每个图形画面的顶部显示 Windows 标题栏。标题栏包括窗口的名称、最大化、最小化和关闭按钮（在标题栏的最右端），以及控制菜单按钮（在标题栏的最左端）。

要以全屏的方式来显示画面（不显示标题栏），画面的大小必须同屏幕的大小相同（或者要大些）。如果画面的尺寸小于屏幕的大小，即使在全屏的模式下，标题栏将仍然被显示出来。标准的模板风格对于两种画面大小都适用。

您可以在创建画面时覆盖画面的此默认值，也可以稍后再覆盖。

背景颜色

所有新图形画面背景中显示的颜色。

预览

该对话框还将显示应用缺省设置的画面的预览。

了解绘图环境

Citect 图形编辑器是一个多功能的绘图环境，使用它，您可以为您的工程开发一种高性能的功能界面。

另请参见

[网格](#)

[引导线](#)

[选项](#)

[颜色](#)

[缩放](#)

网格

可以使用网格精确地对齐和放置对象。网格被激活后，您创建、移动或调整尺寸的所有对象和对象组都会自动与最近的网格节点对齐。

要显示网格，请执行以下操作：

1. 选择视图 | 网格设置。
2. 点击显示网格复选框。

要对齐网格，请执行以下操作：

- 选择视图 | 对齐网格。

缺省情况下，网格会被设置为 8 像素，初始位置位于画面的左上角。

要更改缺省网格尺寸和位置，请执行以下操作：

- 选择视图 | 网格设置。将显示“网格设置”对话框。

另请参见：

[网格设置对话框](#)

网格设置对话框

此对话框使您可以为 [网格](#) 定义初始位置和间隔。可通过网格精确地对齐和放置对象。

间隔

网格点之间的水平和垂直间隔（以像素为单位）。可以指定的最小网格尺寸为 3 × 3 像素。

初始位置

指定网格的初始位置（基础点）。

显示网格

在屏幕上显示网格。

对齐网格 (F8)

激活网格。网格被激活后，您创建、移动或调整尺寸的所有对象和对象组都会自动与最近的网格节点对齐。

引导线

可以使用水平和垂直引导线作为直缘，精确地对齐和放置对象。当对象的边缘或中心靠近引导线时，该边缘或中心将与引导线对齐。

要设置引导线，请执行以下操作：

1. 选择**视图 | 标线设置**。
2. 要在画面上水平设置引导线，点击**水平**。
3. 为引导线输入一个位置（到画面顶部的距离），然后点击**设置**。
4. 要在画面上垂直向下设置引导线，点击**垂直**。
5. 为引导线输入一个位置（到画面左边缘的距离），然后点击**设置**。
6. 点击**显示标线**复选框。
7. 点击**对齐标线**复选框。
8. 点击**确定**。

要关闭引导线，请执行以下操作：

- 选择**视图 | 对齐标线**。

要移动引导线，请执行以下操作：

1. 在引导线上移动光标。
2. 点击并按住鼠标左键。
3. 将引导线移动到一个新位置。
4. 松开鼠标键。

要用鼠标创建新引导线，请执行以下操作：

1. 在引导线上移动光标。
2. 按住 **Ctrl** 键。
3. 点击并按住鼠标左键。
4. 将引导线移动到一个新位置。
5. 松开鼠标按钮和 **Ctrl** 键。

要用鼠标删除引导线，请执行以下操作：

- 将引导线拖到画面边缘。

另请参见：

[引导线设置对话框](#)

引导线设置对话框

可通过此对话框分别定义 32 条水平和垂直 [引导线](#)。引导线充当直缘，使您可以精确地对齐和放置对象。当对象的边缘或中心靠近引导线时，该边缘或中心将自动与引导线对齐。

方向

引导线的方向（水平或垂直）。

显示引导线

在屏幕上显示引导线。

对齐引导线 (F7)

激活引导线。引导线被激活后，您创建、移动或调整尺寸的所有对象和对象组都会自动与最接近的引导线对齐。

选项

您可以为自己的绘图环境定义常规选项。

要为绘图环境定义常规选项，请执行以下操作：

1. 选择工具 | 选项。将出现“选项”对话框。
2. 输入相关的详细信息。

选项	描述
新建时显示属性	使相关属性对话框在向画面中添加对象时自动显示。
复制时显示特性	使相关属性对话框复制画面上的对象时自动显示。
保存模板警告	使提醒信息在您修改然后保存现有模板时显示。修改现有模板时，与之关联的所有图形画面都不会被更新，直至您执行“更新画面”，根据此模板更新每一画面。
修改 AN 域	允许您修改任何对象的动点 (AN) 数。注意，您不能将 AN 更改为图形画面中的现有 AN 数。
禁止精灵窗口	禁止精灵窗口在新精灵被添加到画面或现有精灵被编辑时显示。如果已禁止精灵窗口，针对精灵的每个固有对象会显示一个窗口。
显示组按钮	显示精灵对话框中的组按钮。组按钮针对精灵的每个固有对象显示一个窗口。
编译询问消息	在修改工程和选中 运行 后启用“要编译吗？”消息框。在 运行 被选中后，工程通常会被自动编译（如果此工程已被修改）。
快速 Runtime 显示	快速显示 Runtime 系统中的图形画面。
列出系统画面	指定系统画面将被包含在： <ul style="list-style-type: none"> • 图形编辑器的“打开和保存”对话框中的画面列表。 • “上一个画面属性”和“下一个画面属性”，用于定义画面的浏览次序。

选项	描述
	<ul style="list-style-type: none"> • Citect 浏览器内容区域中的文件列表。 系统画面带有一个感叹号 (!) 前缀。
显示 3.xx/4.xx 版工具	启用旧版 (第 3 版和第 4 版) 工具箱。此工具箱中包含一些就工具 (如滑钮和条形图), 因为可以使用对象属性进行配置, 这些工具已经不再需要。
快速更新画面	影响图形编辑器的“更新画面”工具的操作。如果已选中“快速更新画面”, 图形编辑器将只更新修改过的画面。如果未选中, 所有工程画面都会被更新。
透明粘贴	允许您指定在位图被粘贴在图形画面上时变为透明的颜色。这适用于粘贴自剪切板的或从其它应用程序导入的位图。 注意: 其它应用程序本质上不支持透明数据位。如果将某位图粘贴到外部应用程序中, 透明位将显示为透明粘贴颜色。
ActiveX 自动画面保存	使您可以在插入 ActiveX 控件前保存图形画面。如果您插入自定义的内置 ActiveX 控件 (通常会导致图形编辑器停止工作), 这可以防止数据丢失。启用“自动画面保存”后, 如果插入不稳定的 ActiveX 导致这种事件, 您可以最小化工作丢失的可能性。您可以在重新打开图形编辑器时正常恢复保存的画面。 其中的选项有: <ul style="list-style-type: none"> • 插入 ActiveX 控件前保存画面: 图形画面使用当前画面名称自动保存。 • 在保存前提示: 显示一则询问, 问您是否希望在插入 ActiveX 控件前保存画面。 • 不自动保存: 图形画面不会被自动保存, 询问也不会显示。

3. 点击**确定**。

另请参见
[选项](#)

颜色

所有静态和动画对象 (包括画面背景、导入图像、符号、元文件和位图) 都支持真彩色 (1670 万种颜色)。

只要特定画面或对象属性有一个颜色值, 就会显示当前颜色按钮。

要选择与按钮上当前显示的颜色不同的颜色, 请单击向右的小黑色箭头。这会启动颜色选择器。

颜色选择器

颜色选择器的前 11 行显示标准的一组颜色, 包括透明色 (用黑色十字标记)。其余行显示用户定义的颜色, 称之为**收藏颜色**。这包括闪烁颜色, 表示为用对角线分开的双颜色块。

要选择显示在颜色选择器上的一种颜色, 请单击它。

如果没有显示所需颜色, 可以选择创建自定义颜色, 或者从一个图形画面中匹配一种现有颜色。

匹配图形画面上的现有颜色：

1. 确认要匹配的颜色存在于当前显示在图形编辑器中的画面上。
2. 从颜色选择器中，选择“颜色选取器”工具（外观类似取色器）。
3. 使用“颜色选取器”单击要匹配的颜色。
4. 已选择的颜色现在将会再现在“颜色值显示”按钮上。

创建自定义颜色：

1. 在颜色选择器中，单击“编辑”按钮。这将调出“编辑收藏颜色”对话框。
2. 使用“编辑收藏颜色”对话框创建要使用的颜色（有关详细信息，请参见 [编辑收藏颜色对话框](#)）。
3. 如果需要，为颜色指定一个名称。
4. 使用**添加**按钮将该颜色添加到颜色选择器上显示的“收藏颜色”中。
5. 单击**确定**。现在，就可以选择您刚刚创建的颜色了。

另请参见

[编辑收藏颜色对话框](#)

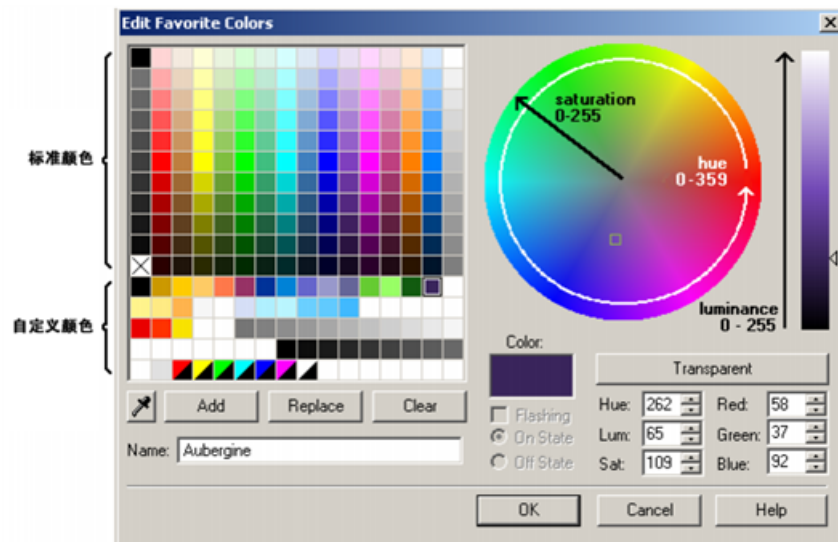
[交换颜色对话框](#)

[调整颜色对话框](#)

“编辑收藏颜色”对话框

使用“编辑收藏颜色”对话框，您可以实现以下操作：

- 可以通过点击色轮，直观地选择您希望使用的颜色。
- 可以通过点击亮度条来确定颜色的亮度级别。
- 可以通过输入色调、饱和度和亮度的值来创建颜色。
- 可以通过输入红、绿和蓝的颜色值来创建颜色。
- 可以通过添加、替换和删除颜色网格中的颜色来修改“颜色选择器”中可用的颜色。
- 可以对颜色进行命名，使其能够通过工具提示在颜色网格中被识别出来。



“编辑收藏颜色”对话框包含以下元素：

颜色网格

显示预定义的颜色选择。前 11 行显示一组标准的颜色，包括透明色（用黑色十字标记）。其余各行显示当前添加为收藏的用户预定义颜色。包括闪烁颜色在内，表示为用对角线分开的双颜色块。

此网格中出现的颜色在“颜色选择器”中都可用。

添加

将颜色面板中当前显示的颜色添加到颜色网格中用户预定义的收藏。如果在颜色网格中没有可用的位置，您可以改为使用**替代**或**清除**按钮。

替代

此按钮允许您改变颜色网格中的颜色。选择要更改的颜色，调整其颜色值，然后点击**替代**按钮。将对选中的颜色进行更新，应用所做的更改。

清除

删除选中的颜色，使其在颜色网格中的位置保留为“未使用”。

名称

使您能够将某名称与某个预定义的颜色进行关联。在“颜色选择器”中可以将该名称作为工具提示查看，从而更加方便地在众多相似的颜色中区分出特定颜色。

输入一个名称，然后点击**添加**按钮，您就可以将该名称与某个新建的颜色相关联。也可以将某个名称用于现有的颜色，方法是：选择该颜色，键入一个名称，然后点击**替代**按钮。

注意：预定义的颜色标签已经被定义为颜色名称。

颜色

颜色面板显示的颜色都是由“编辑收藏颜色”对话框所应用的当前设置创建的。

请注意，“编辑收藏颜色”对话框中显示的值会进行自动调整，以正确表示此面板中当前显示的颜色。每个字段中的值都不是独立的。

色轮

使用色轮，您可以直观地选择颜色。它表示了一个循环布局中颜色的完整色谱，其颜色饱和度不断增加，直到超出范围。只需点击色轮，即可选择颜色。

亮度条

使您可以直观地选择要应用于某颜色的亮度。点击亮度条上的适当位置，即可应用相应的亮度级别。随着亮度条中颜色的不断增加，表示颜色从纯黑逐渐变为纯白。

色调

指定**颜色**面板中当前显示的颜色色调值。色调主要用于区分不同的颜色。色调值的范围为 0（零）到 359，表示色轮的程度。例如，0 表示右侧纯红，180 表示左侧纯青。

饱和度

指定**颜色**面板中当前显示的颜色饱和度级别。随着饱和度级别的增大，所选颜色将从灰色逐渐变为纯主色。饱和度值的范围为 0（零）到 255。

亮度

指定**颜色**面板中当前显示的颜色亮度值。它表示颜色的亮度，随着该值的不断增大，颜色将从黑变为纯白。亮度值的范围为 0（零）到 255。

注意：当使用 HLS 值创建颜色时，您可能会发现，当重新打开此对话框时，为某颜色指定的 HLS 值已经发生变化。原因在于，RGB 值不如 HLS 值精确，有时会导致多个 HLS 值分配了同一个 RGB 值。结果，从 RGB 值生成 HLS 值时，有些值就会发生改变。

红

表示用于创建**颜色**面板中当前显示颜色的红色成分。该值的范围为 0（零）到 255。如果要使用 RGB 值创建颜色，需要调整此设置。

绿

表示用于创建**颜色**面板中当前显示颜色的绿色成分。该值的范围为 0（零）到 255。如果要使用 RGB 值创建颜色，需要调整此设置。

蓝

表示用于创建**颜色**面板中当前显示颜色的蓝色成分。该值的范围为 0（零）到 255。如果要使用 RGB 值创建颜色，需要调整此设置。

透明

点击此按钮可以选择透明度。一旦将透明作为一种颜色，则会显示背景颜色或透明对象的背景颜色。

闪烁

如果希望创建闪烁颜色，请选中此复选框。在颜色网格中，闪烁颜色显示为用对角线分开的面板。要创建闪烁颜色，您必须选择**开状态**和**关状态**颜色。

交换颜色对话框

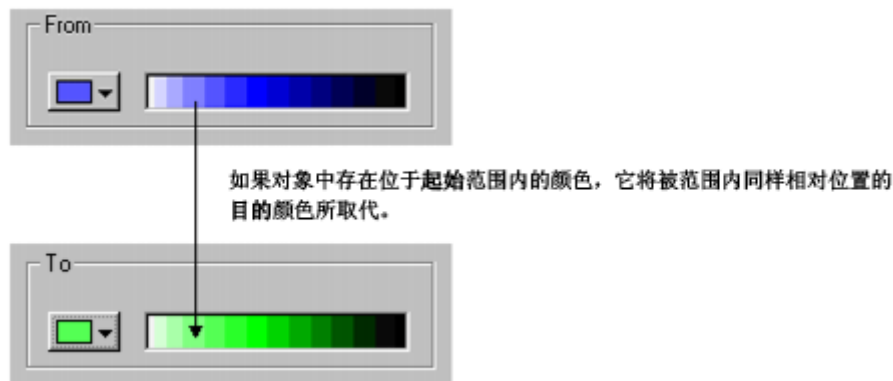
可以使用交换颜色对话框将对象的颜色（或对象组、位图）交换为新颜色。

注意：如果位图中包含亮度值很高或很低的区域（即白色或黑色区域），当交换其中的颜色时，可能会发现意外的结果。这是因为这些区域几乎不包含任何颜色，只有亮度。例如，无论如何调整位图其它部分的颜色，代表反射高光（例如钢传送带发出的明亮光）的

区域将保持无色。因此，在使用交换颜色对话框和其它颜色工具前，您必须熟悉色彩、亮度、饱和度的概念，以防止图形页面上图像的效果与预期不符。

从

对象的当前颜色。如果点击**交换范围**复选框，它将显示一系列亮度各异（从白到黑排列）的颜色。存在于对象中的此范围内的任何颜色都将被**到**范围内的相应颜色替换，如下所示：



到

原始对象颜色要改成的颜色。如果点击**交换范围**复选框，它将显示一系列亮度各异（从白到黑排列）的颜色。这使您可以一次交换整个颜色范围。

替换所有颜色

指定将对象中所有颜色变为新颜色。

交换范围

指定交换一系列颜色。**从**和**到**框指示范围内的开始颜色。

注意：不能使用选定的**交换范围**反转颜色。这表明，举例来说，不能在一次交换中将淡绿色交换为深红色并将深绿色交换为淡红色。

“调整颜色”对话框

可以在“调整颜色”对话框中，对图形编辑器对象进行再映射，以改进颜色。

色调（程度）

用户可以在“色调”区域中设置要映射到或从中映射的色调范围。显示的栏跨越的值范围为0（零）到359度，表示色调颜色值的循环性质。可以在右侧的字段中键入每个滑钮的数值。

- **起始色调范围**栏上方的滑钮用于选择将映射的颜色范围的起点。
- **起始色调范围**栏下方的滑钮用于选择将映射的颜色范围的终点。

- **结束色调范围**栏上方的滑钮用于选择将映射到的颜色范围的起点。由于值范围是循环的，因此选定的区域可以环绕至该栏的左侧。

请注意，选择排除的颜色范围将灰显，允许从视觉上估计选定的范围。

亮度 (%)

借助亮度滑钮，可以在 -100% 到 100% 的范围内提高颜色的亮度。如果滑钮增加到零以上的值，则颜色将逐渐向白色变化。如果滑钮被设置为零以下，则颜色将逐渐向黑色变化。可以在右侧的字段中输入滑钮的数值。

仅限已选色调复选框仅将亮度设置应用于将再映射的颜色。如果将该框保留为取消选中状态，则可以针对所有颜色调整亮度。

饱和度 (%)

借助饱和度滑钮，可以在 -100% 到 100% 的范围内提高颜色的饱和度。如果滑钮增加到 0 以上的值，则颜色将逐渐向主色变化。如果滑钮被设置为零以下，则颜色将逐渐向灰度变化。可以在右侧的字段中输入滑钮的数值。

仅限已选色调复选框仅将饱和度设置应用于将再映射的颜色。如果将该框保留为取消选中状态，则可以针对所有颜色调整饱和度。

缩放

使用缩放命令来查看正在绘制对象的放大视图。缩放命令在屏幕上显示一个缩放框，它可以放大光标周围区域，这样能使用户精确的定位或绘制对象。

程序	描述
显示缩放窗口	选择 视图 显示缩放 。
隐藏缩放窗口	双击控制菜单窗口（在缩放窗口上）或者选择 视图 显示缩放 。

用户可以把光标变为十字光标就能扩展正绘制区域的全部宽度和长度。当从绘制区域移开时，正常的光标重新出现，这时允许选择命令和工具。

过程	描述
在十字光标和正常光标间切换	选择 视图 十字光标 。
隐藏状态栏	选择 视图 显示状态栏 。再选择此命令来重新显示状态栏。
隐藏工具箱	双击控制菜单窗口（在工具箱上）或者选择 视图 显示工具箱 。再选择此命令来重新显示工具箱。
隐藏工具栏	选择 视图 显示工具栏 。再选择此命令来重新显示工具栏。

过程	描述
当正在使用鼠标时，改变鼠标移动的速度	选择视图 鼠标低速。
当正在使用光标键时，改变光标移动的速度	选择视图 光标键低速。 提示： 通过使用左、右、上和下光标键来移动屏幕上的光标。

使用库

可以将经常使用的对象或对象组（包括位图对象）作为符号存储到库中。然后将这些符号粘贴到页面中。

与任何其它类型的对象一样，将符号从库中粘贴到图形页面后，可以移动、调整大小、改变形状、置于顶层和编辑其属性。

可以将符号从库中粘贴到页面中：

- 作为**未链接**的符号；粘贴的符号不会随库中符号的更改而更新。
- 作为**链接**的符号；页面中的符号会在库中的符号更改时更新（以改变库中符号的属性、打开库和在库中对其进行编辑）。如果对页面中符号进行编辑，然后更改库中的源符号，则粘贴的符号将发生变化。例如，如果将粘贴的符号大小增大两倍，则库中符号的大小也会增大两倍，粘贴的符号再次增大两倍。可以通过使用**编辑 | 切断链接**命令切断指向库的链接。

将对象保存到库中时，该对象的当前属性也随之一同保存。将对象作为符号粘贴到图形页面中时，它们会用作符号的缺省设置。粘贴的符号的外观属性与正常对象的外观属性不同；用户只能指定可见性属性。

将符号粘贴到页面中时，不能通过双击符号从页面中访问构成符号的对象。要显示这些对象的属性，请按下 **Control (CTRL)** 键，同时双击特定对象。或者，可以从“工具”|“组”中选择“定位对象”，然后单击**确定**。但是，如果指向库的链接被保留，则这些属性中的大多数属性将为只读。

例如，如果已定义了具有特定安全分类的命令按钮，并需要在大量图形页面中使用该按钮，这时候符号非常有用。可以将其保存为符号，每次需要使用按钮时，从库中粘贴该符号即可。每次粘贴时，符号的属性都相同。

注意：可在工程中使用的符号范围非常广泛。这些符号存储在“包含”工程的多个库中。保存库时，库名称的前八个字符必须对该库唯一。

复制对象到库中

您可以将一个对象复制到库中以便在其他图形页面中使用。

复制对象到符号库中（即创建一个符号）：

1. 单击**选取**。

2. 选取一个（或一组）对象。
3. 选择**编辑 | 复制到库**。将出现“复制到”对话框。

“复制到”对话框

这个对话框使您可以复制一个对象(对象组)作为库中的一个符号。

特性	描述
符号	符号的名字。
库	存储符号项目库。
项目	库存储到的项目。
激活预览	显示符号的预览图象。

注意：要编辑符号，单击**编辑**。要生成一个新符号，单击**新建**。

“新建库”对话框

这个对话框使您建立一个符号库、精灵库或者超级精灵库（最多 32 个字符）。为新库键入一个**名字**。（该页面名称的前八个字符必须同其它页面名称相区别。）

使用符号

可以创建在图形页面中使用的符号。

如何创建新的符号：

1. 单击**新建**，或选择**文件 | 新建**。
2. 选择**类型：符号**

注意：也可以在页面中创建对象或对象组，然后选择**编辑 | 复制到库**。

如何打开现有符号：

1. 单击**打开**，或选择**文件 | 打开**。
2. 选择**类型：符号**。
3. 选择存储库的**工程**。
4. 选择存储符号的**库**。
5. 选择所需的符号。
6. 单击**确定**。

注意：要从库中删除符号，请选择符号名称，然后单击**删除**。

如何保存当前符号：

1. 单击**保存**，或选择**文件 | 保存**。
2. 选择存储库的工程。
3. 选择要存储符号的库。
4. 输入符号的名称。
5. 单击**确定**。

有关在图形页面中使用符号的详细信息，请参见 [粘贴的符号对象](#)。

位图

位图图像是用像素阵列（或点阵）而不是单个实体表示的绘制对象。位图作为单一对象处理，可以对它进行移动、复制和改变形状等操作。由于可以编辑位图中的单个像素，因此可以将位图用于晕映图像和图像混合。

可以用位图编辑器创建位图，也可以从其他基于 Windows 的位图编辑器中导入位图。

在实时系统中，位图的显示方式与其他对象不同。位图直接映射到屏幕上（即，图像中的每个像素对应于屏幕上的一个像素）。对象被储存为一系列指令，并按照它们被画到图形画面上的顺序画到屏幕上。

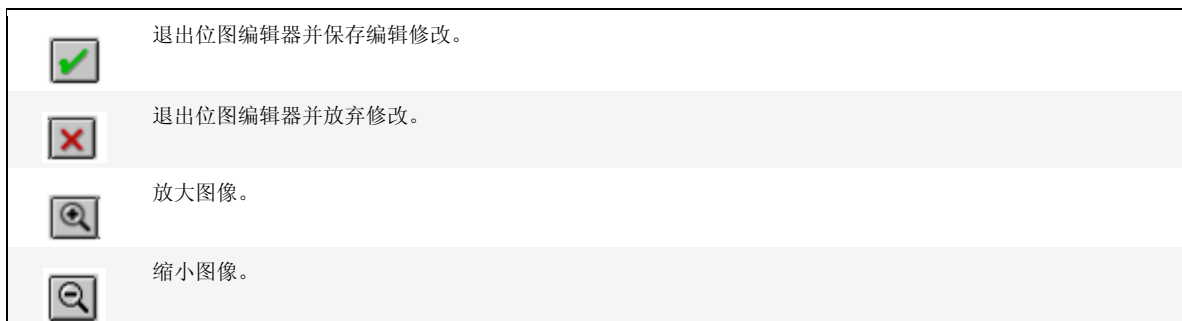
位图编辑器

可以使用位图编辑器创建和编辑位图图像。可以在图形画面上像使用动画符号一样使用位图图像。

位图编辑器的背景色总是透明的，显示为白色的像素中心有一个黑色的点。绘制时，单击（并按住）鼠标右键。

闪烁颜色用对分像素表示，指示所使用的开启状态和关闭状态的颜色。

位图编辑器的工具栏有以下按钮：





选择位图中一种颜色作为当前色（键盘快捷键为 **Shift+P**）。也可以从颜色样本中选择当前色。



显示**位图大小**对话框，可以在此查看图像的当前尺寸并编辑图像的边线。

调整位图的大小：

1. 在图形编辑器中，单击位图。
2. 选择**工具 | 位图编辑器**，或者按 **F9**。
3. 单击**调整大小**。随即会显示“位图大小”对话框。
4. 选择一种模式。单击**变大**可放大图像，单击**缩小**可缩小图像。
5. 针对位图的上下左右指定增加或删除多少个像素点，然后单击**确定**。

将位图中的一种颜色作为当前色：

1. 在图形编辑器中，单击位图。
2. 选择**工具 | 位图编辑器**，或者按 **F9**。
3. 单击**取色器**，然后单击图像中的一种颜色。选定的颜色即成为当前色，在图像中的其他位置单击即可应用该颜色。

将一个（或多个）对象转换为位图：

1. 选择对象。
2. 选择**工具 | 转换为位图**。

注意：只有在 8 位 (256) 颜色模式下才能执行“转换为位图”操作。

调用位图编辑器：

- 选择**工具 | 位图编辑器**。

（从其他应用程序）粘贴位图：

1. 在外部应用程序中创建图像。
2. 使用该外部应用程序的复制命令将图像复制到计算机的剪贴板。
3. 切换到图形编辑器。
4. 选择**编辑 | 粘贴**。

可通过选定对象然后选择**编辑 | 属性**来编辑粘贴的位图。

导入图形：

1. 选择**文件 | 导入**。随即会显示“导入”对话框。
2. 使用“导入”对话框选择要导入的文件。
3. 单击**确定**或单击要导入的文件并将它拖到图形编辑器中的画面上。

可通过选定对象然后选择**编辑 | 属性**来编辑导入的位图。

导入闪烁图形：

1. 选择**文件 | 导入为闪烁的**。随即会显示“主导入”对话框。
2. 选择要用于闪烁图像的第一个文件。
3. 单击**确定**。随即会显示“闪烁导入”对话框。
4. 选择要用于闪烁图像的第二个文件。

“导入”对话框

可以使用“导入”对话框导入其他应用程序产生的图形。

如果选择了**导入为闪烁的**，则会相继出现两个“导入”对话框，允许您选择要设置为闪烁符号的两个图像。**导入主要内容**对话框允许选择所使用的初始图像；**导入闪烁**对话框允许选择要使用的第二个图像。

查找

图形所在的盘符和目录。

文件名

图形文件名

文件类型

图形文件的类型。支持下列文件格式：

- Windows 3.0 位图 (*.BMP, *.DIB, *.RLE)
- PCX 格式的位图 (*.PCX)
- 文本文件 (*.TXT)
- AutoCAD DXF 文件 (*.DXF)，仅限二维。也支持二进制格式。
- Windows 图元文件 (*.WMF)
- EPS 文件 (*.EPS)
- 传真图像 (*.FAX)
- Ventura 图像 (*.IMG)
- JPEG 文件 (*.JPG, *.JIF, *.JFF, *.JGE)
- PCD 类型文件 (*.PCD)
- 可移植网络图形文件 (*.PNG)。（不支持具有 alpha 通道的 PNG 文件。）
- Targa 格式文件 (*.TGA)
- TIF 格式文件 (*.TIF)
- WordPerfect 文件 (*.WPG)

Chapter: 19 使用对象

对象是添加到图形页面中的基本画图实体。对象是使用画图工具箱里的工具绘制的，而且可以在绘制完成后移动、改变形状或复制。对象由一组属性定义，这些属性是在绘制对象时分配的，或者在绘制完成后通过双击分配（这些属性将覆盖任何与其有冲突的 Cicode 显示函数）。

大多数对象可以分配键盘命令和访问权限，并且可以在表达式返回某参数值或变量标签更改状态时，通过运行时动态更改对象的方式对其进行配置。甚至可以用作更改变量状态的滑钮。

注意：如果对象是组的一部分、粘贴精灵或符号的一部分或页面模板的一部分，则仍可以访问其属性。秩序按下 **Control (CTRL)** 键同时双击对象。或者，选择工具 | **定位对象**，单击对象，然后单击**确定**。但如果仍然存在指向原始精灵/符号/页面模板的链接，则对象属性大部分为只读。

每种类型的对象都有各自的工具：

	自由手画线对象		直线对象
	长方形对象		椭圆对象
	多边形对象		管道对象
	文本对象		数字对象
	按钮对象		符号集对象
	趋势对象		Cicode 对象
	粘贴的符号对象		粘贴的精灵对象
	ActiveX 对象		过程分析器用户指南中的“过程分析器对象”
	数据库交换控制对象		Vimeo Web 门限控制对象

另请参见
[使用组](#)
[改变对象的形状](#)
[使用位图](#)
[导入图形](#)

使用组

可以组合多个对象。组包含唯一一组属性（大多数属性与对象的设置相同），用于确定组作为整体的运行时行为。（组合中各个对象的属性保持不变。）通过定义组属性，可以指定整个组在特定运行时条件下的动态变化（例如，表达式返回某个参数值或变量标签更改状态时）。

要编辑或查看组的属性，请双击该组。如果页面中包含多个组，请选择**工具 | 定位对象**。这样就可以看到页面中的组和对象，以便更容易地选择要编辑的对象。而且可以显示组成组的对象或组的属性。（还可以通过按下 **Control (CTRL)** 键并双击对象来编辑组中对象的属性。或者，从**工具**菜单中选择**定位对象**，选择对象，然后单击**确定**。）如果页面中的组中包含组，此操作非常有用。

组可以是对象和其它组的混合。

另请参见
[改变对象的形状](#)

重塑对象

可以对管道、连线、多边形对象进行编辑，以改变其形状。这些对象中的每个对象都由被称为节点的结构支撑点间一系列连续绘制线条组成。选择对象时，节点可见。每个节点以一个位于特定的对象支撑点的小的方括号来显现。在一个连线或管道的起始和结尾处总会有一个节点，在对象形状的改变方向的地方也有节点。

管道、连线、多边形对象的形状能够以许多方式改变。节点可以被单个或按组选取并平移到不同位置，因此来改变对象的形状。管道、连线、多边形对象也支持节点的添加和删除。

重塑线条对象

线条对象也有节点，但是比起管道、连线、多边形对象来说就有很大的限制。

一条直线仅能够由两个节点组成，一个起始节点和一个结尾节点。可以单独选取这些节点，以将线条平移至不同的位置，或至少改变其方向。线条对象不支持节点添加。为了达到添加节点至线条对象同样的效果，可以创建一个连线对象来代替。删除一个线条对象两点之间的任何一点将会完全删除整个线条对象。

另请参见
[使用位图](#)

使用位图

位图图像是特殊对象，由像素或点阵列表示，而不是单独的实体。位图被当作可以移动、复制和改变形状的单对象。可以使用 **Citect** 位图编辑器创建和编辑位图。由于可以在位图中编辑各个像素，因此可以将位图用于更“艺术”的图像，如晕映和图像混合。

另请参见
[导入图形](#)

导入图形

图形编辑器有几种文件格式过滤器，允许你从其他应用程序导入图形，如画图程序、图解程序、演示软件包、扫描器等。图形被导入后，你就能够用图形编辑器来编辑图形了。

图形文件能够从一个第三方应用程序（如 **Windows** 资源管理器）拖动，然后放置在图形编辑器的画面上。

缺省情况下，被导入位图无效颜色将采用抖动方式进行处理。要禁用此功能，请在图形编辑器中选择 **工具 | 选项**，然后清除 **粘贴时模糊扩散位图** 选项。如果禁用了模糊扩散处理，无效颜色就会被调色板里最接近的颜色代替。

对象属性

对象属性是在“属性”对话框里定义的。当绘制对象时，会显示“属性”对话框，允许您定义所需的属性。

您也可以双击对象（或选择 **编辑 | 属性**）以显示“属性”对话框。

如果对象属于某个组，某个粘贴精灵或符号，或画面模板，您可以通过按 **CTRL** 键并双击该对象来访问对象属性。或者，选择 **工具 | 定位对象**，点击该对象，然后点击 **确定**。

另请参见
[外观](#)
[平移](#)
[缩放](#)
[填充](#)
[输入](#)
[滑钮](#)
[访问](#)

外观

点击 **外观** 标签来定义对象的外表，如线条的风格和阴影。您也可以说明对象何时将被隐藏而不被操作员看见（例如当变量 **DIGITAL_TAG** 为 **OFF** 状态时）。

“外观”标签左边的复选标记告诉您“外观”属性是否已被配置。对话框右下方的标签中的复选标记指明哪个属性已被配置。

点击其它标签以便为对象定义更多属性。大多数属性一同发挥作用，例如，一个对象能够同时拥有颜色填充、移动和缩放属性。

另请参见
[对象属性](#)
[移动](#)

移动

单击**移动**标签可垂直或水平移动对象或者旋转对象，这取决于表达式的返回或标签的状态。

“移动”标签左侧的复选标记表示配置移动属性的时间。对话框右下方标签中的复选标记指明已配置的属性。

单击其它标签可为对象定义更多属性。大多数属性一同发挥作用，例如，一个对象能够同时拥有颜色填充、移动和缩放属性。

另请参见
[对象属性](#)
[缩放](#)

缩放

单击**缩放**标签可以垂直或水平缩放对象，这取决于表达式的返回或标签的状态。

“缩放”标签左侧的复选标记何时配置“缩放”属性。对话框右下方标签中的复选标记指明要配置的属性。

单击其它标签可为对象定义更多属性。大多数属性是一起协作的，例如，一个对象能够同时拥有颜色填充、平移和缩放属性。

另请参见
[对象属性](#)
[填充](#)

填充

点击**填充**分页，指定用于填充对象的颜色，和填充的高度。填充属性可以显著变化，具体取决于表达式返回的值，或标签的状态等（例如，可以使用此分页可视化反映储存罐物料高度）。

“填充”分页左侧的复选标记说明是否已配置了填充属性。对话框右下方的分页中的复选标记指明哪个属性已经配置。

点击其它分页可以为对象定义更多属性。大多数属性是一起作用的，如一个对象能够同时拥有颜色填充、移动、和缩放属性。

另请参见
[对象属性](#)
[输入](#)

输入

点击**输入**分页标签以指定操作员点击对象时要执行的命令以及要记录的信息。你也可以为对象定义键盘命令，并且用区域和安全权限来限制其范围。

“输入”分页标签左边的复选标记告诉您何时已配置了输入属性。对话框右下方分页中的复选标记指明哪个属性已经配置。

点击其他分页标签可定义其他对象属性。大多数属性一同发挥作用，例如，一个对象能够同时拥有颜色填充、移动和缩放属性。

另请参见
[对象属性](#)
[滑钮](#)

滑钮

点击**滑钮**分页标签可将对象作为滑钮来使用。一个变量能与对象相关连，而且当操作员移动对象时，变量的数值就将改变。对象能够被设置成垂直和/或水平的滑动，或者旋转。

滑钮分页标签左边的复选标记告诉您滑动属性是否已经被配置。对话框右下方标签中的复选标记指明要配置的属性。

点击其他分页标签可定义其他对象属性。大多数属性一同发挥作用，例如，一个对象能够同时拥有颜色填充、移动和缩放属性。

另请参见
[对象属性](#)
[访问](#)

访问

点击**访问**标签为对象分配一个区域或权限。不具备相应访问权限的操作员将不能使用滑钮、对象特定的键盘命令等。它还允许您在某些运行条件下禁用此对象。这就意味着对象可被变花、变灰、或甚至隐藏。

“访问”标签左边的复选标记告诉您“访问”属性是否已被配置。对话框右下方的标签中的复选标记指明哪个属性已被配置。

点击其它标签以便为对象定义更多属性。大多数属性一同发挥作用，例如，一个对象能够同时拥有颜色填充、移动和缩放属性。

清除属性

点击**清除属性**以删除此属性的配置详细信息。

应用

点击“应用”以使所做的更改生效。借助“应用”，可以查看所做的更改，而不会关闭“属性”对话框。

另请参见
[对象属性](#)

操作对象

操作对象的方式很多，如移动、改变大小和分组等。

另请参见
[选取对象](#)
[移动对象](#)
[改变对象大小](#)
[删除对象](#)
[锁定/解锁对象](#)
[将对象分组](#)
[复制和粘贴对象](#)
[改变对象的重叠](#)
[对齐排列对象](#)
[旋转对象](#)
[镜像对象](#)
[定位对象](#)

选取对象

选取一个单个对象：

1. 在图形编辑器中，单击**选择**。
2. 单击相应的对象。出现了对象的大小控制手柄，并且当光标在对象上时它的形状由箭头形变为手形。

注意：如需同时选取更多的其它对象时，按住 **Ctrl** 键，然后顺序点击对象。如需从一个组合选取中取消某个已选对象时，仍就按住 **Ctrl** 键，然后再次点击对象。如需选取画面上所有对象时，从编辑菜单使用**全部选取**。如需取消所有已选对象时，点击对象以外的任何地方。

用大门框来选取一个对象组合：

1. 在图形编辑器中，单击**选择**。
2. 单击并按住鼠标左键拖动越过页面。这会创建一个临时的边界框。松开鼠标按钮就会选取框内的所有对象。
3. 松开鼠标键。

注意：如需同时选取更多的其它对象或放弃已选取的对象时，按住 **Ctrl** 键然后顺序点击对象。如需快速选取画面上所有对象，您可以使用编辑菜单中的**全部选取**。如需取消所有已选对象时，点击对象以外的任何地方。

另请参见

[操作对象](#)

[移动对象](#)

移动对象

可以通过单击对象并将其拖至新的位置来移动对象。

如果选择对象时即将其移动，则在页面中移动对象时将显示对象边界的轮廓。如果在移动对象之前按住鼠标 1 秒钟或更长时间，则会在移动对象时显示对象本身，这样可以更好地查看更精确放置的结果。

另请参见

[操作对象](#)

[改变对象大小](#)

改变对象大小

您可以轻松改变对象的大小。

改变对象大小：

1. 选择某个对象，然后将光标平移到大小控制手柄之上。光标将变为两面箭头，显示您可以拖动手柄以改变对象大小的方向。
2. 点击并拖动手柄至一个新位置。在您拖动时将会显示对象的边界框。
3. 松开鼠标键。

在对象的一角选择一个大小控制手柄，可以同时改变高度和宽度。在您移动一个角大小控制手柄时，如果您按住 **Ctrl** 键，对象将保持其比例关系（如，正方形依然保持为正方形）。

选择对象节点：

1. 选择相应的对象。节点的选择只适用于线条、管道、连线、或多边形对象。
2. 放置鼠标指针至节点上并按住。鼠标指针将会变成一个小的十字架形状。
3. 点击鼠标左键。选定的节点的颜色将改变，与背景色相反（暗背景时为亮色，亮背景时为暗色）。

如果您选择了一个节点，然后点击同一对象里的另外一个节点，将取消选择第一个节点。要选择多个节点，请按住 **Ctrl** 键，然后点击要选择的每个节点。（按住 **Ctrl** 键，您可以点击一个已经选定的节点以取消选择。）再次点击同一个节点，会在选择和取消选择间切换。如需取消选择所有的节点，点击节点以外的任何一个地方。

平移对象节点：

1. 选择节点。
2. 放置鼠标指针至节点上并按住。鼠标指针将会变成一个小的十字架形状。
3. 点击并按住鼠标左键。光标将变为放置符号。
4. 拖动选定的节点到所需的位置。
5. 松开鼠标键。

选择和平移多个节点将会在选定的节点间保持图形对象的比例。

添加节点至对象：

1. 选择相应的对象。
2. 放置鼠标指针至要添加新节点的确切点上的图形对象上并按住。鼠标指针将会变成一个指示手柄的形状。
3. 按 **Insert** 键，或任何一个可用的加号 (+) 键。

依您使用的键盘不同，加号键可能在数字键区，或可从主键盘通过 **Shift** 键进行访问。

从对象删除节点：

1. 选择节点。
2. 按 **删除** 或减号 (-) 键。

如果没有选择节点，按 **删除** 或减号键将删除对象。

另请参见

[操作对象](#)

[删除对象](#)

删除对象

可以删除多余的对象。

要删除某对象（或对象组），请执行以下操作：

1. 选择该对象（或对象组）
2. 选择 **编辑 | 删除**，或按 **删除** 键（或减号 (-) 键）。

另请参见

[操作对象](#)

[锁定/解锁对象](#)

锁定/解锁对象

在复杂画面上（对象很多），选取一个不包含所有对象的离散对象组（在选定区域里）可能会很困难（例如当一个对象隐藏在另外一个对象下面时）。为了避免无意中选取某个对象，可以将其锁定在一个位置。当一个对象被“锁定”后，它就不能被选取、删除、移动或编辑。对象只能在 **编辑** 菜单的 **解除锁定模式** 选项未被选中时才能锁定。

锁定对象：

1. 选定相应的对象。

2. 选择**编辑 | 锁定对象**。

解锁对象：

1. 选择**编辑 | 解除锁定模式**。
2. 选定相应的对象。
3. 选择**编辑 | 解锁对象**。

另请参见

[操作对象](#)

[将对象分组](#)

将对象分组

可以通过将对象分组使其更易于操作。

要将对象分组，请执行以下操作：

1. 点击**选择**。
2. 选择要分组的对象，然后点击**分组**（或选择**排列 | 分组对象**）。

要取消对象分组，请执行以下操作：

1. 点击**选择**。
2. 选择要分组的对象，然后点击**取消分组**（或选择**排列 | 取消对象分组**）。

要更改组的属性，请执行以下操作：

1. 点击**选择**。
2. 双击该组。将显示“属性”对话框。
3. 根据需要对属性进行更改。
4. 或者，选择**工具 | 转到对象**，选择该组，然后点击**确定**。

另请参见：

[操作对象](#)

[复制和粘贴对象](#)

复制和粘贴对象

你可以把对象复制并粘贴到其他的图形画面。

复制对象至剪贴板：

1. 单击**选取**。
2. 选取一个（或一组）对象。
3. 单击**复制**，或选择**编辑 | 复制**。

从剪贴板粘贴一个对象(或一组对象)：

- 单击**粘贴**，或选择**编辑 | 粘贴**。

剪切(删除)一个对象：

1. 单击**选取**。

2. 选取一个（或一组）对象。
3. 单击**剪切**，或选择**编辑 | 剪切**。

从剪贴板粘贴一个对象(或一组对象):

- 单击**粘贴**，或选择**编辑 | 粘贴**。

你可以用剪贴板在不同的图形画面间或在与其它图形应用程序间传送对象。缺省情况下，被粘贴位图的无效颜色将采用抖动方式进行处理。为了禁止颜色模糊扩散，可以在图形编辑器里从**工具菜单**里选择**选项**，然后除去**粘贴时模糊扩散处理位图**选项的钩。

取消你的最近的画图操作:

- 单击**撤销**，或选择**编辑 | 撤销**。

除了位图的编辑操作之外，可以取消当前绘图会话期间的**所有操作**。

取消撤销(或重做)你的最近的画图操作:

- 选择**编辑 | 重做**。

另请参见

[操作对象](#)

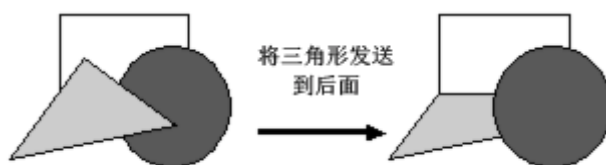
[改变对象的重叠](#)

改变对象的重叠

在图形编辑器中，对象可以重叠。在有对象重叠之处，新对象总是置于原有对象之上。可以在此显示顺序中前后移动对象来改变它们的重叠方式。

将一个（或一组）对象放置在其他对象之后以使所有对象都重叠在它（它们）之上:

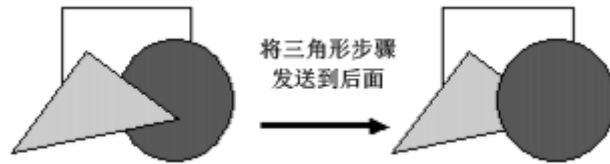
1. 单击**选择**。
2. 选取一个（或一组）对象。
3. 单击**发送至底层**或选择**排列 | 发送至底层**。



一个对象可以被一次向下和向上移动一层(而不一定非是一次到最下层，或一次到最上层)。

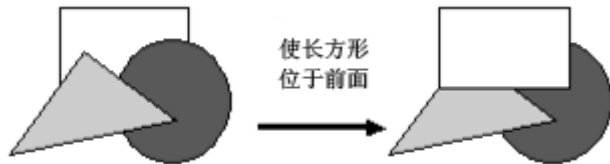
将一个(或一组)对象向下发送一步:

1. 单击**选择**。
2. 选取一个（或一组）对象。
3. 单击**后移一层**或选择**排列 | 后移一层**。



将对象提升至最上一层:

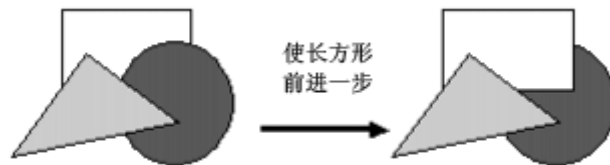
1. 单击**选择**工具。
2. 选取一个（或一组）对象。
3. 单击**提升至最上一层**或选择**排列 | 提升至最上一层**。



一个对象可以被一次向下和向上移动一层(而不一定非是一次到最下层，或一次到最上层).

将一个(或一组)对象向上提升一步:

1. 单击**选择**。
2. 选取一个（或一组）对象
3. 单击**前移一层**或选择**排列 | 前移一层**。



另请参见
[操作对象](#)
[对齐排列对象](#)

对齐排列对象

可以垂直地、平行地、或二者同时在两个方向上精确地对齐排列一组对象。

对齐排列对象:

1. 点击**选择**。
2. 选择相应的对象。
3. 选择**排列 | 对齐方式**。此时将出现“对齐”对话框。

选项	描述
----	----

选项	描述	
垂直	顶部对齐	使选定对象的顶边对齐
	居中	沿垂直方向对齐选定对象的中点
	底部对齐	使选定对象的底边对齐
	均等	沿垂直方向以均等的间距对齐选定对象的中点
	无	不更改选定对象的垂直对齐方式
水平	左	使选定对象的左边界对齐
	居中	沿水平方向对齐选定对象的中点
	右	使选定对象的右边界对齐
	均等	沿水平方向以均等的间距对齐选定对象的中点
	无	不更改选定对象的水平对齐方式

另请参见
[操作对象](#)
[旋转对象](#)

旋转对象

可以将对象向右旋转 90°（顺时针）或向左旋转 90°（逆时针）。

如何旋转对象（或对象组）：

1. 单击**选择**。
2. 选择对象。
3. 选择**排列 | 旋转**。

如何旋转文本对象：

1. 单击**选择**。
2. 选择对象。
3. 选择**工具 | 转换为位图**。
4. 选择**排列 | 旋转**。
5. 旋转对象（或对象组）的旋转方向。

对象将按所选的方向旋转 90 度。要将对象旋转 180 度，请单击方向按钮两次。

另请参见
[镜像对象](#)
[操作对象](#)

镜像对象

可以相对于水平轴或垂直轴镜像对象。

如何相对于水平轴或垂直轴镜像对象（或对象组）：

1. 单击**选择**。
2. 选择对象
3. 选择**排列 | 镜像**。
4. 选择镜像对象（或对象组）的轴。

另请参见

[定位对象](#)

[操作对象](#)

定位对象

使用“定位对象”对话框可以查找、选择和访问当前画面上或当前画面使用的画面模板上对象的属性。可以选择画面（或模板）上的对象、精灵、符号和组，以及组成这些精灵、符号和组的图形元素。

定位当前画面上的对象(或组、精灵、符号或画面模板)并显示其属性：

1. 选择**工具 | 定位对象**。
2. 在树状结构中定位对象（或组、精灵、符号或画面模板），或者在**对象 AN**框中键入相关 AN。
包含多个对象（或其他图形元素）的对象旁边有一个加号(+)。点击+号可以看见这些组成对象。
3. 双击树状结构中的对象或单击**确定**可显示其属性。

Chapter: 20 了解对象类型

有多种不同的对象类型，每种都有其独特的属性系列。有关常用于所有对象类型的属性，请参见 [定义通用对象属性](#)。

另请参见

[自由手画线对象](#)

[直线对象](#)

[长方形对象](#)

[椭圆对象](#)

[多边形对象](#)

[管道对象](#)

[文本对象](#)

[数字对象](#)

[按钮对象](#)

[符号集对象](#)

[趋势对象](#)

[Cicode 对象](#)

[粘贴的符号对象](#)

[粘贴的精灵对象](#)

[ActiveX 对象](#)

[过程分析器用户向导中的“使用过程分析器”](#)

[数据库交换控制对象](#)

[Vijeo 网关控制对象](#)

手画线对象

使用手画线工具，可以绘制线条。可以对线条执行移动、改变大小、改变形状、提升至最上层等操作，还可以像其他类型的对象一样编辑其属性。

要绘制手画线，请执行下列操作：

1. 点击自由手工具。



2. 将光标移动到开始画线的位置。
3. 点击并拖拽光标绘制线条。
释放鼠标键时，会显示线条的对象属性对话框。

另请参见

[手画线属性 - 外观（一般）](#)

[了解对象类型](#)

手画线属性 - 外观（一般）

手画线图具有下列一般的外观属性：

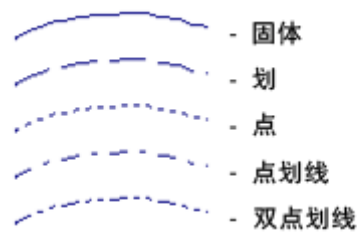
[线条] 宽度

线条的粗细（以像素为单位）。通过点击字段右边的向上和向下的箭头或在字段中直接输入另外一个数值，可以改变线条的粗细。

如果使线条的粗细比 1 个像素宽，那么此线条必须为实线。

[线条] 风格

线条的风格。可以选择下列一种线条风格：



要更改风格，请从该字段右侧的菜单中选择风格。

[线条] 颜色

线条的颜色。

[填充] 填充

填充复选框决定对象是否用颜色填充。如果选中了该框，就会有一条不可见的线连接您的线条的两端。位于这条不可见线和您的线条之间的所有对象均会被填充。



[填充] 颜色

对象要填充的颜色。您选取的作为填充用的颜色在这里是静态的。

要指定随 Runtime 条件变化的填充颜色，请点击**填充**分页。

如果您已启用了填充（颜色）属性，请注意此处选择的颜色将覆盖“填充颜色”（开/关）的“关”颜色、“填充颜色”（多状态）的 ABC 颜色、“填充颜色”（阵列）的“阵列颜色 0”，以及“填充颜色”（梯度）的“最小处”颜色。

要获取剩余属性分页的帮助，请参见 [定义通用对象属性](#)。

直线对象

可以使用直线工具绘制直线。直线可以移动、调整大小、改变形状、提升至最上层等，可以采取与其它类型对象完全相同的方式编辑它们的属性。

要绘制直线：

1. 点击**直线**工具。



2. 将光标移动到要开始画线的位置。
3. 点击并拖动以绘制线条。（如果在绘制线条时按住 **Ctrl** 键，将只能绘制垂直或水平线条。

释放鼠标键时，会显示线条的对象属性对话框。

另请参见

[直线属性 - 外观（通用）](#)
[了解对象类型](#)

直线属性 - 外观（通用）

直线具有以下通用外观属性。

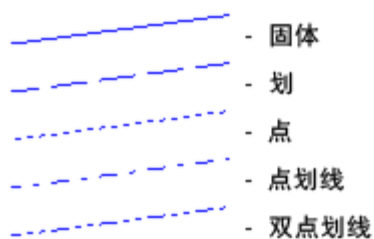
[线条] 宽度

线条的宽度（以像素为单位）。可以通过点击字段右边的向上和向下箭头或在字段中直接输入另外一个参数值来改变线条的粗细。

如果使线条的宽度大于 1 像素，那么此线条必须为实线。

[线条] 风格

线条的风格。可以选择下列任意线条风格：



要更改风格，请从该字段右侧的菜单中选择风格。

[线条] 颜色

线条的颜色。

要获取有关其余属性分页的帮助，请参见 [定义通用对象属性](#)。

长方形对象

使用“长方形”工具，可以绘制长方形和正方形。长方形和正方形可以移动、调整大小、改变形状、提升至最上层等，还可以像其它类型对象一样编辑其属性。

如何画一个长方形：

1. 点击**长方形**工具。



2. 将光标移动到开始绘制长方形的位罝。
3. 点击并拖动鼠标至长方形结束的点，然后放开鼠标键。如果您按住 **Shift** 键，然后开始画长方形，将从中心开始向外画。
释放鼠标键时，会显示长方形的对象属性对话框。

如何画一个正方形：

1. 点击**长方形**工具。
2. 点击（并按住）**Ctrl** 键。
3. 移动鼠标至想要正方形起始的地方，然后点击并按住鼠标键。
4. 拖动鼠标至正方形的结束点，然后放开鼠标键。如果您按住 **Shift** 键和 **Ctrl** 键，然后开始绘制正方形，将从中心开始向外画。
释放鼠标键时，会显示正方形的对象属性对话框。

另请参见

[长方形属性 - 外观（通用）](#)
[了解对象类型](#)

长方形属性 - 外观（通用）

长方形有以下总的外观属性：

[线条] 宽度

长方形边线的粗细（单位为像素）。通过点击字段右边的向上和向下箭头，或在字段中直接输入另外一个参数值，可以改变线条的粗细。

如果使线条的粗细比 1 个像素宽，那么此线条为实线。

[线条] 风格

长方形的边线的风格。可以选择下列任意线条风格：



要更改风格，请从该字段右侧的菜单中选择风格。

[线条] 颜色

长方形的边线的颜色。

[填充] 填充

此填充选项决定了长方形是否会被填充进一种颜色。

[填充] 颜色

用于对长方形进行填充的颜色。你选取的作为填充用的颜色在这里是静态的。

要指定填充颜色随运行时的情况而改变，可点击“填充”分页。如果您已启用了填充（颜色）属性，请注意此处选择的颜色将覆盖“填充颜色”（开/关）的“关”颜色、“填充颜色”（多状态）的 ABC 颜色、“填充颜色”（阵列）的“阵列颜色 0”，以及“填充颜色”（梯度）的“最小处”颜色。

[对象类型] 附加线条

如果长方形被定义成“凸起”或“凹进”（点击“3D 效果”分页标签），此选项将为长方形添加一条附加的低光颜色的线条（1 个象素宽）。






[梯度] 颜色

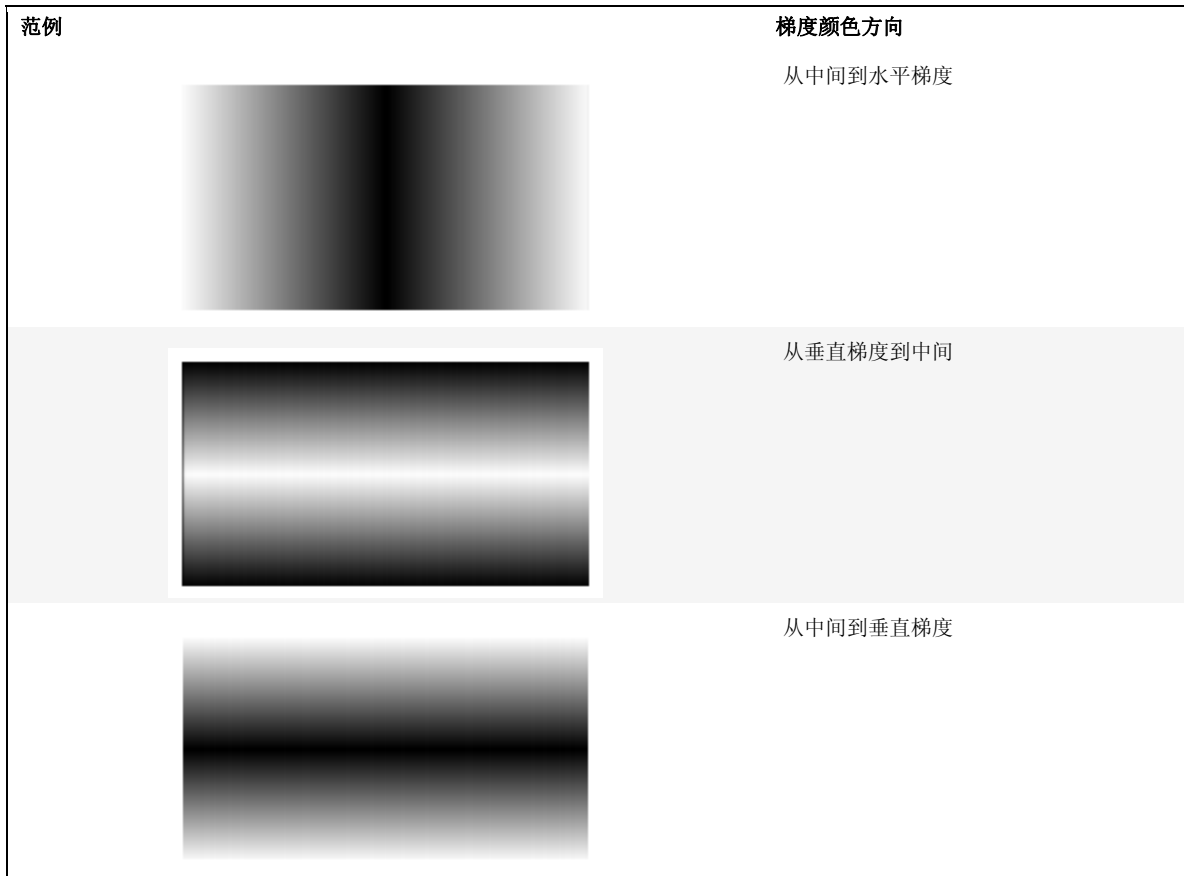
控制填充颜色和梯度颜色间的梯度填充的颜色。仅当选中**填充**和**梯度填充**选项时，此选项才可用。梯度在运行时得到更新，以反映两个所选颜色间的梯度。梯度填充支持闪烁颜色。

梯度对于对象不会旋转，例如，如果某个对象包含一个从左到右的梯度填充，旋转 90 度后（在运行时或在图形编辑器中），梯度仍为从左到右。

[梯度] 方向

用于梯度颜色的方向。可以将下表作为向导，从而选择需要的梯度颜色方向。

范例	梯度颜色方向
	从左到右
	从右到左
	从上到下
	从下到上
	从水平梯度到中间

**[对象类型] 边界**

为长方形的边线添加一条附加的线条（1 个像素宽）以使其变粗。

[对象类型] 角半径

控制长方形的角半径。输入 0 到 32 之间的参数值。参数值越大，长方形的角越圆。

如果半径大于 0，则**附加线条**和**边界**选项不可用。

要获取其余属性分页的帮助，请参见 [定义通用对象属性](#)。

椭圆对象

您可以使用“椭圆”工具来绘制椭圆、圆、弧线和扇形。可以对椭圆对象执行移动、改变大小、改变形状、提升至最上层等操作，还可以像其他类型的对象一样编辑其属性。

如何画椭圆：

1. 点击**椭圆**工具。



2. 移动鼠标光标至边界长方形（大方框）的一角，然后点击（并按住）鼠标键。
3. 拖动鼠标光标至边界长方形相对的一角，然后放开鼠标键。如果您按住 **Shift** 键，然后开始画椭圆，将从圆心开始向外画。
释放鼠标键时，会显示椭圆的对象属性对话框。

如何画圆：

1. 点击**椭圆**工具。
2. 点击（并按住）**Ctrl** 键。
3. 移动鼠标光标至边界长方形（大方框）的一角，然后点击（并按住）鼠标键。
4. 拖动鼠标光标至边界长方形相对的一角，然后放开鼠标键。如果您按住 **Shift** 键和 **Ctrl** 键，然后开始画圆，将从圆心开始向外画。
释放鼠标键时，会显示圆的对象属性对话框。

另请参见

[椭圆属性 - 外观\(通用\)](#)

[了解对象类型](#)

椭圆属性 - 外观(通用)

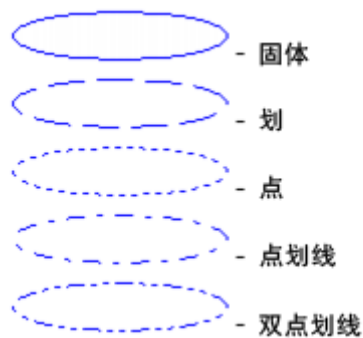
椭圆有以下总的外观属性：

[线条] 宽度

椭圆边线的粗细（单位为像素）。通过点击字段右边的向上和向下箭头或直接在字段中输入另外一个数值，可以更改其粗细。如果线条的粗细大于 1 个像素，那么此线条为实线。

[线条] 风格

椭圆边线的风格。您可以选择以下任一线条风格：



要更改风格，可以从此框右边的菜单中选择一种风格。

[线条] 颜色

椭圆边线的颜色。

[填充] 填充

“填充”选项决定了是否会使用一种颜色对椭圆进行填充。

[填充] 颜色

用于对椭圆进行填充的颜色。您选择作为填充用的颜色是静态的。

要指定填充颜色随运行时的情况而改变，可点击“填充”分页。如果您已启用了填充（颜色）属性，请注意此处选择的颜色将覆盖“填充颜色”（开/关）的“关”颜色、“填充颜色”（多状态）的 ABC 颜色、“填充颜色”（阵列）的“阵列颜色 0”，以及“填充颜色”（梯度）的“最小处”颜色。

[梯度] 颜色



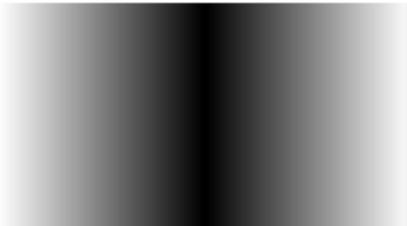
控制填充颜色和梯度颜色间的梯度填充的颜色。仅当选中的**填充**和**梯度填充**选项时，此选项才可用。梯度在运行时得到更新，以反映两个所选颜色间的梯度。梯度填充支持闪烁颜色。

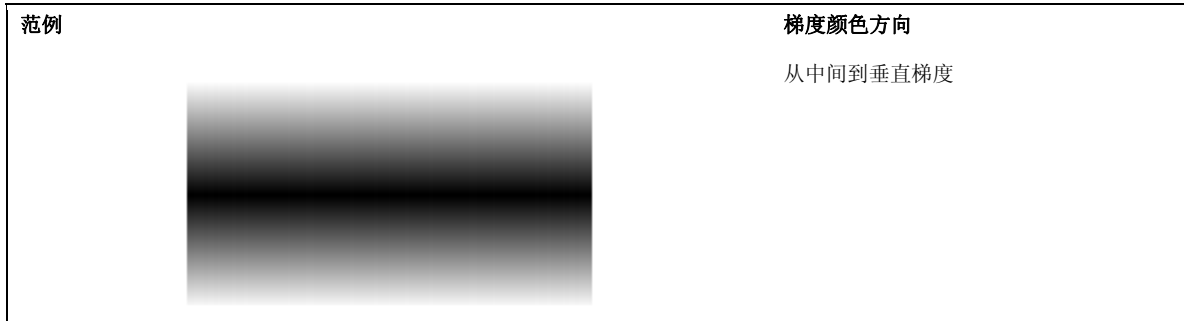
梯度对于对象不会旋转，例如，如果某个对象包含一个从左到右的梯度填充，旋转 90 度后（在运行时或在图形编辑器中），梯度仍为从左到右。

[梯度] 方向

用于梯度颜色的方向。可以将下表作为向导，从而选择需要的梯度颜色方向。



范例	梯度颜色方向
	从上到下
	从下到上
	从水平梯度到中间
	从中间到水平梯度
	从垂直梯度到中间



[对象类型] 椭圆

如果您希望此对象是一个完整的椭圆，请选择此选项。



对于一个完整的椭圆，您无需指定起始和结束的角度。

[对象类型] 扇形

如果您希望从椭圆里去掉一部分（也就是说您希望得到一个扇形），请选择此选项。

如果选择此选项，您就可以指定一个起始角度和一个结束角度：

起始角度

要从椭圆中去掉部分的角度（从 0 开始顺时针计算）。例如，如果您输入一个起始角度 50 度，您将得到一个扇形，如图：



结束角度

要在椭圆中保留部分的角度（从 0 开始顺时针计算）。例如，如果输入一个结束角度 150 度，您将得到一个扇形，如图：



起始和结束角度能够结合在一起使用，以带来不同的效果。例如，一个起始角度 270 度，和一个结束角度 150 度将会产生一个扇形，如图：



[对象类型] 弧线

如果您想画弧线，请选择此选项。

如果选择此选项，您就可以指定一个起始角度和一个结束角度：

起始角度

起始角度定义的是要从椭圆中去掉部分的角度（从 0 开始顺时针计算），并剩下部分的弧线。例如，如果输入一个起始角度 50 度，您将得到一条弧线，如图：



结束角度

结束角度定义的是要在椭圆中保留部分的角度（从 0 开始顺时针计算）。例如，如果输入一个结束角度 150 度，您将得到一个扇形，如图：



起始和结束角度能够结合在一起使用，以带来不同的效果。例如，一个起始角度 270 度，和一个结束角度 150 度将会产生一条弧线，如图：



要获取其他属性的帮助，请参见 [定义通用对象属性](#)。

多边形对象

使用多边形工具可绘制多边形和多边连线。可以对多边形执行移动、改变大小、改变形状、提升至最上层等操作，还可以像其他类型的对象一样编辑其属性。

如何绘制一个多边形：

1. 单击**多边形**工具。



2. 将光标移到要开始绘制多边形的位置，然后单击并按住鼠标键。
3. 在第一条线的结尾处，放开鼠标键。
4. 依次将光标移到多边形上的每个点并点击鼠标键（在绘制第一部分后无需点击和拖动）。

如只需水平或垂直绘制多边形，请在按住 **Ctrl** 键的同时绘制多边形。

5. 如需完成多边形的绘制，请双击鼠标键。
多边形绘制完成时，会显示对象属性对话框。

如何画一个多边连线：

1. 单击**多边形**工具。
2. 将光标移到要开始绘制多边连线的位置，然后单击并按住鼠标键。
3. 在第一条线的结尾处，放开鼠标键。
4. 依次将光标移到多边连线上的每个折点并点击鼠标键（在绘制第一部分后无需点击和拖动）。
如只需水平或垂直绘制，请在按住 **Ctrl** 键的同时绘制多边连线。
5. 要完成多边连线的绘制，请双击鼠标键。
释放鼠标键时，会显示圆的对象属性对话框。
刚开始时，对象将会是一个多边形。要将其改变成一条多边连线，请在对象的属性对象框中将其定义为[对象类型] 打开。

另请参见

[多边形属性 - 外观（外观）](#)
[了解对象类型](#)

多边形属性 - 外观（通用）

多边形有以下总的外观属性：

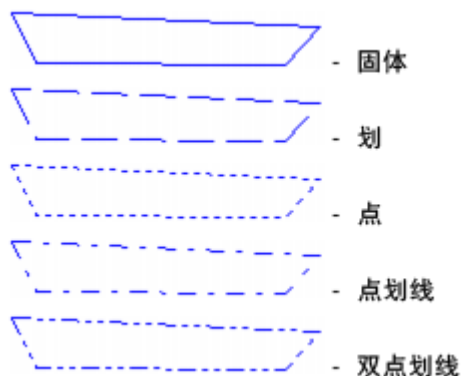
[线条] 宽度

多边形线条的粗细 (单位为像素)。通过点击字段右边的向上和向下箭头或在字段中直接输入另外一个数值，可以改变线条的粗细。

如果要使线条的粗细比 1 个像素宽，那么它必须是实线。

[线条] 风格

多边形线条的风格。可以选择下列任何一种线条风格：



要更改风格，请从该字段右侧的菜单中选择所需的风格。

[线条] 颜色

多边形边线的颜色。

[填充] 填充

此填充复选框决定多边形是否会被填充上一种颜色。

[填充] 颜色

用于对多边形进行填充的颜色。选择作为填充用的颜色在这里是静态的。

要指定填充颜色随运行时情况而改变，可点击“填充”分页。如果您已启用了填充（颜色）属性，请注意此处选择的颜色将覆盖“填充颜色”（开/关）的“关”颜色、“填充颜色”（多状态）的 ABC 颜色、“填充颜色”（阵列）的“阵列颜色 0”，以及“填充颜色”（梯度）的“最小处”颜色。



[梯度] 颜色





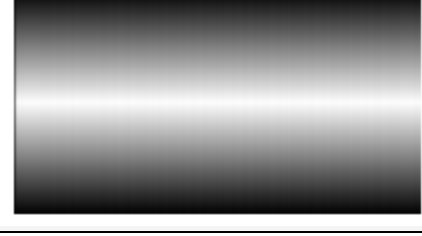
控制填充颜色和梯度颜色间的梯度填充的颜色。仅当选中**填充**和**梯度填充**复选框时，此选项才可用。梯度在运行时会被更新，以反映两个选定颜色间的梯度。梯度填充支持闪烁颜色。

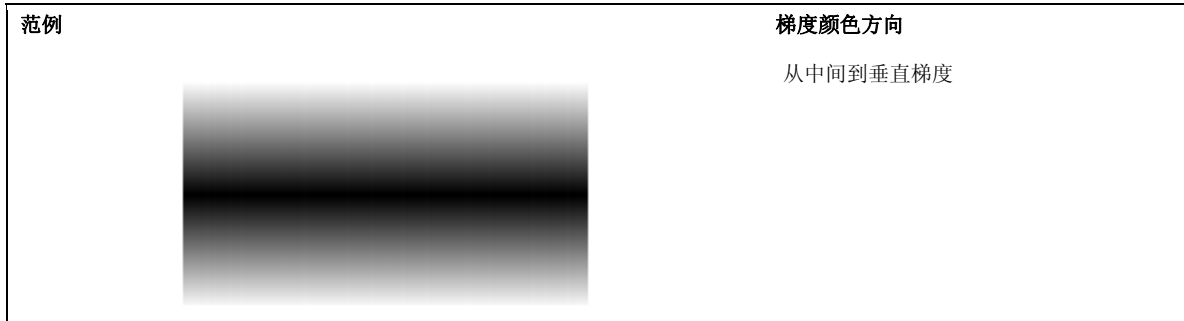
梯度对于对象不会旋转，例如，如果某个对象包含一个从左到右的梯度填充，旋转 90 度后（在运行时或在图形编辑器中），梯度仍为从左到右。

[梯度] 方向

用于梯度颜色的方向。可以将下表作为向导，从而选择需要的梯度颜色方向。

范例	梯度颜色方向
	从左到右
	从右到左

范例	梯度颜色方向
	从上到下
	从下到上
	从水平梯度到中间
	从中间到水平梯度
	从垂直梯度到中间



[对象类型] 开放

将对象定义成一个多边连线(第一点与最后一点没有连接起来)。

[对象类型] 闭合

将对象定义成一个多边形 (第一点与最后一点连接起来)。

要获取其余属性分页的帮助, 请参见 [定义通用对象属性](#)。

管道对象

使用管道工具可绘制具有自动三维阴影的管道。可以对管道执行移动、改变大小、改变形状、提升至最上层等操作, 还可以像其他类型的对象一样编辑其属性。

如何绘制一个管道:

1. 点击管道工具。



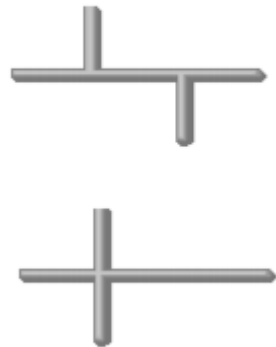
2. 将光标移到要开始绘制管道的位置, 然后单击并按住鼠标键。
3. 在第一条线的结尾处, 放开鼠标键。
4. 依次将光标移到路径上的每个点并单击鼠标键 (在绘制好第一段后不需要点击和拖动)。
5. 要完成管道的绘制, 请双击鼠标键。
管道绘制完成时, 会显示对象属性对话框。

提示: 如只需水平或垂直绘制管道, 请按住 **Ctrl** 键绘制管道。

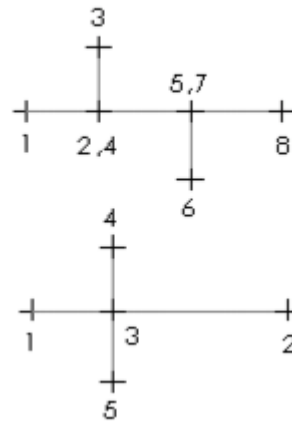
绘制复杂管道排列

使用管道工具, 可以绘制复杂的管道排列 (包括'T'形点和交叉点)。如下图所示的管道, 以及画它们所需的鼠标点击顺序:

要绘制此管道...



...跟随此鼠标点击次序



提示： 使用坐标线以在每次点击时帮助精确定位。

另请参见
[管道属性 - 外观 \(通用\)](#)
[了解对象类型](#)

管道属性 - 外观 (通用)

管道具有下列通用外观属性：

[线条] 宽度

管道的粗细（单位为像素）。通过点击字段右边的向上和向下箭头或在字段中直接输入另外一个数值，可以改变线条的粗细。所有管道必须至少有 1 个像素宽。

[线条] 高光颜色

管道在光亮处的颜色；即管道上最明亮的颜色。

[线条] 低光颜色

管道阴暗处的颜色；即管道上最暗的颜色。

要获取其余属性分页的帮助，请参见 [定义通用对象属性](#)。

文本对象

可使用文本工具在页面上键入文本。可以对文本执行移动、改变大小、改变形状、提升至最上层等操作，还可以像其它类型的对象一样编辑其属性。

要添加文本：

1. 单击**文本**工具。



2. 用键盘键入您的文本。（按 **Enter** 换行。）
3. 将光标移动至要放置文本的地方，然后点击鼠标左键。
点击鼠标键放置文本时，会显示对象属性对话框。

另请参见

[文本属性 - 外观（通用）](#)

[了解对象类型](#)

文本属性 - 外观（通用）

文本有以下通用外观属性

字体

用于显示文本的字体。使用滚动条向右滚动可查看可用字体，也可以向此域中直接键入部分或完整的字体名称。

风格

选择希望文本使用的风格：常规、**粗体**、**粗斜体**还是*斜体*。

大小

定义文本的大小（磅值）。可用大小可能会因所选打印机和所选字体的不同而有所不同。

[排列] 左对齐

选择此单选按钮可使文本在文本框中左对齐。

[排列] 右对齐

选择此单选按钮可使文本在文本框中右对齐。

[排列] 居中

选择此单选按钮可使文本在文本框中居中对齐。

[效果] 删除线

选中此框可使文本显示时带有一条线贯穿它的线。

[效果] 下划线

选中此框将使文本带下划线。

文本

此字段包含将显示在画面上的文本。可以输入任何键盘字符。可以在此处编辑该文本，也可以从图形画面上直接编辑。在此字段编辑文本非常有用，因为可以在应用其他字体和颜色修改的同时应用文本修改。

此文本会根据您定义的显示值属性自动变化。

前景色

文本的颜色。

注意：显示数值中有数个单选按钮（开/关、多状态等）。当选择不同的选项时，对话框右边部分的外观将随之变化。

另请参见

[文本属性 - 外观（显示数值）](#)

文本属性 - 外观（显示数值）

文本具有以下显示数值外观属性。选择以下五个选项之一可更改对话框左侧和右侧的外观。

[类型] 开/关

更改在某特定条件满足时显示的文本，以及条件未满足时显示的其它文本。例如，您可以在一个特定的变量标签报警时显示一条报警信息，而当标签没有报警时显示正常的信息。

参见 [文本属性 - 外观显示数值（开/关）](#)。

[类型] 多状态

如果在不同的时间，有多个条件以不同的组合发生，可以使用此选项。选择此选项以为每个组合显示不同的文本。

例如，三个数字量变量标签（A、B 和 C）在任意时间都可能是“开”或“关”状态。你可以为每个开/关组合显示不同的信息。换言之，您可以针对以下每个开/关组合显示不同的消息：**ABC、ABC、ABC、ABC、ABC、ABC、ABC、ABC。**

参见 [文本属性 - 外观显示数值（多状态）](#)。

[类型] 阵列

允许您输入一个返回整数的表达式。可以为每一个整数（从 0 到 255）显示不同的文本。例如，您能够为一个模拟量标签的每个状态显示一个不同的信息。

参见 [文本属性 - 外观显示数值（阵列）](#)。

[类型] 数值

用数值格式（可以指定此格式）来显示标签或表达式的值。

参见 [文本属性 - 外观显示数值（数值）](#)。

[类型] 字符串

以字符串的形式显示表达式的值。

参见 [文本属性 - 外观显示数值（字符串）](#)。

文本属性 - 外观显示数值（开/关）

文本具有以下“开/关”显示数值属性：

开文本显示时间

在**开文本域**（下面）中输入的文本将在此处输入的条件满足时显示。文本的最大长度为 128 个字符。

如需插入标签或函数，点击此域右侧的**向导**按钮。此按钮显示两个选项，**插入标签**和**插入函数**。

关文本

在以上条件不满足时而显示的文本。可以使用任何键盘字符，最多可以使用 256 个字符。

例如，可以在 **CV110_FAULT.On** 为 false 时（即传送带 110 上没有报警时）显示消息 **Conveyor 110 Normal**。

开文本

在以上条件满足时显示的文本。可以输入任何键盘字符，最多可以输入 256 个字符。

例如，可以在 **CV110_FAULT.On** 为 true 时（即传送带 110 上没有报警时）显示消息 **Conveyor 110 Alarm**。

点击**清除属性**可清除属性详细信息并禁止属性。

另请参见

[文本属性 - 外观（显示数值）](#)

[文本属性 - 外观显示数值（多状态）](#)

[文本属性 - 外观显示数值（阵列）](#)

[文本属性 - 外观显示数值（数值）](#)

[文本属性 - 外观显示数值（字符串）](#)

文本属性 - 外观显示数值（多状态）

文本具有以下多状态显示数值属性：

条件

在此处输入的条件可能在不同时间以不同方式一起发生。可以用每个不同的组合来决定显示的文本。

默认的条件个数是 3，但还可以使用**添加**按钮来添加更多的条件（直到最多 5 个条件 – 可以提供 32 个组合）。您也可以使用**删除**按钮来删除条件，但是您必须始终在此域中输入条件。如需输入条件，点击相关行（A、B、C 等），然后点击**编辑**。如需插入标签或函数，点击**向导**按钮。此按钮显示两个选项：插入标签和插入函数。

状态文本

以上条件的每一种组合时所显示的文本。可以输入任何键盘字符。

例如：

要显示有关阀门状态的不同消息，可以按以下方式填写**条件**和**状态文本**域：

条件	状态文本
A 打开反馈	ABC 阀门不正常
B 关闭反馈	ABC 阀门不正常
C 打开输出	ABC 阀门已关闭
	ABC 阀门不正常
	ABC 阀门不正常
	ABC 阀门打开
	ABC 阀门不正常
	ABC 阀门不正常

在此例子中，**Open_Feedback** 和 **Close_Feedback** 是代表阀门数字量输入的变量标签，而 **Open_Output** 是代表阀门输出的变量标签。因此，ABC 表示 **Open_Feedback** 开启，**Close_Feedback** 和 **Open_Output** 均关闭。针对此组合，文本“阀门不正常”将显示，因为阀门在它应该关闭时打开了。相同的逻辑类型也同样适用于其它状态。

点击**清除属性**可清除属性详细信息并禁止属性。

另请参见

[文本属性 - 外观（显示数值）](#)

[文本属性 - 外观显示数值（开/关）](#)

[文本属性 - 外观显示数值（阵列）](#)

[文本属性 - 外观显示数值（数值）](#)

[文本属性 - 外观显示数值\(字符串\)](#)

文本属性 - 外观显示数值（阵列）

文本具有以下阵列显示数值属性：

阵列的表达式

输入将返回一个或多个整数的表达式。针对每个返回的整数，将显示一段不同的文本。

如果返回参数值是：

- 小于 0（零），将设置为 0（零），并将触发运行时硬件报警。
- 大于 255，将设置为 255，并将触发运行时硬件报警。
- 实数（非整数），将被四舍五入到最接近的整数。

如需插入标签或函数，点击此域右侧的**向导**按钮。此按钮显示两个选项：插入标签和插入函数。

阵列文本

为以上输入的阵列表达式返回的每个整数所显示的文本。可以输入任何键盘字符。

例如，要显示有关马达状态的不同消息，您可以按以下方式填写**阵列的表达式**和**阵列文本**：

阵列的表达式	阵列文本	
MOTOR_STATUS	0	正在运行
	1	正在启动
	2	正在停止
	3	已停止
	4	已出现故障
	5	已被孤立

在此例子中，MOTOR_STATUS 是一个表示马达状态的模拟量变量标签。每次马达改变状态时，返回一个整数（0 = 正在运行，1 = 正在启动等），并显示相应的文本。

点击**清除属性**可清除属性详细信息并禁止属性。

另请参见

[文本属性 - 外观（显示数值）](#)

[文本属性 - 外观显示数值（开/关）](#)

[文本属性 - 外观显示数值（多状态）](#)

[文本属性 - 外观显示数值（数值）](#)

[文本属性 - 外观显示数值（字符串）](#)

文本属性 - 外观显示数值（数值）

文本有以下数值显示值属性。

数值表达式

在此处输入的表达式的值将显示在图形画面上。显示时使用的格式将用以下选定格式。

如需插入标签或函数，点击此域右侧的**向导**按钮。此按钮显示两个选项：插入标签和插入函数。

格式

在以上填写的表达式将会以此处填写的格式来显示。例如，模拟变量标签 **MOTOR_STATUS** 返回整数 0-5。如果您输入此标签作为上述数值表达式，并输入 **###** 作为您的格式，将交替显示 **0.00**、**1.00**、**2.00**、**3.00**、**4.00** 和 **5.00**。可以从下拉菜单里选择一种格式，或键入您自己的格式。如果数值表达式是一个单独的变量，其格式将被您在此处输入的格式覆盖。

点击**清除属性**可清除属性详细信息并禁止属性。

另请参见

[文本属性 - 外观（显示数值）](#)

[文本属性 - 外观显示数值（开/关）](#)

[文本属性 - 外观显示数值（多状态）](#)

[文本属性 - 外观显示数值（阵列）](#)

[文本属性 - 外观显示数值（字符串）](#)

文本属性 - 外观显示数值（字符串）

文本具有以下字符串显示数值属性。

字符串表达式

此处输入的表达式将以字符串的形式显示在图形页面上。

如需插入标签或函数，点击此域右侧的**向导**按钮。此按钮显示两个选项：插入标签和插入函数。

单击**清除属性**可清除属性的详细内容，然后禁止属性。要定义该对象的其它属性，请点击相关分页。

要获取有关其余属性分页的帮助，请参见 [定义通用对象属性](#)。

另请参见

[文本属性 - 外观（显示数值）](#)

[文本属性 - 外观显示数值（开/关）](#)

[文本属性 - 外观显示数值（多状态）](#)

[文本属性 - 外观显示数值（阵列）](#)

[文本属性 - 外观显示数值（数值）](#)

数字对象

使用“数字”工具，您可以使用数字来表示标签或表达式。当您在画面中放置一个数字时，您所需要做的只是输入相关的变量标签或表达式。可以对数字进行移动、调整大小、提升至最上层等操作，还可以像其他对象类型一样编辑其属性。

（“文本”工具也具有同样的功能。）

如何在图形画面中添加数字：

1. 点击**数字**工具。



2. 将光标移动到您想要显示此数字的位置，然后点击鼠标键。出现“文本属性”对话框，您可以在其中输入相关变量标签或表达式。

有关各“属性”分页的帮助，请参见 [文本属性 - 外观（通用）](#)和 [定义通用对象属性](#)。

按钮对象

可使用按钮工具在图形画面上绘制按钮。然后可以为其指定安全权限和命令。

可以对按钮进行移动、调整大小、改变形状、提升至最上层等操作，还可以像其他类型一样编辑其属性。

绘制按钮：

1. 单击**按钮**工具。



2. 将鼠标移到要开始绘制按钮的位置并按（住）鼠标键。
3. 将鼠标拖到要结束按钮的位置并释放鼠标键。
释放鼠标键时，会显示对象属性对话框。

另请参见

[按钮属性 - 外观\(通用\)](#)

[了解对象类型](#)

按钮属性 - 外观(通用)

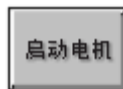
按钮具有下列通用外观属性。

[类型] 文字

选择此选项可在按钮上显示文字。如果选择此选项，**文本**和**字体**字段将显示在对话框的右边。

如果选择**文本**或**符号**类型选项，则可以使用 XP 风格的按钮。要将按钮配置为使用 Windows XP 风格，请选中 **XP 风格**复选框。在运行期间，XP 风格的按钮在使用键盘控制时具有蓝色边框，而当鼠标停留在它上面时具有橙色边框。将按钮放置在画面上时，缺省选中 **XP 风格**复选框。

注意：按 ^n 或 Enter 键可将文本另起一行。例如，Start^nMotor 将显示为：



字体

用于显示文本的字体。使用滚动条向右滚动可查看可用字体，也可以在此字段中直接键入部分或完整的字体名称。

注意：对按钮字体不再支持背景颜色属性。从以前版本导入的任何按钮的背景颜色均将设置为缺省颜色。

风格

选择文本风格是常规、加粗、斜体加粗还是斜体。

大小

定义文本的大小(磅值)。可用大小可能会因所选打印机和所选字体的不同而变化。

自定义填充颜色

选中此复选框以启用“填充颜色上”和“填充颜色下”复选框。

填充颜色上

处于“上”状态时的按钮颜色。

填充颜色下

处于“下”状态时的按钮颜色。

注意：“填充颜色上”和“填充颜色下”选项仅对文本和符号类型才可用。

[排列] 左对齐

选中此单选按钮可左对齐多行文本。对齐的文本将保持在按钮的中心。这对单行文本无效。

[排列] 右对齐

选中此单选按钮可右对齐多行文本。对齐的文本将保持在按钮的中心。这对单行文本无效。

[排列] 居中

选中此单选按钮可居中对齐多行文本。对齐的文本将保持在按钮的中心。这对单行文本无效。

[效果] 删除线

选中此框可使文本显示为有一条线贯穿。

[效果] 下划线

选中此框将使文本带下划线。

文本

此字段包含将显示在画面上的文本。可以输入任何键盘字符。可以在此处编辑该文本，也可以从图形画面上直接编辑。在此字段编辑文本非常有用，因为可以在应用其他字体和颜色修改的同时应用文本修改。

此文本会根据您定义的显示值属性自动变化。

前景色

文本的颜色。

[类型] 符号

选择此选项可在按钮上显示符号。如果选择此选项，则**设置**按钮会显示在对话框右边。单击**设置**可选择要显示在按钮上的符号。同时还会显示所选符号的图形。

[类型] 目标

选择此选项时，按钮上不会显示任何文本或符号，会有一个透明的外表。

模式

透明按钮有三种不同的模式：

- **BORDER_3D**：按钮上只有 3-D 边线框（透明外表）。
- **BORDER**：按钮上只有一个细线边框。
- **TARGET**：按钮完全透明。这构成一个屏幕目标。

要获取剩余属性分页的帮助，请参见 [定义通用对象属性](#)。

符号集对象

符号集工具允许您用不断变化的符号来表示不断变化的 **Runtime** 条件。通过点击此工具，然后点击图形画面，您就能够定义在每种条件下显示的符号。

在一个符号系列被输入并放置后，它能够被移动、改变大小、改变形状、提升至最上层等，可以采用与任何其它类型对象相同的方式修改其属性。

要添加符号集：

1. 点击**符号**工具。



2. 移动鼠标光标至你想要在画面上放置的地方，然后用鼠标左键点击。
点击鼠标键时，会显示对象属性对话框。
3. 填写符号集的相关属性，然后点击**确定**。

注意：符号集外表有数个单选按钮（相关描述见下文）。当选择不同的选项时，对话框右边部分的外观将随之变化。

[类型] 开/关

选取此选项以当某个特定的表达式为 **TRUE** 时显示某个符号，而当另外一个表达式为 **FALSE** 时显示另外一个符号。例如，您可以在一个特定的变量标签报警时显示一个红色的符号，而当标签没有报警时显示绿色的符号。

参见 [符号集属性 - 外观通用（开/关）](#)。

[类型] 多状态

如果在不同的时间，有多个条件以不同的组合发生，可以使用此选项。选择此选项以为每个组合显示不同的符号。

例如，三个数字量变量标签（A、B 和 C）在任意时间都可能是“开”或“关”状态。你可以为每个开/关组合显示不同的符号。也就是说，您可以针对每个开/关 ABC 组合显示不同的符号：**ABC**、**ABC**、**ABC**、**ABC**、**ABC**、**ABC**、**ABC**。

参见 [符号集属性 - 外观通用（多状态）](#)。

[类型] 阵列

阵列选项允许您输入一个返回整数的表达式。可以针对每个唯一的整数（从 0 到 255）显示一个唯一的符号。例如，您可以为一个模拟量标签的每个状态显示一个不同的符号。

参见 [符号集属性 - 外观通用（阵列）](#)。

[类型] 动画

选择此单选按钮可显示实际动画（顺序显示几个不同的符号）。

一旦被选中，对话框上的单选按钮将更改此对话框右侧的外观。这些选项仅在以下文档中记录一次。

参见 [符号集属性 - 外观通用（动画）](#)。

另请参见

[了解对象类型](#)

符号集属性 - 外观通用（开/关）

符号集具有以下通用外观（开/关）属性：

开符号显示时间（128 个字符）

在**开符号**域中输入的符号显示在此处输入的条件何时为 TRUE。在**Off 符号**域中输入的符号显示在此处输入的条件何时为 FALSE。

要插入标签或函数，点击此域右侧的“向导”按钮。此按钮将显示两个选择，**插入标签**和**插入函数**。

Off 符号

在以上条件不满足时显示的符号。点击**设置**选择符号，或点击**清除**清除当前选择。

例如，您可以在 **MIX_RUNNING** 为 false 时显示 Off 符号。

开符号

在以上条件满足时而显示的符号。点击**设置**选择符号，或点击**清除**清除当前选择。

例如，您可以在 **MIX_RUNNING** 为 true 时显示“开”符号。

点击**应用**或**确定**使更改生效，或点击**取消**放弃或退出。单击**清除属性**清除属性详细信息，然后禁止属性。要定义该对象的其他属性，请点击相关分页。

另请参见

[符号集对象](#)

[符号集属性 – 外观通用（多状态）](#)

[符号集属性 – 外观通用（阵列）](#)

[符号集属性 – 外观通用（动画）](#)

[了解对象类型](#)

符号集属性 – 外观通用（多状态）

符号集具有以下通用外观（多状态）属性：

条件

您在此处输入的条件可能在不同时间以不同方式一起发生。您可以用每个不同的组合来决定显示的符号。

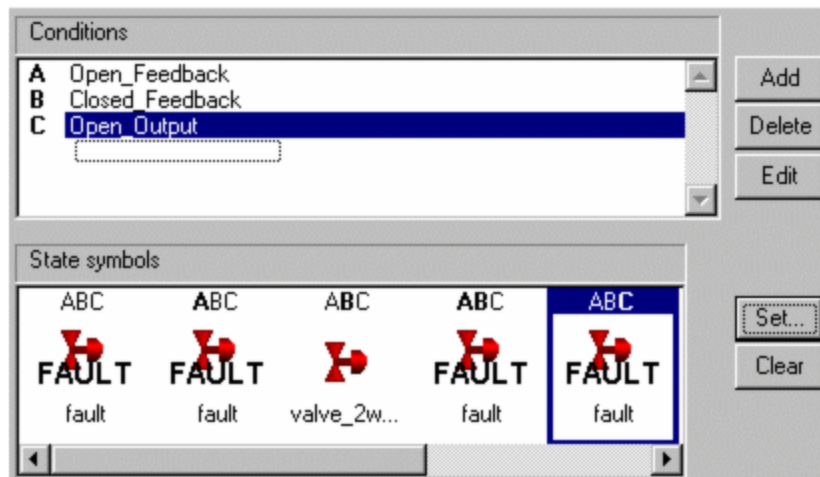
如需输入条件，点击相关行（A、B、C 等），然后点击**编辑**。您可以使用**添加**按钮添加更多条件（最多 5 个，提供 32 个组合）。如需插入标签或函数，点击**向导**按钮。此按钮显示两个选择，**插入标签**和**插入函数**。您也可以使用**删除**按钮来删除条件，但是此域中始终要有条件。空白的条件（而不是被删除的）将在运行时被视为 `false`。

状态符号

以上条件的每一种组合时所显示的符号。点击**设置**按钮选择符号，或点击**清除**清除当前选择。

例如：

如需为一个阀门的各种状态显示不同的符号，可以按如下所示填写**条件**和**状态符号**域：



在此例子中，**Open_Feedback** 和 **Close_Feedback** 是代表阀门上数字量输入内容的变量标签，而 **Open_Output** 是代表阀门上输出内容的变量标签。因此，**ABC** 表示 **Open_Feedback** 为“开启”状态，而 **Close_Feedback** 和 **Open_Output** 都为“关闭”状态。此组合时，故障符号将被显示，因为阀门在它应该关闭时打开了。相同的逻辑类型也同样适用于其它状态。

单击**清除属性**可清除属性的详细内容，然后禁止属性。要定义该对象的其他属性，请点击相关分页。

另请参见

[符号集对象](#)

[符号集属性 – 外观通用（开/关）](#)

[符号集属性 – 外观通用（阵列）](#)

[符号集属性 – 外观通用（动画）](#)

[了解对象类型](#)

符号集属性 – 外观通用（阵列）

符号集具有以下通用（阵列）外观属性：

阵列的表达式

输入返回一个或多个整数的表达式。为每个返回的整数，将显示一个不同的符号。

如果返回参数值是：

- 小于 0（零），将设置为 0（零），并将触发运行时硬件报警。
- 大于 255，将设置为 255，并将触发运行时硬件报警。
- 实数（非整数），将被四舍五入到最接近的整数。

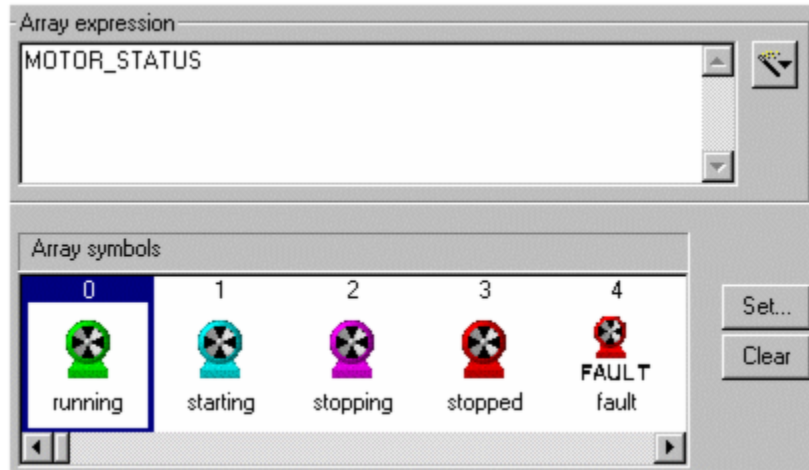
如需插入标签或函数，点击此域右侧的**向导**按钮。此按钮将显示两个选择，**插入标签**和**插入函数**。

阵列符号

为以上输入的阵列表达式返回的每个整数所显示的符号（符号 0 将在表达式返回整数为零时使用，符号 1 将在表达式返回整数为 1 时使用等等）。

点击**设置**按钮选择符号，或点击**清除**清除当前选择。

例如，要使用不同的符号来表示马达的不同状态，您可以按如下方式填写**阵列的表达式**和**阵列符号**：



在此例中，**MOTOR_STATUS** 是一个模拟量变量标签，用于表示马达状态。每次马达改变状态，返回一个整数（0 = Running、1 = Starting 等），并显示相应的符号。

点击**清除属性**可清除属性的详细内容，然后禁止属性。要定义该对象的其他属性，请点击相关分页。

另请参见

[符号集对象](#)

[符号集属性 – 外观通用（开/关）](#)

[符号集属性 – 外观通用（多状态）](#)

[符号集属性 – 外观通用（动画）](#)

[了解对象类型](#)

符号集属性 – 外观通用（动画）

符号集有以下通用外观（动画）属性：

动画条件/动画时候

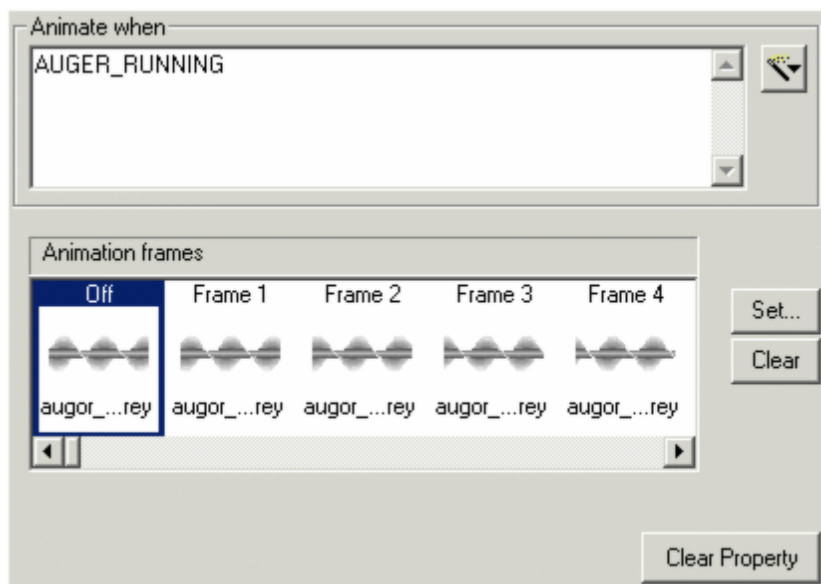
只要此表达式为 **true**，动画就会运行。只要此表达式为 **false**，“关闭”图形（见下图）将显示。

如需插入标签或函数，点击此域右侧的**向导**按钮。此按钮将显示两个选择，**插入标签**和**插入函数**。

动画帧

从图形 1 向上的符号是那些将被用来实现动画效果的。它们在表达式为真时顺序显示。符号显示的频率由 [Page]AnmDelay 参数确定。“关闭”图形中的符号将显示上面的表达式何时为 **FALSE**。

例如，要用动画表示一个运行的钻头，您可以填写**动画条件/动画时候**和**动画帧**字段，如下所示：



在此例子中，**AUGER_RUNNING** 是一个可用标签，它在钻头运行时为 **TRUE**。动画帧（帧 1 向上）中的符号是按照能够使运行的钻头能够动画地显示的顺序来设计的。“关闭”动画帧中的符号将显示 **AUGER_RUNNING** 何时为 **FALSE**。

点击**应用**或**确定**使更改生效，或点击**取消**放弃或退出。单击**清除属性**可清除属性的详细内容，然后禁止属性。要定义该对象的其他属性，请点击相关分页。

要获取其余属性分页的帮助，请参见 [定义通用对象属性](#)。

另请参见

[符号集对象](#)

[符号集属性 – 外观通用（开/关）](#)

[符号集属性 – 外观通用（多状态）](#)

[符号集属性 – 外观通用（阵列）](#)

[了解对象类型](#)

趋势对象

趋势工具允许您用鼠标（点击和拖动）来添加趋势至图形画面。

在绘制了趋势对象后，它能够被移动、改变大小、改变形状、提升至最上层等，它的属性能够被编辑，就如任何其它类型的对象一样。

如何添加一个趋势至一个画面：

1. 点击**趋势**工具。



或选择**对象 | 趋势**。

2. 移动光标至想要趋势起始的地方然后点击并按住鼠标键。
3. 拖动光标至趋势结束的点然后放开鼠标键。
将显示**趋势属性**对话框。为每支趋势笔分配趋势标签，并选择合适的颜色。

另请参见
[趋势属性](#)
[插入趋势对话框](#)
[了解对象类型](#)

趋势属性

趋势有以下总的外观属性：

集群名称

运行趋势图表的集群的名称。趋势图表只能与一个群集进行通信，因此，您不能将趋势与单个趋势图表上的多个群集相混淆。

注意： 要混合单个趋势图表上不同群集的趋势，您需要使用过程分析器。

如果只有一个群集，或客户机只连接到一个群集，此属性可以为空，并推断连接的单个群集的值。

如果客户机连接到多个群集，则必须指定此字段。

趋势笔

图形（最多 31 个字符）上显示的趋势笔（包括其颜色）。你最多可以使用八只趋势笔。

双击所选趋势笔，或点击**编辑**，更改趋势标签和趋势笔颜色。要插入趋势标签，点击**向导**，显示 [插入趋势对话框](#)。

如果在一个趋势窗口里显示多条趋势标签，且每条的采样周期各不相同，则具有最小采样周期的趋势的周期将被当作总的显示周期来使用。

注意： 如果趋势对象是一个组的一部分、粘贴的精灵或符号的一部分，或是画面模板的一部分，您仍然能够访问它的属性。按住 **Control (CTRL)** 键并双击对象。或者，也可以选择“工具”菜单中的**转至对象**，点击该对象，然后点击**确定**。注意，尽管如此，如果它是粘贴的精灵或符号的一部分，或模板的一部分，您不能编辑已有的趋势笔，您只能添加新的趋势笔。

如果您正在配置一个 SPC 控制图，必须在趋势标签后面添加一个后缀以表示它是 SPC 类型。可通过精灵轻松配置 SPC 模板。使用精灵，但不定义这些趋势标签。下表列出了可用的 SPC 类型：

SPC 定义	SPC 类型
--------	--------

SPC 定义	SPC 类型
<标签名称>.X	在一个子组内 (X - bar) 原始数据的平均值
<标签名称>.XCL	X - bar 的中心线
<标签名称>.XUCL	X - bar 的控制上限
<标签名称>.XLCL	X - bar 的控制下限
<标签名称>.R	在一个子组内 (R - bar) 的原始数据的范围
<标签名称>.RCL	R - bar 的中心线
<标签名称>.RUCL	R - bar 的控制上限
<标签名称>.RLCL	R - bar 的控制下限
<标签名称>.S	在一个子组内 (S - bar) 原始数据的标准偏离
<标签名称>.SCL	S - bar 的中心线
<标签名称>.SUCL	S - bar 的控制上限
<标签名称>.SLCL	S - bar 的控制下限

其中, <标签名称>可以是任何趋势标签, 例如:

Pen 1	PIC117_PV.XCL
Pen 2	PIC117_PV.XUCL
Pen 3	PIC117_PV.XLCL
Pen 4	PIC117_PV.X

如果您使用 PageTrend() 函数来显示此趋势画面, 此处保留为空。

显示所有趋势类型为周期

选择此项后, 将启用要显示的所有趋势笔 (周期性和事件) 为周期性。事件和周期趋势数据可以显示在同一图表中。如果未选择此框, 则事件和周期性趋势笔的样式将有所不同, 且必须显示在不同的图表上。

注意: 在设计用于周期性趋势的预定义模板中默认设置此选项。只需要针对自定义模板启用此选项。

[采样] 采样个数 (5 个字符)

在无需滚动的情况下，您能够在您的趋势窗口中显示的采样个数 (1 - 32767) (即趋势对象的宽度)。缺省值依每个采样所占的像素个数和您的显示分辨率而定。趋势对象的宽度等于**每个采样所占像素个数 x 采样的个数**。

注意： 对于一个有意义的趋势图，**每个采样所占像素个数 x 采样的个数**应该小于显示的宽度。例如，一个 XGA 屏幕有 1024 像素宽。如果您每个采样使用 10 个像素，102 个采样可以在屏幕上无需滚动即可显示。

[采样] 每个采样所占像素个数 (2 个字符)

每个采样的显示宽度。趋势对象的宽度等于**每个采样所占像素个数 x 采样的个数**。缺省值是 1 个像素。

单击**清除属性**可清除属性的详细内容，然后禁止属性。要为该对象定义其他属性，请单击相关分页标签。

要获取其余属性分页的帮助，请参见 [定义通用对象属性](#)。

插入趋势对话框

此对话框使您能够选择一个趋势标签。要插入一个趋势标签，请选择标签名，然后点击**确定**按钮。标签在鼠标光标处插入。

另请参见
[趋势对象](#)
[了解对象类型](#)

Cicode 对象

Cicode 对象工具允许您只需通过鼠标（单击和拖动）来添加一个 Cicode 对象至图形画面。

Cicode 对象可以是任何命令(如一个函数等)。当图形画面在运行期间显示时，该命令会不断地运行。Cicode 对象也可以被赋予一个键盘输入键序列，允许您在运行期间选取它并输入键盘命令。

添加 Cicode 对象后，就可以对它进行移动等操作，而且可以编辑其属性，就像其他类型的对象一样。

如何将 Cicode 对象添加到画面：

1. 单击 **Cicode 对象** 工具，



或选择**对象 | Cicode 对象**。

2. 将鼠标移到要添加对象之处，然后单击鼠标左键。

3. 定义对象的相关属性，然后单击**确定**。

另请参见

[Cicode 对象属性 - Cicode \(通用\)](#)
[了解对象类型](#)

Cicode 对象属性 - Cicode (通用)

Cicode 对象具有以下通用属性：

命令 (254 个字符)

不断执行的 Cicode 命令。你可以使用任何 Cicode 命令、内置 Cicode 函数或用户编写的函数。命令被不断地执行（同时显示画面），例如：

```
命令 DspSymAnm(25, "Pumps.Slurry1", "Pumps.Slurry2", "Pumps.Slurry3");
```

本例中的命令使用了内置函数 DspSymAnm()。此函数不断地显示三个符号 ("Pumps.Slurry1"、"Pumps.Slurry2"、"Pumps.Slurry3")(在 AN 25)。

你也可以通过使用 Cicode 函数 DspGetAnCur() 来编写一个通用函数以获取 AN 数，例如：

```
命令 DspSymAnm(DspGetAnCur(), "Pumps.Slurry1", "Pumps.Slurry2", "Pumps.Slurry3");
```

本例中的命令不断显示三个符号 ("Pumps.Slurry1"、"Pumps.Slurry2"、"Pumps.Slurry3") (以当前 AN)。

如果使用的是实际动画，每个符号以使用计算机设置向导设置的频率来显示（此频率也可由 [Page] AnmDelay 参数决定）。要将一个动点添加到画面，请添加一个 Cicode 对象，不要使用命令。

单击**清除属性**可清除属性的详细内容，然后禁止属性。要为该对象定义其他属性，请单击相关分页标签。

要获取剩余属性分页的帮助，请参见 [定义通用对象属性](#)。

使用动点

图形画面（对象在该页面中显示）中的每个点称为动点。添加对象（文本、符号、管道等）到页面中时，会自动为动点分配一个编号（称为 AN）。

可以使用的对象编号受计算机性能的限制，尽管这在现代计算机上不是问题。最好尝试保持对象编号（也就是 AN）小于 3000。

前两个 AN 用于自动显示系统信息，如消息、报警信息和页面详细信息。在某些应用程序中（如趋势页面），还保留了其它一些 AN。

可以使用 **Cicode 对象** 工具将单个动点添加到图形页面（无需使用命令添加 Cicode 对象）。

注意：可以通过引用对象的 AN 来在页面中查找任何对象。要按 AN 查找对象，请使用工具菜单中的**定位对象**工具。

粘贴的符号对象

使用粘贴符号工具，可将符号从库中插入到图形画面上。



使用此工具粘贴符号后，可对其执行移动、改变大小、改变形状、提升至最上层等操作，并且像其他对象那样编辑其属性。

粘贴的[库]符号能够与其来源链接，这样对原件的更改也会被应用于粘贴的符号。

要（在粘贴后）显示符号中对象的属性，可按住**控制(CTRL)**键双击对象。或者，也可以选择工具 | **定位对象**，点击该对象，然后点击**确定**。

要了解有关生成符号的更多信息，请参见 [使用库](#)。

如何将符号从库中粘贴到画面上：

1. 点击**粘贴符号**工具，或选择**编辑 | 粘贴符号**。
2. 要粘贴链接的符号，请选中**链接**复选框。要粘贴不链接的符号，请取消选中**链接**复选框。

如何取消链接：

1. 选定要取消链接的符号。
2. 选择**编辑 | 删除关联**。

另请参见

[粘贴符号对话框](#)

[符号属性 - 外观（通用）](#)

粘贴符号对话框

可通过此对话框将符号从库中粘贴到图形画面（或模板）中。

“粘贴符号”对话框具有下列属性：

符号

工程中的符号表。

要向图形画面中添加符号，请使用滚动条定位到该符号的缩略图，然后选定它并点击**确定**按钮（或双击该缩略图）。若要编辑库中的对象，请将其选定并点击**编辑**。要生成一个新符号，请单击**新建**。

注意：如果符号旁边有一个小钻石形标记，表示它是一个闪烁符号（参见下面的示例）。



库

存放符号的库。

链接

如需粘贴一个与它的库保持链接的符号，请选中此框。如果库里的符号被修改，链接的符号也将自动被更新。

可以从编辑菜单中用**取消链接**命令在任何时间来取消链接，但是无法在取消链接后再重新恢复一个符号与库之间的链接。

如果选择了**粘贴符号为闪烁的**，则会相继出现两个对话框，允许您选择要设置为闪烁符号的两个图像。通过**主要选择符号**对话框，可以选择要使用的初始符号；通过**闪光选择符号**对话框，可选择第二个符号。如果位图的大小不同，则闪烁符号会缩放到主要图像的大小。如果一个符号是其自身的闪烁符号，则只会显示主要状态。

符号属性 - 外观（通用）

此对话框显示一个图像，其中包含图形、名称和路径。点击**设置**更改符号，或双击此图像。显示“选定符号”对话框，您可以从中选择新符号。

要获取有关其余属性分页的帮助，请参见 [定义通用对象属性](#)。

如需修改一个被粘贴的符号里的对象的属性：

1. 点击**选择**工具。
2. 按下控制 (CTRL) 键并双击对象。
3. 更改此对话框中的相关属性。或者，选择**工具 | 转到对象**，选择该对象，然后点击**确定**。

另请参见

[粘贴的精灵对象](#)
[了解对象类型](#)

粘贴的精灵对象

使用粘贴精灵工具，可将精灵插入到图形画面上。



使用此工具粘贴精灵之后，可对其执行改变大小、旋转、移动、拷贝、复制、粘贴、提升至最上层等操作。

要（在粘贴后）显示精灵中某个对象的属性，可按住**控制(CTRL)**键双击该对象。

ActiveX 对象

您可以将 ActiveX 对象并入项目中。这意味着，您可以使用独立于 Vijeo Citect 开发的组件。

ActiveX 工具可用于在图形画面中插入 ActiveX 对象。选择并放置一个 ActiveX 对象之后，即可将其移动、改变其大小、改变其形状，将其提升到页面最前面等，而且可以像处理任何其它对象一样编辑其属性。

管理关联数据源

如果 ActiveX 对象与数据源相关联（例如，它将数据存储到 DBF 文件），则需要考虑在另一台机器上或通过互联网客户机（互联网显示客户机或 Web 客户机）运行包含该数据源的工程的影响。

如果将数据源的路径硬编码为本机上的位置，则当该工程移走或远程运行时将找不到该数据源。例如，数据库交换 ActiveX 控件连接到该工程路径中的 recipe.DBF 文件。如果在具有不同安装路径的另一台计算机上恢复使用它的工程，将需要重新创建该数据源以获取所有配方。



不希望的设备操作

- 尽可能地避免在您的工程中对数据源的文字路径进行编程。
- 当工程或数据源被重定位到其他计算机时，应测试程序和程序对象并在将工程返回使用之前更正所有的文字路径。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

避免重新创建数据源的一种方法是将任何关联的数据源放在网络的中心位置。例如，如果数据源位于 SQL 服务器上，将可以从该共同网络的每台机器访问它。

插入一个 ActiveX 控件：

1. 点击 **ActiveX** 工具，



或选择**编辑 | 插入 ActiveX 控件**。

2. 选取一个 ActiveX 控件并点击**插入**。

另请参见

[ActiveX 对象属性](#)

[标签关联](#)

[对象识别](#)

ActiveX 对象属性

所有 ActiveX 对象的两个通用属性标签是“**标签关联**”和“**可见性**”，这两个标签沿垂直方向出现在“**外观**”标签上。所有其它标签的内容和数目取决于每个 ActiveX 对象的个别设计。这由对象的创建者具备的灵活性和支持来决定。

有关配置这些附加标签的详细信息，请参考随 ActiveX 对象一起提供的文档。

另请参见

[标签关联](#)

标签关联

可以在 ActiveX 对象的属性和变量标签间建立关联。

要在 ActiveX 对象的属性和变量标签间创建关联：

1. 双击 ActiveX 对象。此时将显示“属性”对话框。
2. 点击**标签关联**分页。
3. 从**属性**列表中选择一个属性。
4. 点击**向导 | 插入标签**按钮。
5. 从列表选取一个标签，然后点击**确定**。

另请参见

[Vijeo Citect 用户指南中的“ActiveX 对象属性”](#)

[Vijeo Citect 用户指南中的“了解对象类型”](#)

ActiveX 对象属性 - 外观（标签关联）

ActiveX 对象具有以下外观属性：

属性

ActiveX 对象的“属性”与定义对象的功能和外观的元素有关。此处列出了选定的 ActiveX 对象的所有可用属性，与对象的创建者定义的一样。

列表中的每一项左侧的复选框表明标签是否与属性相关联。如果选中了该框，则意味着已为属性定义了一个关联。如果要清除此标签关联，只需取消选中该框，或从“关联属性与标签.”字段中清除该标签。

关联属性与标签

您可以在一个 ActiveX 对象属性与变量标签之间创建一个关联，以使其中一项中不断变化的值反映在另外一项中。要创建关联，首先需要从属性列表中选择一属性。**关联属性与标签**标签将改为显示属性名称。

通过点击“向导”按钮并从可用标签列表中进行选择，选择要与一个属性关联的变量标签。或者，可以将标签名键入**关联属性与标签**字段中。

注意：您只能在此字段中使用变量标签名。在定义 ActiveX 属性标签关联时，函数、表达式和常量不受支持。

您只能将一个与属性兼容的变量标签与属性相关联。要显示兼容标签类型的列表，请选择相应的属性，然后点击“**属性类型列表**”。此时将显示一个兼容数据类型的列表，或者显示一条消息，告诉您不存在兼容的类型。

如果不存在与属性兼容的类型，则“**关联属性与标签**”和“**更新关联条件**”字段将被禁用。

如果指定的标签对选定的 ActiveX 属性可能不适合，则“**类型评估对话框**”将显示一条提醒信息。在以下条件下，可能会发生这种情况：

- 与属性兼容的类型不同于标签类型。
- 标签类型比与属性兼容的类型小，也就是说数据可能会被截取或丢失。

可捆绑

如果对属性“可捆绑”，则意味着对象属性可以将值更改通知自动发送到 Vijeo Citect，并确认来自关联标签的所有值更改。这意味着，只要任何一方的值发生更改，属性和关联的标签将会自动更新。

如果属性不可捆绑，则属性/标签关联只能依据在“**更新事件关联**”字段中选择的事件进行同步。

请注意，如果一个对象属性是可捆绑的，则缺省情况下，**更新事件关联**字段将自动设置为<**属性更改通知**>。如果希望标签关联按更具体的事件加以更新，则可以更改此设置。

对于不可捆绑的属性，您可以通过在“**更新事件关联**”字段中选择“**更新后**”来模仿<属性更改通知>的行为。

属性状态

表示选定属性的读/写状态。在 ActiveX 对象中，某些属性是只读的，而其它属性将接受值更改。如果一个属性被标记为“只读”，则表示其值是固定的，而且只能由 Vijeo Citect 读取。如果其状态是读/写，则可以通过标签关联在运行时修改属性。

更新事件关联:

这定义了您希望对于选定属性及其关联的标签何时进行值更新。使用“**事件**”字段右侧的菜单，可以查看可用于触发标签关联更新的事件。

可与特定属性一起使用的可用事件由对象的创建者预定义。它们通常包括用户交互事件（例如鼠标点击）、时间事件（如新的一天或新的一个月）、或值更改（如“更新后”）。

属性文档

大多数 ActiveX 对象都具备描述对象的控件和功能的文档。某些对象包含一个单独的帮助文件，其它对象可能有简单的文本提示来说明每个属性。这取决于对象创建者已纳入的内容。

属性文档字段显示选定属性的帮助信息，或提供用于获取选定属性所需帮助的指导说明。通常会出现以下消息：

“点击右边‘?’按钮以显示此属性的帮助主题”

“**帮助**”按钮将显示 ActiveX 对象的帮助文件（如果包含），通常与选定属性相关的主题会被显示。此外，还应提供 ActiveX 对象可能会在属性对话框中调用的其他标签上提供的设置有关的信息。

清除属性

“清除属性”按钮用于清除所有 ActiveX 对象属性的标签关联。要清除特定对象属性的标签关联，请清除相应属性左侧的复选框。

如果您无意中点击了“**清除属性**”按钮，则可以通过点击“**取消**”按钮并重新打开 ActiveX 属性对话框来恢复标签关联。

对象标识

您可以对 ActiveX 对象进行标识。

如何标识 ActiveX 对象：

1. 在图形编辑器中，双击“ActiveX”对象。将显示“属性”对话框。
2. 点击**访问**标签。
3. 点击**标识**标签。
4. 在**对象名称**字段中为您的 ActiveX 对象**赋予**一个名称。
5. **赋予**您的 ActiveX 对象一个**事件类别**。
6. 点击**确定**。

另请参见

[对象属性 - 访问（标识）](#)

对象属性 - 访问（标识）

对象有以下访问标识属性：

[标识] 对象名称

使用此字段，您可以为您的 ActiveX 对象赋予一个名称（最多 254 个字符）。使用 `ObjectByName()`、`ANByName()` 和 `CreateControlObject()` `Cicode` 函数时，它将对对象进行标识。

此名字可以是任何字母或数字字符的组合。

[标识] 事件类

可以为您的 ActiveX 对象的事件类分配一个名称（16 个字符）。然后，您就可以使用这个名称来创建一个 `Cicode` 函数，从而跟踪事件。

如果使用 `CitectVBA` 来访问 ActiveX 对象，请不要更改缺省参数值。如果更改了缺省参数值，则 `CitectVBA` 无法访问对象。如果更改了事件类，您可以点击**清除属性**，将其复位回缺省参数值。

[保存数据] 保存画面过渡间的 ActiveX 数据

选择此复选框，可以更改画面间保存的控件。例如，如果您希望使用 ActiveX 编辑控件来保存控件中的当前文本，则可以选择此复选框，这样，下次进入此画面时，就会显示相同的文本。

注意： ActiveX 控件可能会也可能不会以您期望的方式实现数据长久保留。某些控件不会长久保留数据，因此您无法保存和恢复该数据。

注意： 即使某个 ActiveX 控件以您期望的方式实现数据长久保存，建议数据仅为当前会话而保留。如果 `Vijeo Citect` 运行时关机并重新启动，更新的数据将不可用。

数据库交换控制对象

通过“数据库交换 ActiveX 控件，可以连接到数据源（如数据库），还可以提取、显示和编辑配方值。此控件使用 `Citect` 图形编辑器插入图形画面并在其上配置。相关信息，请参见数据库交换对象在线帮助 [数据库交换对象](#) 在线帮助。

Vijeo Web Gate 控制对象

`Vijeo Web Gate` 控制对象提供了 HMI 目标的远程查看器，该查看器可以嵌入到 `Vijeo Citect` 图形画面中。根据 `Web Gate` 的设置，您可以远程查看 HMI 显示，从远程 PC 读写目标计算机上的数据。相关信息，请参见 `Vijeo Web Gate` 对象联机帮助。

Chapter: 21 定义通用对象属性

本节介绍通用于数个对象类型的属性。有些还通用于对象组。

另请参见

[3D 效果](#)

[可见性](#)

[移动](#)

[缩放](#)

[填充颜色](#)

[填充方向](#)

[触击命令](#)

[键盘命令](#)

[滑钮](#)

[访问](#)

3D 效果

可以将 3D 效果应用于对象，从而使它们更加逼真。

将 3D 效果应用到某个对象：

1. 绘制对象/组合（或粘贴一个符号）。属性分页对话框将自动显示，除非您在图形编辑器中关闭了“新建对象时显示属性”选项。（对于一个组合，属性对话框不会自动显示；您必须双击该组合。）
2. 点击“外观”分页。
3. 点击“3D 效果”分页（在该对话框的右边）。
4. 该对话框将显示多个可用于处理对象的选项。如需激活其中任何选项，请点击相应的选项（或选项左边的选择圆圈）。
5. 通过选取一定的选项将会有一些附加的域被显示，使得您能够更进一步处理对象。按要求输入更详细的信息，使用“帮助”按钮以获取有关每个域的详细信息。
6. 点击“确定”。

另请参见

[对象属性 - 外观（3D 效果）](#)

[定义通用对象属性](#)

对象属性 - 外观（3D 效果）

对象有以下 3D 效果属性：

[效果] 无

选择此选项，可以将对象显示为无任何特别效果（如阴影、浮雕等）。

[效果] 阴影

选择此选项，可以使用阴影效果显示对象，例如：



深度

阴影在对象右下方伸展的距离（以像素为单位）。此选项可以改变对象与其阴影之间的外观距离，例如：



阴影颜色

阴影的颜色。阴影的颜色将不会随着运行时条件的改变而动态地变化。

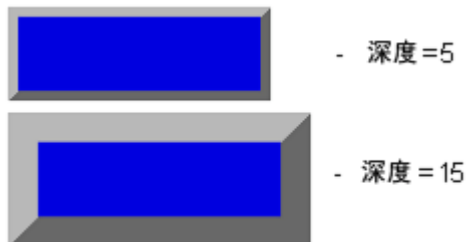
[效果] 突出

选择此选项，可以将对象显示为一个凸起的三维实体，例如：



深度

对象的边界从凸起的表面延伸出来的距离（以像素为单位）。使用此选项，可以改变从凸起的表面至您的图形画面之间的外观距离，例如：



高亮颜色

对象直接点亮的“边界”的颜色。



低亮颜色

对象在阴影中的“边界”颜色。



[效果] 凹陷

选择此选项，可以将对象显示为好像低于您的图形画面一样，例如：



深度

对象的边界从凹陷的表面延伸出来的距离（以像素为单位）。使用此选项，可改变从凹陷的表面至您的图形画面之间的外观距离，例如：



高亮颜色

对象直接点亮的“边界”的颜色。



低亮颜色

对象在阴影中的“边界”颜色。



[效果] 浮雕

选择此选项，可以将对象显示为好像是您的图形画面上的浮雕，例如：

Embossed Text

深度

对象的浮雕表面凹陷的距离（以像素为单位）。使用此选项，可改变浮雕表面至您的图形画面之间的外观距离，例如：

Embossed Text - 深度 = 1

Embossed Text - 深度 = 2

高亮颜色

对象右下边界的颜色。

Embossed Text 高亮颜色

低亮颜色

对象左上边界的颜色。

低亮颜色 Embossed Text

点击**应用**或**确定**，保存更改，或点击**取消**，退出。要定义该对象的其他属性，请点击相关分页。

可见性

您可以决定一个对象是否可见。

如何隐藏/显现一个对象：

1. 双击要隐藏的对象。
2. 选择**外观**分页标签。
3. 选择**可见性**分页标签（对话框的右边）。
4. 单击**何时隐藏**字段右边的**向导**按钮。
5. 选择**插入标签**或**插入函数**，依您想与您的对象关联的条件。
6. 在**何时隐藏**字段中输入一个表达式。当此表达式为真时您的对象将被隐藏。
7. 单击**确定**。

另请参见

[对象属性 - 外观（可见性）](#)

[定义通用对象属性](#)

对象属性 - 外观（可见性）

对象和组有以下的可见性属性：

隐藏条件

只要在此字段中输入的表达式为真，对象/组就会隐藏。输入表达式的字符不得超过 128 个。例如，如果您希望对象/组对除管理员以外的所有操作员隐藏，则可以输入以下内容：

```
NOT GetPriv( _Super, _SectionA )
```

其中，_Super 和 _SectionA 是标签。

如需插入标签或函数，点击字段右边的**向导**按钮。此按钮将显示两个选项：插入标签和插入函数。

注意： 如果某个组被隐藏，则组中的所有对象（以及其他组）也将被隐藏，而不管其各自的属性设置。如果组是可见的，则其中的对象将依其自身的属性来显示。

点击**清除属性**，可清除属性的详细信息，然后禁止属性。

移动

用户可以控制对象的移动。

如何配置移动的对象或组：

1. 绘制对象/组（或粘贴符号）。属性标签对话框将自动显示，除非在图形编辑器中关闭了**新建对象时显示属性**选项。（如果是组，属性对话框不会自动显示；必须双击该组才能显示。）
2. 单击**平移**标签。
3. 单击**水平**、**垂直**或**旋转**标签（位于对话框右侧）。
4. 输入平移**表达式**（表达式将在运行时移动对象/组）。
5. 根据需要输入更详细新的信息，使用**帮助**按钮可获得有关每个字段的详细信息。
6. 单击**确定**。

另请参见

[对象属性 - 移动（水平）](#)

[对象属性 - 移动（垂直）](#)

[对象属性 - 移动（旋转）](#)

[组和对象移动 - 范例](#)

对象属性 - 移动 (水平)

对象和组可以在运行时从一边移动到另一边，当特定表达式的参数值改变时，对象和组也将动态地变化。缺省情况下，随着表达式的参数值的增加，对象/组将会向右移动（增量）。随着表达式的参数值的减少，对象/组将会向左移动（增量）。

例如，此属性可用于显示煤炭在运输带上移动的位置。

注意： 如果启用了水平滑钮，则无法使用水平移动。可以使用任意移动组合来配置组及其对象（即组可以在其对象旋转时进行垂直移动）。

对象和组有以下水平移动属性：

移动表达式

在此字段中输入的表达式参数值（最多 253 个字符）决定了对象/组的水平移动。缺省情况下，当表达式返回最小参数值时，对象/组将会移动至最左边。当表达式返回最大参数值时，对象/组将会移动至最右边。返回中间参数值时，对象/组将会移动至最小值和最大值偏移之间的相应位置。

如需插入标签或函数，点击字段右边的**向导**按钮。此按钮将显示两个选项：插入标签和插入函数。

[移动表达式] 指定范围

选择此复选框，可手动指定移动表达式的最小和最大参数值，而不使用缺省参数值。对于一个包含模拟变量标签的表达式，缺省的参数值为表达式中最后变量标签中的工程零刻度和满刻度值。如果此模拟变量标签没有工程零刻度和满刻度值，那么缺省值是 0（零）和 32000。对于没有标签的表达式，缺省值为 0（零）和 100。

[移动表达式] 最小值

输入表达式的最小参数值。当表达式返回此参数值时，对象/组将会移动至最左边，**最小**偏移。如果您选择**指定范围**复选框，则您只能在此输入一个参数值。

[移动表达式] 最大值

输入表达式的最大参数值。当表达式返回此参数值时，对象/组将会移动至最右边，**最大**偏移。如果您选择**指定范围**复选框，则您只能在此输入一个参数值。

[偏移] 最小

当**平移表达式**返回最小参数值时，对象/组将会向左移动的距离（从起始的对象/组中心的像素数）。

您可以按字段右边的上下箭头，或直接在字段中输入另一个参数值来修改偏移参数值。

[偏移] 最大

当**平移表达式**返回最大参数值时，对象/组将会向右移动的距离（从起始的对象/组中心的像素数）。

您可以按字段右边的上下箭头，或直接在字段中输入另一个参数值来修改偏移参数值。

注意：您可以通过在“偏移”字段内输入负距离值，或交换（**最小值**和**最大值**字段内的）表达式极限值来使对象/组在其为最小值时移至右边，在其为最大值时移至左边。

点击**清除属性**，可清除属性的详细内容，然后禁止属性。

另请参见

[对象属性 - 移动（垂直）](#)

[对象属性 - 移动（旋转）](#)

对象属性 - 移动（垂直）

对象和组合可以在运行时被上下移动，当一个特定的表达式的值改变时动态地变化。缺省情况下，表达式的值增加时对象/组合将会向上移动（增加）。表达式的值减少时对象/组合将会向下移动(增加)。

此属性可被用来显示一个电梯的移动。

注意：如果垂直滑钮已启用，则不能使用垂直移动功能。组及其对象可用任意移动组合来配置（即组可以在其中的对象旋转时垂直移动，等）。

对象和组具有以下垂直移动属性：

移动表达式

在此域输入的表达式的值（最多 253 个字符）将确定对象/组的垂直移动。缺省情况下，当表达式返回它的最小值时，对象/组合将会移动至最下边。当表达式返回它的最大值时，对象/组将会移动至最上边。中间值时，对象/组将会移动至最大最小值之间相应的位置。

要插入标签或函数，按此域右侧的**向导**按钮。此按钮显示两个选项：插入标签和插入函数。

[移动表达式] 指定范围

选中此框可手动为移动表达式指定**最小值**和**最大值**，而不使用缺省值。对于包含模拟变量标签的表达式，缺省值是表达式的最后一个变量标签中的工程零刻度和满刻度值。如果此模拟变量标签没有工程零刻度和满刻度值，那么缺省值是 0 (零) 和 32000。没有标签的表达式里缺省值为 0 (零) 和 100。

[移动表达式] 最小值

输入表达式的最小值。当表达式返回此值时，对象/组合将会向下移动**最小**偏移量。如果已选中**指定范围**框，您只能在此输入一个值。

[移动表达式] 最大值

输入表达式的最大值。当表达式返回此值时，对象/组将会向上移动**最大**偏移量。如果已选中**指定范围框**，您只能在此输入一个值。

[偏移] 最大处

当**平移表达式**返回其最大值时，对象/组将会向上移动的距离（距原始对象/组中心的像素数）。

你可以按域右边的上下箭头或直接输入来修改偏移值。

[偏移] 最小处

当**平移表达式**返回其最小值时，对象/组合将会向下移动的距离（距原始对象/组中心的像素数）。

你可以按域右边的上下箭头或直接输入来修改偏移值。

注意：您可以通过在偏移域内输入负的距离值或交换极限值（位于**最小值**和**最大值**域中）来使得对象/组在其为最小值时移至上边和在其为最大值时移至下边。

另请参见

[对象属性 - 移动 \(水平\)](#)

[对象属性 - 移动 \(旋转\)](#)

对象属性 - 移动 (旋转)

运行时，一旦特定表达式的值发生改变，对象和组可能会动态旋转。缺省情况下，表达式的值增加时，对象/组将顺时针旋转（增加）。表达式的值降低时，对象/组将逆时针旋转（增加）。

此属性可被用来显示一个采矿业中的圆形的储存器的旋转外观。

注意：如果旋转滑钮已启用，则不能使用旋转移动功能。组及其对象可用任意移动组合来配置（即组可以在其中的对象旋转时垂直移动，等）。

对象和组具有以下旋转移动属性：

角度表达式（253 个字符。）

在此域输入的表达式的值将会决定对象/组合的旋转。运行时，当表达式返回它的最小值时，对象/组合将会旋转至逆时针极限。当表达式返回它的最大值时，对象/组合将会旋转至顺时针极限。中间值时，对象/组合将会旋转至最大最小值之间相应的位置。

要插入标签或函数，按此域右侧的**向导**按钮。此按钮显示两个选项：插入标签和插入函数。

[角度表达式] 指定范围

选中此框可手动为角度表达式指定**最小值**和**最大值**，而不使用缺省值。对于包含模拟变量标签的表达式，缺省值是表达式的最后一个变量标签中的工程零刻度和满刻度值。如果此模拟变量标签没有工程零刻度和满刻度值，那么缺省值是 0 (零) 和 32000。没有标签的表达式里缺省值为 0 (零) 和 100。

[角度表达式] 最小值

输入表达式的最小值。当表达式返回此值时，对象/组合将会逆时针旋转最小偏移量。如果您已选中**指定范围框**，您只能在此输入一个值。

[角度表达式] 最大值

输入表达式的最大值。当表达式返回此值时，对象/组将以最大偏移量为单位顺时针旋转。如果您已选中**指定范围框**，您只能在此输入一个值。

[角度] 最小处

当**角度表达式**返回其最小值时，对象/组逆时针旋转的角度（从 0 开始的度数）。

你可以按域右边的上下箭头或直接输入来修改角度值。

[角度] 最大处

当**角度表达式**返回其最大值时，对象/组顺时针旋转的角度（从 0 开始的度数）。

你可以按域右边的上下箭头或直接输入来修改角度值。

注意：您可以通过在角度域内输入负的角度值或交换表达式极限值（位于**最小值**和**最大值**域中）来使得对象/组在其为最小值时逆时针旋转至极限和在其为最大值时顺时针旋转至极限。

[中心轴偏移] 快速

点击此单选按钮能快速简单地选择对象/组旋转所围绕的点。快速选项赋予你选择 9 点的权利（左上、右下等），这些点显示在对话框右边的图形内。要选择一个，只需用鼠标点击即可。

[中心轴偏移] 定制

点击此单选按钮来定义您自己的中心轴。当您选择此选择按钮，右边将显示两个域以允许您画定您的中心轴位置。在第一个域内指定向右的距离，而在第二个内指定向下的距离。中心轴就是以这两个值来画的。

例如，如果你输入 8 作为水平偏移，13 作为垂直偏移，则中心轴将位于从对象/组中心向右 8 个像素、向下 13 个像素处。

在偏移距离域内输入负值以将中心轴向左（而不向右），向上（而不向下）移动。

另请参见

[对象属性 - 移动\(水平\)](#)

[对象属性 - 移动\(垂直\)](#)

组和对象运动 - 范例

可使用任何运动（水平、垂直和旋转）组合来配置组及其对象。以下此范例用图形方式说明了这些组合是如何工作的。

范例 1：旋转组并将对象从左向右移动

如果组被配置为从 0 度旋转到 60 度，它的一个对象被配置为从左向右移动，该对象将进行这两项运动。它会按照自己的属性从左向右移动，同时作为组的一部分旋转。但请记住，“左”和“右”是以组而非画面的原始方向为参照的。随着组的旋转，“水平”也旋转。当组旋转 15 度时，“左”实际为 285 度（不是 270 度），而“右”实际为 105 度（不是 90 度）。当组旋转 50 度时，“左”是 320 度，“右”是 140 度，依此类推。

组的原始状态



组向右旋转，椭圆向左移动



范例 2：旋转组并上下移动对象

如果组被配置为从 0 度旋转到 60 度，它的一个对象被配置为上下移动，该对象将进行这两项运动。它会上下移动，同时作为组的一部分旋转。但请记住，“上”和“下”是以组而非画面的原始方向作为参照的。随着组的旋转，“垂直”也旋转。当组旋转 15 度时，“上”实际为 15 度（不是 0 度），而“下”实际为 195 度（不是 180 度）。当组旋转 50 度时，“上”是 50 度，而“下”是 230 度，依此类推。

组的原始状态



组向右旋转，椭圆向上移动



范例 3：顺时针旋转组，逆时针旋转对象

如果组被配置为从 0 度旋转到 -60 度，它的一个对象被配置为从 90 度旋转到 0 度，该对象将进行这两项运动。它会按照自己的属性旋转，同时作为组的一部分旋转。但请注意，对象的旋转是以组而非画面作为参照的。如果组向右旋转 60 度，而对象向左旋转 90 度，那么对象仅相对于画面向左旋转了 30 度。

组的原始状态



组顺时针旋转，椭圆逆时针旋转



注意：通过移动上述范例中所示的椭圆，您实际上更改了组的整体大小。请重点关注此项，因为它可能会影响到对象的填充高度。

另请参见：

[运动](#)

[定义通用对象属性](#)

缩放

可以将对象缩放为所需的大小。

如何配置更改大小的对象或组：

1. 绘制对象（或粘贴符号）。“对象属性”标签对话框将自动显示，除非在图形编辑器中关闭了**新建对象时显示属性**选项。
2. 单击**缩放**标签。
3. 单击**水平**或**垂直**标签（位于对话框右侧）。
4. 输入**缩放表达式**（运行时更改对象大小的表达式）。
5. 根据需要输入更多对象属性详细信息，并使用**帮助**按钮获取有关每个字段的详细信息。
6. 单击**确定**。

另请参见

[对象属性 - 缩放（水平）](#)

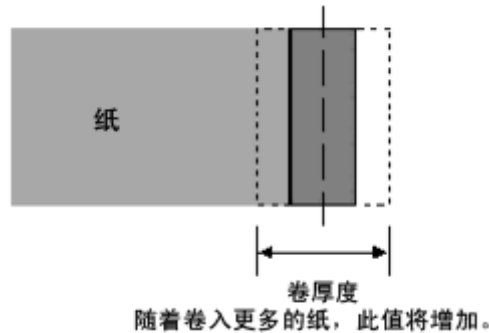
[对象属性 - 缩放（垂直）](#)

[定义通用对象属性](#)

对象属性 - 缩放（水平）

对象的宽度可以在运行时被动态地改变（当特定表达式的值改变时）。表达式的值增加或减少时，对象的宽度将会根据原始宽度的百分比相应地增加或减少；即当它被加入图形画面时。

例如，一个造纸厂的卷纸机的侧视图能够显示纸卷的厚度：



对象有以下水平缩放属性：

缩放表达式

输入此域的表达式的值（最多 253 个字符）确定对象的水平缩放（宽度）。缺省情况下，当表达式返回最小值时，对象将会以其（定义在以下缩放域的）最小宽度显示。当表达式返回最大值时，对象将会以其（定义在以下缩放域的）最大宽度显示。

要插入标签或函数，点击此域右侧的**向导**按钮。此按钮显示两个选项：插入标签和插入函数。

[缩放表达式] 指定范围

选中此框可手动为缩放表达式指定**最小值**和**最大值**，而不使用缺省值。对于包含模拟变量标签的表达式，缺省值是表达式的最后一个变量标签中的工程零刻度和满刻度值。如果此模拟变量标签没有工程零刻度和满刻度值，那么缺省值是 0 (零) 和 32000。没有标签的表达式里缺省值为 0 (零) 和 100。

[缩放表达式] 最小值

输入表达式的最小值。当表达式返回此值时，对象的宽度将会减小到它的最小值。如果已选中**指定范围框**，您只能在此输入一个值。

[缩放表达式] 最大值

输入表达式的最大值。当表达式返回此值时，对象的宽度将会增加到它的最大值。如果已选中**指定范围框**，您只能在此输入一个值。

[缩放] 最小处

对象的最小宽度（依其原始宽度的百分比）。当**缩放表达式**返回其最小值时，对象将会被缩小至此宽度。

你可以按域右边的上下箭头或直接输入来修改百分比。可以输入大于 100% 的百分比值。

[缩放] 最大处

对象的最大宽度（依其原始宽度的百分比）。当**缩放表达式**返回其最大值时，对象将会被增大至此宽度。

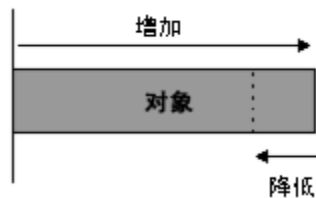
你可以按域右边的上下箭头或直接输入来修改百分比。可以输入大于 100% 的百分比值。

注意：您可以通过交换缩放域内的百分比值（即在**最小处**域内输入高百分比值，而在**最大处**域内输入低百分比值），或交换表达式极限值（在**最小值**和**最大值**域中）来在其为最小值时增加对象的宽度和在其为最大值时减小对象的宽度。

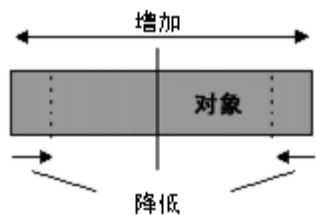
[中心轴偏移] 快速

点击此选项以快速简便地选择对象的三个垂直轴（左侧、中间和右侧）之一。这些轴显示在对话框右侧的图形域内。

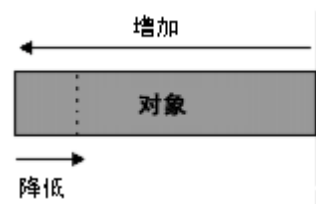
如果你选择**左侧**，所有的宽度变化将会在对象的右侧发生。（即左边的边线将保持不动）：



如果你选择**中间**，宽度变化将会在对象的两侧同等发生。（即中间轴将保持不动）：



如果你选择**右侧**，所有的宽度变化将会在对象的左侧发生。（即右边的边线将保持不动）：



[中心轴偏移] 自定义

点击此单选按钮来定义你自己的中心轴。对话框的右侧会显示一个域，您可以通过该域指定要在离对象中心多远（以像素为单位）的地方放置中间轴。尽管此选项给了你在对象任意处放置中间轴的自由，但一旦放置完毕，缩放过程与快速选项的工作方式完全相同（如上图所示）。

例如，如果你输入 20，中心轴将位于对象中心右侧距其 20 个像素的位置。

在域内输入负值以向左（不向右）移动中心轴。

注意：如果一个组和其中的对象被配置成可在运行时改变大小，组合缩放效果将会与对象缩放效果共同起作用。例如，如果一个组合被配置为在运行时变大一倍，而其中一个对象被配置为缩小大小一倍，则对象将会保持大小不变。但请注意，对象的位置可能会随组变大而改变。

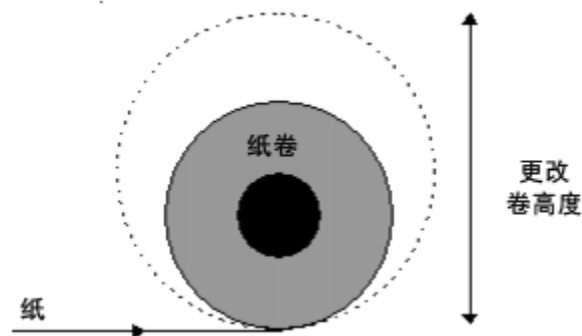
另请参见

[对象属性 - 缩放（垂直）](#)

对象属性 - 缩放（垂直）

您可以在运行时更改对象的高度。表达式的值增加或减少时，对象的高度将会根据原始高度的百分比相应地增加或减少；即当对象被加入图形画面时。

例如，一个造纸厂的卷纸机的侧视图能够显示纸卷的高度（和宽度）：



对象有以下垂直缩放属性：

缩放表达式

输入此域的表达式的值（最多 253 个字符）确定对象的垂直缩放（高度）。缺省情况下，当表达式返回最小值时，对象将会以其（定义在以下缩放域的）最小高度显示。当表达式返回最大值时，对象将会以其（定义在以下缩放域的）最大高度显示。

要插入标签或函数，点击此域右侧的**向导**按钮。此按钮显示两个选项：插入标签和插入函数。

[缩放表达式] 指定范围

选中此框可手动为缩放表达式指定**最小值**和**最大值**，而不使用缺省值。对于包含模拟变量标签的表达式，缺省值是表达式的最后一个变量标签中的工程零刻度和满刻度值。如果此模拟变量标签没有工程零刻度和满刻度值，那么缺省值是 0 (零) 和 32000。没有标签的表达式里缺省值为 0 (零) 和 100。

[缩放表达式] 最小值

输入表达式的最小值。当表达式返回此值时，对象的高度将会减小到它的最小值。如果已选中**指定范围框**，您只能在此输入一个值。

[缩放表达式] 最大值

输入表达式的最大值。当表达式返回此值时，对象的高度将会增加到它的最大值。如果已选中**指定范围框**，您只能在此输入一个值。

[缩放] 最小处

对象的最小高度 (依其原始高度的百分比)。当**缩放表达式**返回它的最小值时，对象将会被缩小至此高度。你可以按域右边的上下箭头或直接输入来修改百分比。可以输入大于 100% 的百分比值。

[缩放] 最大处

对象的最大高度 (依其原始高度的百分比)。当**缩放表达式**返回它的最大值时，对象将会被增大至此高度。

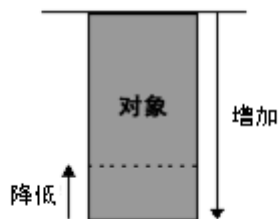
你可以按域右边的上下箭头或直接输入来修改百分比。可以输入大于 100% 的百分比值。

注意：您可以通过交换缩放域内的百分比值（即在**最小处**域内输入高百分比值，而在**最大处**域内输入低百分比值），或交换表达式极限值（在**最小值**和**最大值**域中）来在其为最小值时增加对象的高度和在其为最大值时减小对象的高度。

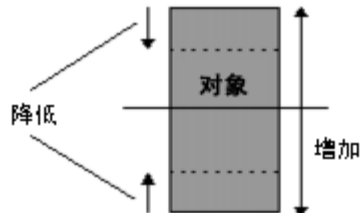
[中心轴偏移] 快速

点击此单选按钮选择对象的三个水平轴（顶部、中间、底部）中的一个。这些轴显示在对话框右侧的图形域内。点击一个轴将其选中。

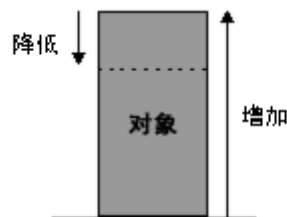
如果选择顶部，所有的高度变化将会在对象上从上向下发生。（即顶部边线将保持不动）：



如果选择中间，高度变化将会在轴的两侧同等发生。（即水平中间轴保持不动）：



如果选择底部，所有宽度变化将会在对象的上边发生。（即下底部边线将保持不动）：



[中心轴偏移] 定制

点击此单选按钮来定义您自己的中心轴。对话框的右侧会显示一个域，您可以通过该域指定要在离对象中心多远（以像素为单位）的地方放置轴。尽管此选项给了您在对象任意处放置中间轴的自由，但一旦放置完毕，缩放过程的工作方式与快速选项完全相同（如上图所示）。

例如，如果你输入 20，中心轴将位于对象中心下方距其 20 个像素的位置。

在域内输入负值以向上（非向下）移动中心轴。

注意：

- 如果某个组和其中的对象被配置为可在运行时改变大小，该组的缩放效果将会与对象缩放效果共同起作用。例如，如果一个组合被配置为在运行时变大一倍，而其中一个对象被配置为缩小大小一倍，则对象将会保持大小不变。但请注意，对象的位置可能会随组变大而改变。
- 当对象属性 - 填充颜色（开/关、多状态等）中的单选按钮被选中时，它们将改变对话框右侧的外观。

另请参见

[对象属性 - 缩放（水平）](#)

填充颜色

可以控制对象使用的填充颜色。

要配置包含变化的填充颜色的对象或组，请执行下列操作：

1. 绘制对象/组（或粘贴一个符号）。属性分页对话框将自动显示，除非您在图形编辑器中关闭了**新建对象时显示属性**选项。（如果是组，属性对话框不会自动显示；您必须双击该组）。
2. 点击**填充**分页。

3. 点击**颜色**分页（对话框右部）。
4. 选择颜色变化的类型（开/关、多状态等）。
5. 输入将在 **Runtime** 中更改对象填充颜色的表达式/条件。
6. 根据需要输入其它对象属性的详细信息。
7. 点击**确定**。

另请参见

[对象属性 - 填充颜色（开/关）](#)

[对象属性 - 填充颜色（多状态）](#)

[对象属性 - 填充颜色（阵列）](#)

[对象属性 - 填充颜色（阈值）](#)

[对象属性 - 填充颜色（梯度）](#)

对象属性 - 填充颜色（开/关）

对象和组有以下填充颜色（开/关）的属性：

[类型] 开/关

选择此单选按钮，可以在某个特定表达式为真时使用一种颜色填充对象/组，而为假时用另外一种颜色填充。例如，当特定变量标签报警时，您可以使用红色填充一个对象/组，而在不报警时使用绿色填充。

[类型] 多状态

如果在不同的时间，有多个条件以不同的组合发生，可以使用此选项。选择此选项，可以使用不同的颜色来填充每个组合的对象/组。

例如，三个数字量变量标签（A、B 和 C）在任意时间都可能是 ON 或 OFF 状态。您可以使用不同的颜色来填充每个 ON/OFF 组合的对象/组。也就是说，您可以使用不同的填充颜色来填充以下每个 ON/OFF ABC 组合：ABC、ABC、ABC、ABC、ABC、ABC、ABC。

[类型] 阵列

阵列选项允许您输入一个返回整数的表达式。对于每个唯一整数（从 0 到 255），您可以使用不同的颜色来填充对象/组。例如，您可以对每个模拟量报警的阈值使用一种不同的填充颜色。

[类型] 阈值

选择此单选按钮，可以在表达式到达一个指定的参数值（阈值）时动态地改变填充颜色。例如，您也许决定填充颜色应该在马达速度大于或等于 4500 rpm 时变成红色，而在小于或等于 100 rpm 时变成白色，但是当速度值介于二者中间时保持灰色。

[类型] 梯度

选择单选按钮，可以动态地渐变填充颜色，为特定表达式返回的每个唯一参数值显示一种不同的颜色。此选项允许您选择两种颜色作为颜色限制。每个返回的参数值所对应的颜色从这些颜色限制的范围中自动选取。结果是从一种颜色渐变成另外一种颜色。

ON 颜色条件

只要此处输入的条件（最多 128 个字符）为真时，就会选择使用 **ON 颜色**（见下）来作为填充颜色。只要条件为假时，就会选择使用 **OFF 颜色**（见下）来作为填充颜色。例如，当 **MIX_RUNNING** 为真时，您可以使用蓝色填充对象/组，而为假时，使用白色来填充。

如需插入标签或函数，点击字段右边的**向导**按钮。此按钮将显示两个选项：插入标签和插入函数。

OFF 颜色

当以上输入条件为假时的填充颜色。例如，当 **MIX_RUNNING** 为假时，您可以使用白色来填充对象/组。

注意： 您在此处选择的颜色将会改变在“外观（通用）”属性分页中指定的任何填充颜色。

ON 颜色

当以上输入条件为真时的填充颜色。例如，当 **MIX_RUNNING** 为真时，您可以使用蓝色来填充对象/组。

注意： 组的填充颜色只有在组内的单个对象没有定义其自身填充颜色时才适用。

另请参见

[对象属性 - 填充颜色（多状态）](#)

[对象属性 - 填充颜色（阵列）](#)

[对象属性 - 填充颜色（阈值）](#)

[对象属性 - 填充颜色（梯度）](#)

对象属性 - 填充颜色（多状态）

对象和组有以下填充颜色（多状态）属性：

[类型] 开关

选择此单选按钮，可以在某个特定表达式为真时使用一种颜色填充对象/组，而为假时用另外一种颜色填充。例如，当特定变量标签报警时，您可以使用红色填充一个对象/组，而在不报警时使用绿色填充。

[类型] 多状态

如果在不同的时间，有多个条件以不同的组合发生，可以使用此选项。选择此选项，可以使用不同的颜色来填充每个组合的对象/组。

例如，三个数字量变量标签（A、B 和 C）在任意时间都可能是 ON 或 OFF 状态。您可以使用不同的颜色来填充每个 ON/OFF 组合的对象/组。也就是说，您可以使用不同的填充颜色来填充以下每个 ON/OFF ABC 组合：**ABC、ABC、ABC、ABC、ABC、ABC、ABC。**

[类型] 阵列

阵列选项允许您输入一个返回整数的表达式。对于每个唯一的整数（从 0 - 255），您可以使用不同的颜色来填充对象/组。例如，您可以对每个模拟量报警的阈值使用一种不同的填充颜色。

[类型] 阈值

选择此单选按钮，可以在表达式到达一个指定的参数值（阈值）时动态地改变填充颜色。例如，您也许决定填充颜色应该在马达速度大于或等于 4500 rpm 时变成红色，而在小于或等于 100 rpm 时变成白色，但是当速度值介于二者中间时保持灰色。

[类型] 梯度

选择单选按钮，可以动态地渐变填充颜色，为特定表达式返回的每个唯一参数值显示一种不同的颜色。此选项允许您选择两种颜色作为颜色限制。每个返回的参数值所对应的颜色从这些颜色限制的范围中自动选取。结果是从一种颜色渐变成另外一种颜色。

条件

您在此输入的条件（每个条件最多使用 128 个字符）将以多种方式在不同时间同时发生。您可以使用每个唯一的组合，来向对象/组强制实施一个不同的填充颜色。

如需输入一个条件，可点击相关行（A、B、C 等），点击**编辑**，并输入条件。您可以使用**添加**按钮，添加更多的条件（最多 5 个，提供 32 个组合）。如需插入标签或函数，点击**向导**按钮。此按钮将显示两个选项：插入标签和插入函数。您也可以点击**删除**来删除条件，但是在字段中必须至少有一个条件。空白的条件（而不是被删除的条件）将在运行时被认为是假。

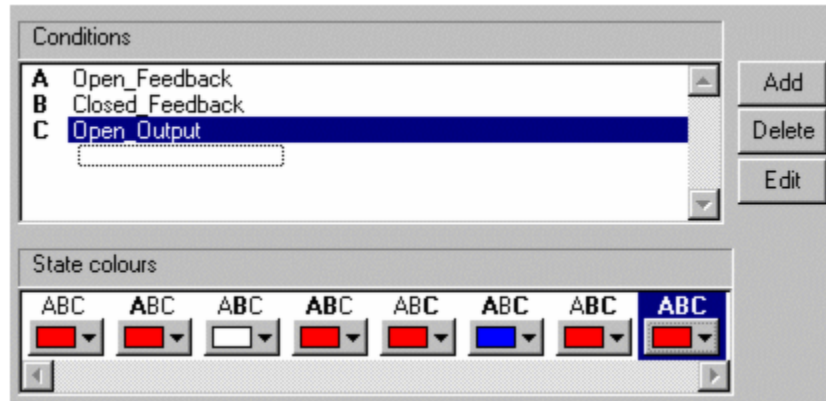
状态颜色

用于上述条件的每个组合的填充颜色。

注意： 您为 **ABC**（所有条件为假）而选的颜色将会改变在“外观（通用）”属性分页中指定的任何填充颜色。

例如：

每当阀门状态变化时，如果要使用不同的颜色来填充对象/组，您可以按如下方式填写**条件**和**状态符号**字段：



在此例子中，**Open_Feedback** 和 **Close_Feedback** 是代表阀门上数字量输入内容的变量标签，而 **Open_Output** 是代表阀门上输出内容的变量标签。因此，**ABC** 表示 **Open_Feedback** 为 ON 状态，而 **Close_Feedback** 和 **Open_Output** 都为 OFF 状态。针对这一组合，使用红色填充色来表示阀门不会相应命令，因为阀门在应该关闭时处于打开状态。相同的逻辑类型也同样适用于其它状态。

注意： 组的填充颜色只有在组内的单个对象没有定义其自身填充颜色时才适用。

另请参见

[对象属性 - 填充颜色（开/关）](#)

[对象属性 - 填充颜色（阵列）](#)

[对象属性 - 填充颜色（阈值）](#)

[对象属性 - 填充颜色（梯度）](#)

对象属性 - 填充颜色（阵列）

对象和组有以下颜色填充（阵列）属性：

[类型] 开/关

选择此单选按钮，可以在某个特定表达式为真时使用一种颜色填充对象/组，而为假时用另外一种颜色填充。例如，当特定变量标签报警时，您可以使用红色填充一个对象/组，而在不报警时使用绿色填充。

[类型] 多状态

如果在不同的时间，有多个条件以不同的组合发生，可以使用此选项。选择此选项，可以使用不同的颜色来填充每个组合的对象/组。

例如，三个数字量变量标签（A、B 和 C）在任意时间都可能是 ON 或 OFF 状态。您可以使用不同的颜色来填充每个 ON/OFF 组合的对象/组。也就是说，您可以使用不同的填充颜色来填充以下每个 ON/OFF ABC 组合：ABC、ABC、ABC、ABC、ABC、ABC、ABC。

[类型] 阵列

阵列选项允许您输入一个返回整数的表达式。对于每个唯一的整数（从 0 - 255），您可以使用不同的颜色来填充对象/组。例如，您可以对每个模拟量报警的阈值使用一种不同的填充颜色。

[类型] 阈值

选择此单选按钮，可以在表达式到达一个指定的参数值（阈值）时动态地改变填充颜色。例如，您也许决定填充颜色应该在马达速度大于或等于 4500 rpm 时变成红色，而在小于或等于 100 rpm 时变成白色，但是当速度值介于二者中间时保持灰色。

[类型] 梯度

选择单选按钮，可以动态地渐变填充颜色，为特定表达式返回的每个唯一参数值显示一种不同的颜色。此选项允许您选择两种颜色作为颜色限制。每个返回的参数值所对应的颜色从这些颜色限制的范围中自动选取。结果是从一种颜色渐变成另外一种颜色。

阵列表达式（128 个字符）

输入返回整数的表达式。对应每个返回参数值，会有一个不同的颜色来填充对象/组。

如果返回参数值是：

- 小于 0（零），将设置为 0（零），并将触发运行时硬件报警。
- 大于 255，将设置为 255，并将触发运行时硬件报警。
- 如果是实数（非整数），它将被截断（例如，8.1 和 8.7 都将被截断为 8）。

如需插入标签或函数，点击字段右边的**向导**按钮。此按钮将显示两个选项：插入标签和插入函数。

阵列颜色

用于填充上述阵列表达式所返回的每个整数的颜色（颜色 0 将在表达式返回整数 0 时使用，颜色 1 将在表达式返回整数 1 时使用，依此类推）。

注意： 您为颜色 0（零）选择的颜色将会改变任何在“外观” - “通用”分页中指定的填充颜色。

例如，要使用不同的符号来表示马达的不同的状态，您可以按如下方式填写**阵列的表达式**和**阵列符号**：



在此例中，**MOTOR_STATUS** 是一个模拟量变量标签，用于表示马达状态。每次马达改变状态时，都将返回一个整数（0 = 运行，1 = 启动，以此类推），并将相应的颜色用于填充对象/组。颜色 5 及其以上的颜色在颜色填充中不起作用，因为标签只返回 5 个唯一整数 (0-4)。

注意： 组的填充颜色只有在组内的单个对象没有定义其自身填充颜色时才适用。

另请参见

[对象属性 - 填充颜色（开/关）](#)

[对象属性 - 填充颜色（多状态）](#)

[对象属性 - 填充颜色（阈值）](#)

[对象属性 - 填充颜色（梯度）](#)

对象属性 - 填充颜色（阈值）

对象和组有以下颜色填充（阈值）属性：

[类型] 开/关

选择此单选按钮，可以在某个特定表达式为真时使用一种颜色填充对象/组，而为假时用另外一种颜色填充。例如，当特定变量标签报警时，您可以使用红色填充一个对象/组，而在不报警时使用绿色填充。

[类型] 多状态

如果在不同的时间，有多个条件以不同的组合发生，可以使用此选项。选择此选项，可以使用不同的颜色来填充每个组合的对象/组。

例如，三个数字量变量标签（A、B 和 C）在任意时间都可能是 ON 或 OFF 状态。您可以使用不同的颜色来填充每个 ON/OFF 组合的对象/组。也就是说，您可以使用不同的填充颜色来填充以下每个 ON/OFF ABC 组合：ABC、ABC、ABC、ABC、ABC、ABC、ABC。

[类型] 阵列

阵列选项允许您输入一个返回整数的表达式。对于每个唯一的整数（从 0 - 255），您可以使用不同的颜色来填充对象/组。例如，您可以对每个模拟量报警的阈值使用一种不同的填充颜色。

[类型] 阈值

选择此单选按钮，可以在表达式到达一个指定的参数值（阈值）时动态地改变填充颜色。例如，您也许决定填充颜色应该在马达速度大于或等于 4500 rpm 时变成红色，而在小于或等于 100 rpm 时变成白色，但是当速度值介于二者中间时保持灰色。

[类型] 梯度

选择此单选按钮，以动态地（平稳地）渐变填充颜色，为特定表达式返回的每个唯一参数值显示一种不同的颜色。此选项允许您选择两种颜色作为颜色限制。每个返回的参数值所对应的颜色从这些颜色限制的范围中自动选取。结果是从一种颜色渐变成另外一种颜色。

颜色表达式

在此字段中输入的表达式参数值（最多 128 个字符）决定了对象/组的填充颜色。即当表达式的参数值达到一个阈值（如下定义），填充的颜色将会改变。

如需插入标签或函数，点击字段右边的**向导**按钮。此按钮将显示两个选项：插入标签和插入函数。

[颜色表达式] 指定范围

选择此复选框，可手动指定颜色表达式的最小和最大参数值，而不使用缺省参数值。（阈值的范围介于**最小值**和**最大值**之间。）对于一个包含模拟变量标签的表达式，缺省的参数值为表达式中最后变量标签中的工程零刻度和满刻度值。如果此模拟变量标签没有工程零刻度和满刻度值，那么缺省值是 0（零）和 32000。对于没有标签的表达式，缺省值为 0（零）和 100。

[颜色表达式] 最小值

输入表达式的最小参数值。对于阈值来说，最小值是 0%。如果您选择**指定范围**复选框，则您只能在此输入一个参数值。

[颜色表达式] 最大值

输入表达式的最大参数值。对于阈值来说，最大值是 100%。如果您选择**指定范围**复选框，则您只能在此输入一个参数值。

阈值 (%)

阈值及其相关颜色。阈值是作为表达式范围（表达式可返回参数值的范围）的一个百分数输入。例如，如果表达式的最小值是 0，而其最大值是 200，缺省的阈值有如下效果：

阈值	相关颜色	意义
< 5%	浅蓝	当表达式返回参数值小于 10，填充颜色为浅蓝。

阈值	相关颜色	意义
< 15%	蓝	当表达式返回参数值小于 30 ，填充颜色为 蓝 。
> 85%	红	当表达式返回参数值大于 170 ，填充颜色为 红 。
> 95%	浅红	当表达式返回参数值大于 190 ，填充颜色为 浅红 。

你可以添加最多 100 个阈值颜色组合。如需添加组合，点击**添加**并输入相关详细信息。如需编辑已有的组合，点击相关行。您也可以通过点击**删除**来删除组合。

任何不在范围内的参数值（如上例中 15% 和 85% 之间的值）将会按照**外观 - 通用**分页中的指定，生成一个静态填充颜色。

注意： 组的填充颜色只有在组内的单个对象没有定义其自身填充颜色时才适用。

另请参见

[对象属性 - 填充颜色（开/关）](#)

[对象属性 - 填充颜色（多状态）](#)

[对象属性 - 填充颜色（阵列）](#)

[对象属性 - 填充颜色（梯度）](#)

对象属性 - 填充颜色（梯度）

对象和组有以下填充颜色（梯度）属性：

[类型] 开/关

选择此单选按钮，可以在某个特定表达式为真时使用一种颜色填充对象/组，而为假时用另外一种颜色填充。例如，当特定变量标签报警时，您可以使用红色填充一个对象/组，而在不报警时使用绿色填充。

[类型] 多状态

如果在不同的时间，有多个条件以不同的组合发生，可以使用此选项。选择此选项，可以使用不同的颜色来填充每个组合的对象/组。

例如，三个数字量变量标签（A、B 和 C）在任意时间都可能是 ON 或 OFF 状态。您可以使用不同的颜色来填充每个 ON/OFF 组合的对象/组。也就是说，您可以使用不同的填充颜色来填充以下每个 ON/OFF ABC 组合：**ABC、ABC、ABC、ABC、ABC、ABC、ABC**。

[类型] 阵列

阵列选项允许您输入一个返回整数的表达式。对于每个唯一整数（从 0 到 255），您可以使用不同的颜色来填充对象/组。例如，您可以对每个模拟量报警的阈值使用一种不同的填充颜色。

[类型] 阈值

选择此单选按钮，可以在表达式到达一个指定的参数值（阈值）时动态地改变填充颜色。例如，您也许决定填充颜色应该在马达速度大于或等于 4500 rpm 时变成红色，而在小于或等于 100 rpm 时变成白色，但是当速度值介于二者中间时保持灰色。

[类型] 梯度

选择单选按钮，可以动态地渐变填充颜色，为特定表达式返回的每个唯一参数值显示一种不同的颜色。此选项允许您选择两种颜色作为颜色限制。每个返回的参数值所对应的颜色从这些颜色限制的范围中自动选取。结果是从一种颜色渐变成另外一种颜色。

颜色表达式

在此字段中输入的表达式参数值（最多 128 个字符）决定了对象/组的填充颜色。缺省情况下，当表达式返回最小值参数时，填充颜色将会是**最小**（定义如下）。当表达式返回最大参数值时，填充颜色将会是**最大**（定义如下）。当表达式返回一个中间参数值时，填充颜色将会从介于**最小**和**最大**之间的范围中选择。

如需插入标签或函数，点击字段右边的**向导**按钮。此按钮将显示两个选项：插入标签和插入函数。

[颜色表达式] 指定范围

选择此复选框，可手动指定颜色表达式的最小和最大参数值，而不使用缺省参数值。对于一个包含模拟变量标签的表达式，缺省的参数值为表达式中最后变量标签中的工程零刻度和满刻度值。如果此模拟变量标签没有工程零刻度和满刻度值，那么缺省值是 0（零）和 32000。对于没有标签的表达式，缺省值为 0（零）和 100。

[颜色表达式] 最小值

输入表达式的最小参数值。当表达式返回此参数值时，对象/组的填充颜色将会是**最小**。如果您选择**指定范围**复选框，则您只能在此输入一个参数值。

[颜色表达式] 最大值

输入表达式的最大参数值。当表达式返回此参数值时，对象/组的填充颜色将会是**最大**。如果您选择**指定范围**复选框，则您只能在此输入一个参数值。

最小

颜色表达式返回其最小参数值时，对象/组的填充颜色。

注意： 您在此处选择的颜色将会改变在**外观 - 通用**分页指定的任何填充颜色。

最大

颜色表达式返回其最大值时，对象/组的填充颜色。

注意： 组的填充颜色只有在组内的单个对象没有定义其自身填充颜色时才适用。

另请参见

[对象属性 - 填充颜色（开/关）](#)

[对象属性 - 填充颜色（多状态）](#)

[对象属性 - 填充颜色（阵列）](#)

[对象属性 - 填充颜色（阈值）](#)

填充高度

可以控制对象中显示的填充高度。

要配置变化填充高度的对象或组，请执行下列操作：

1. 绘制对象/组（或粘贴一个符号）。属性分页对话框将自动显示，除非您在图形编辑器中关闭了**新建对象时显示属性**选项。（如果是组，属性对话框不会自动显示；您必须双击该组）。
2. 点击**填充**分页。
3. 点击**高度**分页（对话框右部）。
4. 输入**方向表达式**（表达式将更改 Runtime 中对象/组的填充高度）。
5. 根据需要输入其它属性。
6. 点击**确定**。

另请参见

[对象属性 - 填充（高度）](#)

[定义通用对象属性](#)

对象属性 - 填充（高度）

对象/组的填充高度可以在运行时变化，只要某个特定表达式的参数值被更改，填充高度就会动态地增加或减小。当表达式的参数值增加或减少时，填充高度也将随之增加或减少（以对象/组容量的百分比率）。如果对象/组的大小在运行时变化，则填充高度将会自动地调节，以保持正确的百分比。

使用的颜色是通过“通用外观”或“颜色填充”而设置的。

此属性可用于显示温度的变化。您甚至可以结合“填充颜色”和“填充高度”属性，生成一个水银温度计，使水银随温度的升高而升高并改变颜色。

对象和组有以下“填充高度”属性：

高度表达式

在此字段中输入的expressions的参数值（最多 253 个字符）决定了对象/组的填充高度。缺省情况下，如果表达式返回最小参数值，则对象/组的填充高度将会是**最小**。如果表达式返回最大参数值，则对象/组的填充高度将会是**最大**。如果表达式返回一个中间参数值，则对象/组的填充高度将会介于**最小**和**最大**之间。

如需插入标签或函数，点击字段右边的**向导**按钮。此按钮将显示两个选项：插入标签和插入函数。

[高度表达式] 指定范围

选择此复选框，可手动指定高度表达式的最小和最大参数值，而不使用缺省参数值。对于一个包含模拟变量标签的表达式，缺省的参数值为表达式中最后变量标签中的工程零刻度和满刻度值。如果此模拟变量标签没有工程零刻度和满刻度值，那么缺省值是 0（零）和 32000。对于没有标签的表达式，缺省值为 0（零）和 100。

[高度表达式] 最小值

输入表达式的最小参数值。当表达式返回此参数值时，对象/组的填充高度将会是**最小**。如果您选择**指定范围**复选框，则您只能在此输入一个参数值。

[高度表达式] 最大值

输入表达式的最大参数值。当表达式返回此参数值时，对象/组的填充高度将会是**最大**。如果您选择**指定范围**复选框，则您只能在此输入一个参数值。

最小

方向表达式返回其最小参数值时，对象/组的填充高度。例如，如果您输入 30，则对象/组将会在表达式返回其最小参数值时填充高度 30%。

您可以通过按字段右边的上下箭头或输入其他参数值来更改此百分比值。

最大

方向表达式返回其最大参数值时，对象/组的填充高度。例如，如果您输入 90，对象/组将会在表达式返回其最大值时填充高度 90%。

您可以通过按字段右边的上下箭头或输入其他参数值来更改此百分比值。

填充方向

当填充高度增加时颜色扩展的方向。有四个选项（每个由一个箭头代表）：**上**、**下**、**左**、**右**。如果您选择“上”，对象/组将会从下至上地填充。如果您选择“左”，对象/组将会从右至左地填充，以此类推。

背景颜色

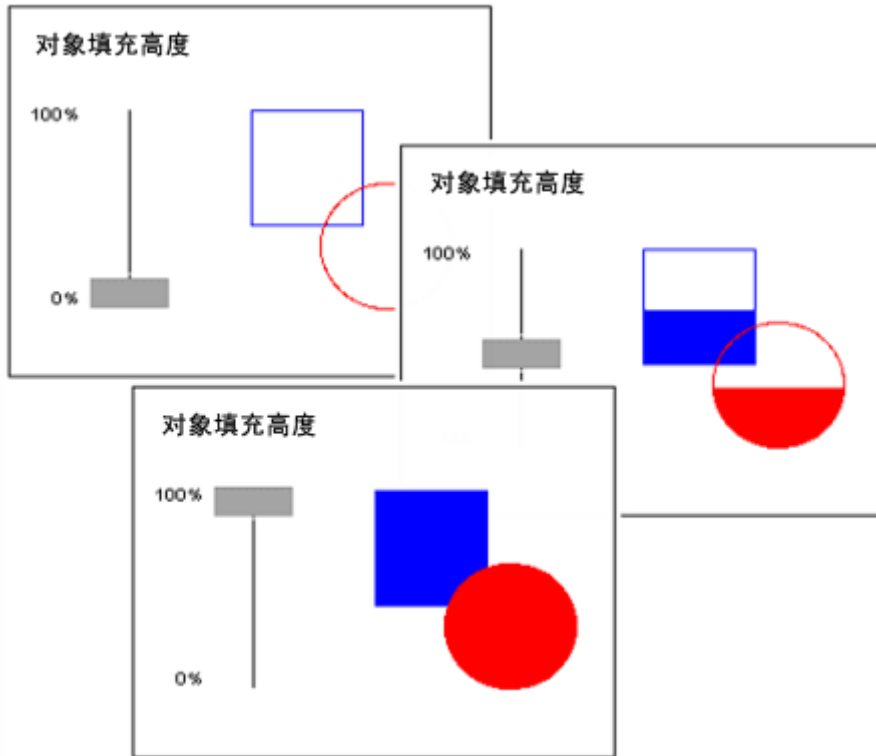
对象/组中未被填充部分的颜色（例如，如果对象/组只有 90%，那么未被填充的 10% 将会用此颜色来显示）。背景通常是透明的。使用透明色，您可以看见对象/组的轮廓，以及画面上对象/组后面的所有内容。

注意： 如果某个组内有一个滑钮对象，则在运行使用时，它可以改变组的整体大小。如果这样，将会相应地调节组的填充高度。

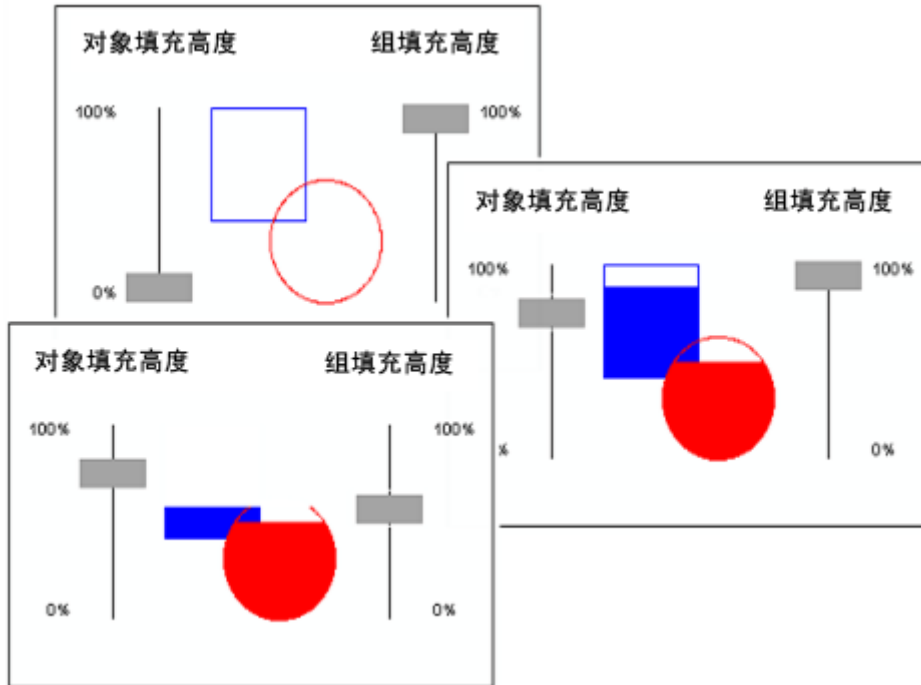
组和对象的填充高度：示例

一个组及其对象可以用不同的填充高度来配置。但是，组的填充高度通常是用于表示组中对象的填充高度。组的填充高度和对象的填充高度相互独立，组的填充高度仅能确定对象的显示程度。

示例 1： 对象的填充高度：



示例 2： 对对象进行分组，并为该组配置一个填充高度



在此示例中，仍然能够正常调节对象的填充高度。组的填充高度可以确定您能够看见的对象的显示程度（以及有多少被组的背景色所覆盖，此例中的背景色为白色）。

触击命令

可以为某个对象或组指定触击命令。

如何赋予对象或组触击命令：

1. 绘制相应的对象/组（或粘贴一个符号）。属性分页对话框将自动显示，除非您在图形编辑器中关闭了**新建对象时显示属性**选项。（如果是组，属性对话框不会自动显示；您必须双击该组）。
2. 点击**输入**分页。
3. 点击**触击**分页（对话框右部）。
4. 在命令字段内输入一个命令（一个能在运行时对象/组被触击时执行的命令）。
5. 根据需要进行更详细新的信息，使用**帮助**按钮可获得有关每个字段的详细信息。
6. 单击**确定**。

另请参见

[对象属性 - 输入（鼠标）](#)

[定义通用对象属性](#)

对象属性 - 输入（鼠标）

鼠标属性使您能够向对象/组赋予命令。在运行时点击对象/组时（即操作员点击对象/组），将执行这些命令。您也可以定义在此时记录的消息。

例如，可以通过按下鼠标键来启动驱动器，或放开鼠标键来停止驱动器；按住鼠标键可以增加变量，诸如此类。同时，时间和日期，以及操作员的名字能够被记录下来。

不满足访问要求的操作员无法使用鼠标在运行时点击对象/组。

对象和组有以下输入（鼠标）属性：

动作

有三种命令动作。您可以选择多种动作类型。每个动作可以有其自己唯一的命令和记录消息（即您可以在鼠标键按下时执行一个任务，而弹起时执行另外一个任务，并且分别记录每个动作的消息）。

[动作] 弹起

选择此选项，当操作员将鼠标指针放在对象/组上，点击并释放鼠标左键时，将执行命令（并记录唯一的消息）。

使用标准的 Windows 按钮，如果操作员在释放鼠标按键前，将光标从对象/组上移开，将不会执行命令（除非您也选择了**按下**选项）。

[动作] 按下

选择此选项，当操作员将鼠标指针放在对象/组上，点击鼠标左键时，将执行命令（并记录唯一的消息）。点击鼠标键后，将立即执行命令。

[动作] 重复

选择此选项，当操作员将鼠标指针放在对象/组合上，按住鼠标左键时，将持续执行命令（并连续记录唯一的消息）。如果操作员将鼠标指针从对象/组上移开，而没有放开鼠标键，命令将停止执行，但是会在鼠标指针重新放回对象/组上时立即再次执行。唯一的例外是，当您同时选择了“按下”选项时，即使鼠标指针从对象/组上移开，也将持续执行命令。

设置第一次执行命令（和第一次记录消息）前的延迟，以及每个重复执行间的延迟。

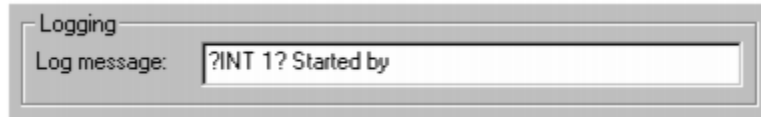
弹起/按下/重复命令

当执行所选的动作时立即执行的命令（系列指令）。命令的长度最多为 253 个字符。

如需插入标签或函数，点击字段右边的**向导**按钮。此按钮将显示两个选项：插入标签和插入函数。

[日志] 日志消息

操作员在运行时间执行所选的操作时，向记录设备的 *MsgLog* 字段发送的文本消息。此消息是纯文本的，而且可以包含使用精灵或超级精灵语法替代标签名。当使用超级精灵语法时，必须指定数据类型。文本中必须包含标签名称。日志消息的长度最多为 32 个字符。



如需将字段数据作为记录消息的一部分，可在配置设备时，在设备格式中插入字段名称。例如，在**设备**视窗的**格式**字段中，可以输入 {MsgLog,20} {FullName,15}。这将包含如下的消息记录：**P2 started by John Smith.**

消息发送到的日志设备是在“通用访问”分页内指定的。

注意： 如果对象属于某个精灵或符号，则可以在精灵/符号被粘贴到画面上后，定义此属性。（按住 **Control (CTRL)** 键并双击对象。）如果您在粘贴前对其进行定义（即您为库中的原形定义），您就不能在此之后对其进行编辑。同样，如果对象是模板的一部分，可以在使用模板创建画面后，定义对象（同样按住 **Control + 双击**）。如果您在模板中定义，就不能在以此模板为基础创建的画面上编辑它。

重复频率

使用此选项，可设置第一次执行命令前的延迟，和相邻的重复命令间的延迟。

您可以按字段右边的上下箭头，或直接在字段中输入另一个参数值来更改此频率。

注意： 如果您为某个组中的对象定义了一个鼠标命令，则组的鼠标命令将不起作用。

键盘命令

可以为一个或一组对象指定一个键盘命令。

为一个或一组对象指定键盘命令：

1. 绘制相应的对象/组（或粘贴一个符号）。对象属性分页对话框将自动显示，除非您在图形编辑器中关闭了**新建对象时显示属性**选项。（如果是组，属性对话框不会自动显示；您必须双击该组）。
2. 点击**输入**分页。
3. 点击（位于对话框右边的）**键盘命令**分页。
4. 输入键序列。
5. 在命令字段中输入一个命令（一个能在运行时当操作员输入上述键序列时执行的命令）。
6. 根据需要输入其他详细信息。
7. 点击**确定**。

另请参见

[对象属性 - 输入（键盘命令）](#)
[定义通用对象属性](#)

对象属性 - 输入（键盘命令）

键盘命令属性使您能够将键盘命令赋予对象/组。键盘命令是一组特别的键序列，当操作员在运行时键入命令时，执行该命令。如需执行一个对象/组键盘命令，操作员应将光标放在对象/组合上，然后使用键盘输入键序列。

您也可以定义每次输入键序列时记录的消息。

例如，您可以将鼠标放在表示储存罐的符号上，键入新的水位高度，从而更改储存罐中的水位。同时，将记录消息，列出时间和日期以及操作员的姓名。

如果操作员不满足在下面的**安全性**里指定的访问要求，将不能在运行时为对象/组输入键盘命令。

对象和组有以下键盘命令输入属性：

键序列

输入键序列，操作员可以使用键序列来执行命令。例如，您可以定义键序列 **### Enter**。在运行时，此键序列将允许您输入一个任意三位数，然后按 **Enter** 键，以从画面上更改变量标签的参数值。

您可以输入任意多的键序列。如需添加键序列，点击**添加**，并键入键序列，或从菜单中选择。如需编辑已有的序列，点击相关行，然后点击**编辑**。您也可以通过点击**删除**来删除键序列。

键序列命令

输入所选键序列时马上执行的命令（系列指令）。命令的长度最多为 253 个字符。

如需插入标签或函数，点击字段右边的**向导**按钮。此按钮将显示两个选项：插入标签和插入函数。

[安全] 等同对象/组区域

选择此复选框，可以将键盘命令赋予等同对象/组区域。只有能够访问此区域的（有所需的权限的）用户能够发出此命令或记录消息。如果您希望将键盘命令赋予其他区域，请不要选择此复选框；而是在下面输入另外一个区域。

[安全] 命令区域

输入此键盘命令所属的区域。只有能够访问此区域的（有所需的权限的）用户能够发出此命令或记录消息。例如，如果在这里输入了区域 1，那么操作员必须具有访问区域 1 的权限（以及所有必需的权限）才可以发布命令。

点击此字段右边的菜单，选择一个区域，或直接输入一个区域号。

注意： 如果对象属于某个精灵或符号，则可以在精灵/符号被粘贴至画面后（**Ctrl + 双击**），定义此属性。同样，如果对象属于某个模板，可以在使用该模板（**Ctrl + 双击**）创建画面后，对属性进行定义。

您可以将此字段留空，方法是选择**等同对象/组权限**复选框。

[安全] 等同对象/组权限

选择此复选框，可以将键盘命令赋予等同对象/组权限。只有具有此权限级别的用户能够发出此命令或记录消息。如果您希望赋予键盘命令不同的权限，请不要选择此复选框；而是在下面输入另一个权限。

[安全] 权限等级

输入用户能够发出此命令或记录消息所必须具备的权限级别。例如，如果您输入权限级别 1，则操作员必须拥有权限级别 1 才能发出此命令。您也可以将此限制与区域限制组合起来使用。例如，如果您向键盘命令赋予了区域 5，权限等级 2，则用户必须拥有权限等级 2 才能访问区域 5。

点击此字段右边的菜单，选择一个权限，或直接输入一个区域号。

注意： 如果对象属于某个精灵或符号，则可以在精灵/符号被粘贴至画面后（**Ctrl + 双击**），定义此属性。同样，如果对象属于某个模板，可以在使用该模板（**Ctrl + 双击**）创建画面后，对属性进行定义。

您可以将此字段留空，方法是选择**等同对象/组权限**复选框。

[日志] 日志消息

操作员在运行时间执行所选的操作时，向记录设备的 *MsgLog* 字段发送的文本消息。此消息是纯文本的，而且可以包含使用精灵或超级精灵语法替代标签名。当使用超级精灵语法时，必须指定数据类型。文本中必须包含标签名称。消息的长度最多为 32 个字符。



如果您希望将字段数据作为记录消息的一部分，则必须在配置设备时，在设备格式中插入字段名称。例如，在**设备视窗**的**格式**字段中，可以输入 {MsgLog,20} {FullName,15}。这将包含如下的消息记录：**P2 started by John Smith**。

消息发送到的日志设备是在“通用访问”分页内指定的。

注意： 如果某个组及其对象都被赋予一个具有相同键序列的键盘命令，则对象的命令优先级高（即不执行组的命令）。

滑钮

您可以创建要在图形页面上显示的滑钮。

要配置滑钮，请执行以下操作：

1. 绘制相应的对象/组（或粘贴一个符号）。属性分页对话框将自动显示，除非您在图形编辑器中关闭了**新建对象时显示属性**选项。（如果是组，属性对话框不会自动显示；您必须双击该组）。
2. 点击**滑钮**分页。
3. 点击**水平**、**垂直**或**旋转**分页（位于对话框右侧）。
4. 将**标签**输入指向滑钮的链接。
5. 输入所有附加详细信息。
6. 单击**确定**。

另请参见

[对象属性 - 滑动（水平）](#)

[对象属性 - 滑动（垂直）](#)

[对象属性 - 滑动（旋转）](#)

对象属性 - 滑动（水平）

可以通过将对象和组链接到变量标签来实现对象/组的水平滑动能改变标签值的效果。当滑钮向右移动时，变量标签值增大。当滑钮向左移动时，变量标签值减小。滑钮也可以自动地移动以反映标签值的变化。

注意： 如果旋转滑钮或水平移动已被启用，则不能使用水平滑钮。

对象和组具有以下水平滑钮属性：

标签

在此域输入的标签的值（最多 79 个字符）将会随着滑钮被向左和右的移动而改变。您可以在图形画面上定义两个滑钮边界。对象/组将不会滑出这两点。在运行时，当滑钮达到它的左手边界时（**最小处偏移**），标签值改变至其最小值。当滑钮达到它的右手边界时（**最大处偏移**），标签值改变至其最大值。

要插入标签，点击此域右侧的**向导**按钮。

[标签] 连续更新标签

如果希望变量标签在滑钮移动期间连续地更新，请选中此框。如果不选中此框，标签值将只在滑钮被释放（即它被移动了，并且操作员放开了鼠标键）时更新。

[偏移] 最小处

对象/组能够向左滑动的距离（距原始对象/组中心的像素数）。当它达到以此距离定义的点时，标签值变化至其最小值。

您可以按域右边的上下箭头或直接输入来修改偏移值。

[偏移] 最大处

对象/组能够向右滑动的距离（距原始对象/组中心的像素数）。当它达到以此距离定义的点时，标签值变化至其最大值。

您可以按域右边的上下箭头或直接输入来修改偏移值。

您可以通过在偏移域内输入负的距离来向左滑动以增加标签的值，和向右滑动以减少标签的值。

注意：如果组中有滑钮对象，它可能会在运行时更改被使用的组的整体大小。如此，组的填充水平将会相应的调节。如果某组和其中的一个对象同时被定义成滑钮，并且它们滑动的方向一致或者其中一个是旋转的，对象将具有优先权（即只有对象会以滑钮方式显示）。

另请参见

[对象属性 - 滑动（垂直）](#)

[对象属性 - 滑动（旋转）](#)

对象属性 - 滑动（垂直）

可通过将对象和组链接到变量标签来实现对象/组的垂直滑动能改变标签值的效果。当滑钮向上移动时，变量标签值增大。当滑钮向下移动时，变量标签值减小。滑钮也可以自动地移动以反映标签值的变化。

注意：如果旋转滑钮或垂直移动已被启用，则不能使用垂直滑钮。

对象和组具有以下垂直滑钮属性：

标签

在此域输入的标签的值（最多 79 个字符）将会随着滑钮被向上和向下的移动而改变。您可以在图形画面上定义两个滑钮边界。对象/组将不会滑出这两点。在运行时，当滑钮达到它的下边界时（**最小处偏移**），标签值改变至其最小值。当滑钮达到它的上边界时（**最大处偏移**），标签值改变至其最大值。

要插入标签，点击此域右侧的**向导**按钮。

[标签] 连续更新标签

如果希望变量标签在滑钮移动期间连续地更新，请选中此框。如果不选中此框，标签值将只在滑钮被释放（即它被移动了，并且操作员放开了鼠标键）时更新。

[偏移] 最大处

对象/组能够向上滑动的距离（距原始对象/组中心的像素数）。当它达到以此距离定义的点时，**标签值**变化至其最大值。

您可以按域右边的上下箭头或直接输入来修改偏移值。

[偏移] 最小处

对象/组能够向下滑动的距离（距原始对象/组中心的像素数）。当它达到以此距离定义的点时，标签值变化至其最小值。

您可以按域右边的上下箭头或直接输入来修改偏移值。

您可以通过在偏移域内输入负的距离来向下滑动以增加标签的值，和向上滑动以减少标签的值。

注意：如果组中有滑钮对象，它可能会在运行时更改被使用的组的整体大小。如此，组的填充水平将会相应的调节。如果一个组和其中的一个对象同时被定义成滑钮，并且它们滑动的方向一致或者其中一个是旋转的，对象将具有优先权（即只有对象会以滑钮方式显示）。

另请参见

[对象属性 - 滑动（水平）](#)

[对象属性 - 滑动（旋转）](#)

对象属性 - 滑动（旋转）

可通过将对象和组链接到变量标签来实现对象/组的旋转滑动能改变标签值的效果。当滑钮顺时针转动时，变量标签值增大。当滑钮逆时针转动时，变量标签值减小。滑钮也可以自动地移动以反映标签值的变化。

注意：如果另两种滑钮中任一种滑钮或者旋转移动被启用，则不能使用旋转滑钮。

对象和组具有以下旋转滑钮属性。

标签

在此域输入的标签的值（最多 79 个字符）将会随着滑钮的旋转而改变。您可以在图形画面上定义两个滑钮边界。对象/组的旋转将不会超出这两点。在运行时，当滑钮达到它的逆时针边界时（**最小处偏移**），标签值改变至其最小值。当滑钮达到它的顺时针边界时（**最大处偏移**），标签值改变至其最大值。

要插入标签，点击此域右侧的**向导**按钮。

[标签] 连续更新标签

如果希望变量标签在滑钮移动期间连续地更新，请选中此框。如果不选中此框，标签值将只在滑钮被释放（即它被移动了，并且操作员放开了鼠标键）时更新。

[角度] 最小处

输入一个逆时针角度（相对于 0 的角度）。滑钮不能逆时针旋转超出此边界。当其达到此边界时，**标签**值改变至其最小值边界。

您可以按域右边的上下箭头或直接输入来修改角度值。

[角度] 最大处

输入一个顺时针角度（相对于 0 的角度）。滑钮不能顺时针旋转超出此边界。当其达到此边界时，**标签**值改变至其最大值边界。

您可以按域右边的上下箭头或直接输入来修改角度值。

注意：您可以通过在角度域内输入负的距离来逆时针旋转以增加标签的值，和顺时针旋转以减少标签的值。

[中心轴偏移] 快速

点击此单选按钮能快速简单地选择对象/组旋转所围绕的点。快速选项为您提供有 9 点的选择（左上，右下等），这些点在对话框图形域内有显示。要选择一个，只需用鼠标点击即可。

[中心轴偏移] 定制

点击此单选按钮来定义您自己的中心轴。当您选择此选择按钮，右边将显示两个域以允许您画定您的中心轴位置。在第一个域内指定向右的距离，而在第二个内指定向下的距离。中心轴就是以这两个值来画的。

例如，如果你输入 8 为水平偏移，13 为垂直偏移，则中心轴将会在对象/组的中心向右 8 个像素，向下 13 个像素。

在偏移距离域内输入负值以将中心轴向左而不是向右，向上而不是向下移动。

注意：如果一个组合和其中的一个对象同时被定义成滑钮，并且它们滑动的方向一致或者其中一个是旋转的，对象将具有优先权（即只有对象会以滑钮方式显示）。

另请参见

[对象属性 - 滑动（水平）](#)

[对象属性 - 滑动（垂直）](#)

访问

您可以确定需要具备的对象访问权限种类。

- [对象的通用访问](#)
- [禁止访问对象](#)

对象的通用访问

可使用“通用”标签定义对象的通用访问特性。

要定义对象的通用访问属性，请执行以下操作：

1. 双击对象。
2. 点击**访问**标签。
3. 点击**通用**标签。
4. 输入所需数据，然后点击**确定**。

另请参见

[对象属性 - 访问 \(通用\)](#)
[定义通用对象属性](#)

对象属性 - 访问 (通用)

使用“对象属性”对话框，您可以为您的 ActiveX 对象设置通用的标识、安全、登录和权限参数。

对象和组有以下通用访问属性：

[标识] 对象/组 AN

显示对象/组的动态点号。AN 唯一标识了对象/组，并且可在 `Cicode` 等函数中使用。

注意： 如果对象属于某个精灵或符号，则可以在精灵/符号被粘贴至画面后 (**Ctrl + 双击**)，完成以下属性。同样，如果对象属于某个模板，可以在使用该模板 (**Ctrl + 双击**) 创建画面后，对属性进行定义。

[标识] 描述

输入对象/组及其各个函数的描述。此字段完全用于输入信息，这些信息应有利于系统的平稳运行和维护。它将不会影响系统的运行方式，而且在运行时不会显示。

[标识] 工具提示

输入一个简洁明了的对象/组的描述（最多 48 个字符）。运行时，当操作员将光标移动到对象/组的上方时，将显示此描述。该信息可以是纯文本，超级精灵语法，或者是两者的结合。如果使用超级精灵语法，则必须包含数据类型。文本中必须包含标签名称。

如果某个组中的某个对象有一个工具提示，将显示此对象的工具提示，而不会显示组的工具提示。如果对象没有工具提示，则将显示组的工具提示。在这种情况下，如果对象是组的成员，而此组又属于另外一个组，则将显示第一个组的工具提示。

如果您将对象置于组之后，则将不会显示其工具提示。但是，请记住，不论组中的对象是什么样的形状，组的边界都是长方形。这也就是说，组中对象间的“空白”空间，实际上是组的一部分。即使您能够看见某个对象，但如果它是在组的后面，也不会显示其工具提示。

[安全] 等同画面区域

选中此选项，可以赋予对象/组等同画面区域；或者，不选择此选项，而在**对象/组区域**字段（见下）中输入另外一个区域值。

[安全] 对象/组区域

输入对象/组所属的区域。不能访问此区域（和不具有所需权限）的用户将不能充分利用对象/组。这些用户不能使用鼠标命令、键盘命令、平移、滑钮等。（为了避免对操作员造成混淆，如果因为缺乏安全权限而无法访问对象/组时，最好将其禁止。被禁止的对象/组，其状态可以是变灰、隐藏或浮雕。）例如，如果您在此输入区域 1，则操作员必须具有访问区域 1 的权限（以及所有所需的权限），才能使用此对象/组。

点击此字段右边的菜单，选择一个区域，或直接输入一个区域号。

[安全] 无权限限制

选择此选项，可以禁止权限限制；或者，不选择此选项，并在下面输入其他权限。不赋予权限限制的意义在于您的安全设置中是否使用了区域：

- **无区域：**所有操作员能够完全控制对象/组。
- **有区域：**如果对象/组没有权限限制，操作员只需查看访问权限就可控制对象/组。

[安全] 权限等级

输入操作员使用此对象/组所需的权限等级。不具有所需权限的操作员将不能完全使用对象/组。这些操作员不能使用鼠标命令、键盘命令、平移、滑钮等。（为了避免对这些操作员造成混淆，如果因为缺乏安全权限而无法访问对象/组时，最好将其禁止。被禁止的对象/组，其状态可以是变灰、隐藏或浮雕。）例如，如果您输入权限等级 1，则操作员必须拥有权限等级 1 才能够使用此对象/组。您也可以将此限制与区域限制组合起来使用。例如，如果您赋予了对象/组区域 5，和权限等级 2，则用户必须拥有权限等级 2 才能访问区域 5。

点击此字段右边的菜单，选择一个权限，或直接输入一个权限号。

注意： 如果对象属于某个组，则用户必须能够访问该组，才能访问此对象。

[日志] 日志设备

此设备用于记录对象/组的键盘命令和鼠标命令的消息。点击此字段右边的菜单，选择一个设备，或输入一个设备名。

注意： 必须包含 MsgLog 字段，格式为日志设备发送消息的格式。

另请参见

[禁止访问对象](#)

禁止访问对象

如果需要，可以禁止访问对象。

如何禁止访问对象：

1. 双击对象。
2. 点击访问分页。

3. 点击**禁止**分页。
4. 指定禁止对象的条件和当对象被禁止时的外观。
5. 点击**确定**。

另请参见

[对象属性 - 访问（禁用）](#)
[定义通用对象属性](#)

对象属性 - 访问（禁用）

对象和组有以下访问（禁止）属性：

禁止条件

只要在此输入的表达式（最大为 128 个字符）为真，对象/组都将被禁止。如果对象/组被禁止，操作员将不能使用任何形式进行输入，如滑钮、鼠标命令、键盘命令等。

如需插入标签或函数，点击字段右边的**向导**按钮。此按钮将显示两个选项：**插入标签**和**插入函数**。

由三种方式可以表示某个对象/组已被禁止：浮雕、变灰和隐藏。

[禁止条件] 区域或权限不足时禁止

如果操作员的区域和权限不能满足其在“访问”中定义的要求，则将其禁止对象/组。

禁止风格：

- **浮雕** - 当被禁止时，对象/组的外观好象是图形画面上的浮雕。
- **变灰** - 当被禁止时，对象/组将变成灰色（不会显示所有的颜色详情）。
- **隐藏** - 当被禁止时，对象/组将完全隐藏，无法查看。

注意： 如果某个组被禁止，则组中的所有对象都将被禁止。

Chapter: 22 定义命令和控件

操作员可以通过命令与 Runtime 系统交互。可以定义三种类型的直接命令控件：

- [触击命令](#)，操作员通过点击特定图形对象（显示在图形画面上）。
- [键盘命令](#)，操作员通过在键盘上键入指令发布的命令。
- [滑动钮控件](#)，操作员使用它更改模拟变量的值。

您可以向所有命令和控件赋予权限，并在每次操作员发布命令时向命令纪录发送一条信息。

触击命令

你可以为在画面上创建的对象赋予点击命令。触击命令使操作员可以通过图形页面上的某个对象传送命令给运行时间系统。（对于按钮，可以通过键盘上的光标键加亮按钮和按住回车键执行命令。）

可以对一个对象定义多个命令，操作员单击对象时执行一个命令，操作员放开鼠标按钮时执行另一个命令，而按住鼠标按钮时连续地执行另一个命令。例如，可以通过按下鼠标键来启动驱动器，或放开鼠标键来停止驱动器；按住鼠标键可以增加变量，诸如此类。

还可以使一个工具提示（帮助文本）与一个对象关联；如果操作员将鼠标指针光标放在对象上，工具提示会显示于一个弹出窗口上。

注意：可以定义页面（包括按钮）上的任意对象的禁用条件。当条件被激活时，对象为灰色并且操作员无法选择它。

另请参见
[触击命令](#)

键盘命令

键盘命令包括一个操作员在键盘上键入的键序列，以及键入键序列时执行的一个指令（或者一系列指令）。

可以定义在下列情况下使用的键盘命令：

- 任何显示于计算机屏幕上的图形画面（系统键盘命令）。
- 仅当特定的图形画面显示时（画面键盘命令）。
- 仅当操作员将鼠标光标指针放到图形画面上的一个对象上时。可以将工具提示与任意对象相关联；当操作员将鼠标指针放在该对象上时，工具提示会显示于一个弹出框上。

对象键盘命令优先于画面键盘命令（优先于系统全局键盘命令）。如果为某个对象定义了一个与画面键盘命令完全相同的键盘命令，那么输入键序列时会执行对象键盘命令，而画面键盘命令则会被忽略。

另请参见
[键盘命令](#)
[系统键盘命令](#)
[键盘键码](#)
[键盘](#)
[为命令定义键序列](#)

滑钮控制

滑钮控制允许操作员通过拖动图形页面上的对象来更改模拟变量的值。滑动钮也会随着变量标签的值的改变自动地移动。

另请参见
[滑钮](#)

系统键盘命令

可以定义系统键盘命令。

要配置系统键盘命令：

1. 如果计划使用任何特殊键，您必须先定义自己的键。
2. 在工程编辑器中，选择**系统 | 键盘命令**。
3. 填写显示的**系统键盘命令**表中的属性。需要输入一个键序列和一个命令。
4. 点击**添加**加入您创建的记录，或点击**替换**修改记录。

另请参见
[系统键盘命令属性](#)
[定义命令和控件](#)

系统键盘命令属性

系统键盘命令具有以下属性：

键序列（32 个字符）

命令的键序列。

命令（253 个字符）

操作员输入键序列时执行的命令（指令集）。

权限（16 个字符）

操作员发布命令所需的权限。

注释（48 个字符）

任何有用的注释。

扩展形式的字段

以下字段通过扩展形式实现（按 **F2**）。

区域（16 个字符）

命令所属的区域。只有属于这个区域的操作员才可以发布命令。例如，如果在这里输入了区域 1，那么操作员必须具有访问区域 1 的权限（以及所有必需的权限）才可以发布命令。

信息记录（32 个字符）

操作员在运行时执行所选的操作时，向记录设备的 *MsgLog* 域发送的文本消息。信息必须是简单文本：

如果包括字段数据使其作为记录信息的一部分，必须在配置设备时插入字段名使其成为设备格式的一部分。例如，在设备视窗的格式域中，可以输入 {MsgLog,20} {FullName,15}。这将会容纳如 **P2 started by John Smith** 这样的消息记录。

消息发向的日志设备在下面的“日志设备”域中指定。

日志设备（16 个字符）

发布命令时信息记录发向的设备。

注意： 必须把 *MsgLog* 字段包含在要发送的信息的记录设备的格式中。

键盘键码

可以定义键盘键码以便更方便发出命令。

定义键盘键码：

1. 选择 **系统 | 键盘键码**。随即会显示“键盘键码”对话框。
2. 输入 **键名称**和**键盘代码**（适用时输入**注释**）。
3. 点击**添加**附加新记录，或点击**替换**修改现有记录。

另请参见

[键盘键码属性](#)
[定义命令和控件](#)

键盘键码属性

键盘键码具有下列属性：

键名称

为键指定的名称。输入一个不超过 16 个字符的值。可以为键盘上的任何一个键定义一个有意义的名称，并在任意的键盘命令中使用这些键。例如，可以将 F1 键定义为登录键。随后在系统中要用到登录键时，使用 F1 键即可。

可以为键盘上的任意键（包括字母数字键 A-Z、a-z 和 0-9）指定键名称。不过，将键定义为命令键之后，就只能作为命令键使用。定义了一个数字字符键（例如字符 A）后，此键就不能作为数据键使用。每次按该键时，都会按该键的定义来识别它，而不会将其识别为本身所代表的字符。键盘键定义通常只使用非字母数字字符键（例如功能键）。

键代码

为键名称指定的代码。输入一个不超过 16 个字符的值。键代码将键名称与实际的键链接。可以将键代码指定为十六进制值，也可以使用已经与该键关联的标准标签。

注释

任何有用的注释。输入一个不超过 48 个字符的值。

扩展形式的字段

以下字段通过扩展形式实现（按 **F2**）。

回显

确定该键使用时是否回显在屏幕上。输入一个不超过 8 个字符的值。

回显	TRUE
----	------

（按下该键时显示（回显）**键名称**。键名称显示在图形画面键盘输入行-动点 1 的图形画面中）

回显	FALSE
----	-------

（按下该键时不显示（回显）**键名称**）

此属性为可选属性。如果不指定回显，其缺省为 True。

键盘类型（16 个字符）

键盘的类型。仅当在同一系统上使用超过一种类型的键盘时，才需要此字段。

如果确实有超过一种类型的键盘，请对第一种键盘使用键盘类型 1，并对每一种类型分别使用一个号码（1, 2, 3, 等），然后对每个键盘进行定义。对于所有通用键可以使用缺省类型（类型 0）。

键盘

可以使用非标准键盘（以及多个键盘）控制 Vijeo Citect。还可以定义键名称。

使用非标准键盘

可以在运行时系统中使用多种类型的键盘。最常用的键盘是 IBM 兼容键盘；这种类型的键盘为缺省键盘。许多工业键盘不符合此标准；如果使用非标准键盘，必须在数据库中对这类键盘中的每一种进行定义。

使用多个键盘

有时，可能需要使用不同类型的多个键盘；例如，在控制室使用 IBM 兼容键盘，而在工厂车间使用密封膜键盘。如果使用不止一种类型的键盘，就必须为每种键盘定义所有键并将每种键盘类型指定给相应的计算机。

必须通过 [Keyboard]Type 参数为每台计算机设置键盘类型。

注意：如果定义使用鼠标键的命令，应测试不要将双击命令误设为单击命令。双击是单击的扩展；单击信息将在双击信息之前被接受。

定义键名称

可以通过一个有意义的名称而不是通过键本身关联键盘键。例如，可以将 F1 键关联为“帮助”键，而将 F2 键关联为“登录”键。在命令中使用该键时，可以使用您定义的名称。

可以为键盘上的任意键（包括字母数字键 A-Z、a-z 和 0-9）指定键名称。不过，将键定义为命令键之后，就只能作为命令键使用。定义了一个数字字符键（例如字符 A）后，此键就不能作为数据键使用。每次按该键时，都会按该键的定义来识别它，而不会将其识别为本身所代表的字符。键盘键定义通常只使用非字母数字字符键（例如功能键）。

另请参见

[为命令定义键序列](#)
[定义命令和控件](#)

为命令定义键序列

要定义命令，必须指定操作员发布该命令时键入的键序列。可以为键序列指定一个单键，例如功能键 F2：

键序列	F2
-----	----

也可以指定数个必须按顺序键入的键，例如功能键 F2 加 Enter 键：

键序列	F2 Enter
-----	----------

注意：如果在序列中使用多个键，则必须用空格键将每个键隔开。

应避免键盘命令混淆，此情况可能在定义了使用一个通用键的数个单独命令时出现。例如，如果将一个命令的键序列定义为 F3，并且将第二个命令的键序列定义为 F3 F4，那么当 F3 被按下时，第一个命令会立即被执行；第二个命令可能也不会执行。为了避免出现此冲突，向通用键盘命令中加入一个**分隔符**，例如：

键序列	F3 Enter
命令	SP1 = 50;

键序列	F3 F4 Enter
命令	SP1 = 100;

这些命令直到操作员输入分隔符（Enter 键）时才会被执行。

另请参见

[使用热键](#)

[使用变量数据输入](#)

[传递多个参数](#)

[向函数传递键盘参数](#)

使用热键

被定义为热键的命令会在这个键被按下时立即执行。可以在键序列中定义一个单键,您只能用热键定义作用于现有键盘缓冲的命令（例如 Backspace 键和 Delete 键）。定义热键时，向下面的例子那样对键序列添加一个星号 (*) 前缀：

键序列	*Backspace
命令	KeyBs()

在运行期间，这个命令与计算机标准键盘上的 Backspace 键的操作完全相同，都是修改输入的错误。（每次按下 Backspace 键时，命令行中的最后一个键就会被删除。）

注意： 热键命令只能被定义为系统（全局）命令。

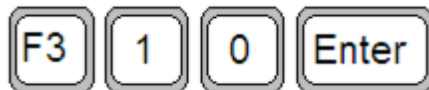
另请参见
[使用变量数据输入](#)

使用变量数据输入

可以配置一个键盘命令用于在运行时间接受变量数据。当系统运行时，操作员可以通过这个命令输入一个值，而这个值将作为一个或多个参数被传递到 Cicode 命令中。您可以定义一个键盘命令使操作员使用不同的值。例如，可以定义下面的命令在运行期间设定变量 SP1:

键序列	F3 ### Enter
命令	SP1 = Arg1;

在此例中，操作员先按下 F3 功能键，输入一个新值，然后按下一个回车键就可以将变量 SP1 设为一个新的值，例如：



将 SP1 的值设为 10。

每一个键序列中的“#”字符代表操作员在命令中输入的一个键盘字符。在上面的例子中，操作员在发布命令时最多可以输入 3 个键盘字符。（# 字符的个数决定了操作员在参数中可以输入的字符的最大个数；如果操作员输入的字符超过 3 个，就会显示一个“无效命令”提醒信息。）

上面例子中命令的发布如下所示：



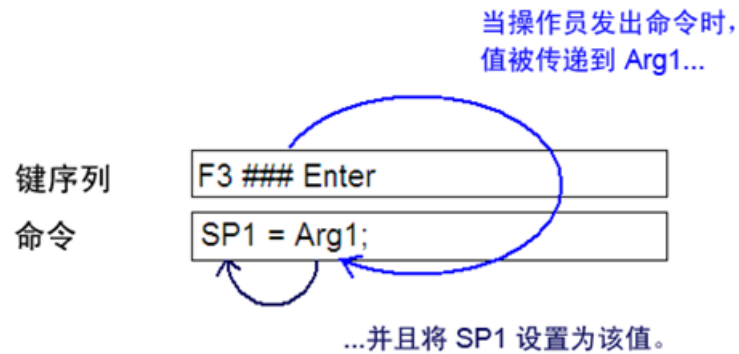
或



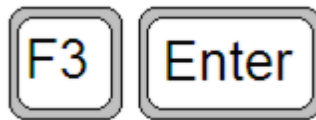
或



这个命令发布时（操作员按下回车键），值被传到命令 `Arg1` 中。



注意： 如果操作员不输入任何数据，（即键序列：



被使用了），`Arg1` 的值为 0，而变量的值被设为 0。为了避免这种情况，请使用 `ArgValue1` 标签，例如：

```
Command SP1 = ArgValue1;
```

`ArgValue1` 标签会检查非法的输入；如果这个输入是无效的，那么变量的值不会改变。也可以使用 `StrToValue()` 函数。

注意 `ArgValue1` 标签和 `StrToValue()` 函数会中断命令。任何其后紧跟 `ArgValue1` 标签或者 `StrToValue()` 函数的指令都不会执行。

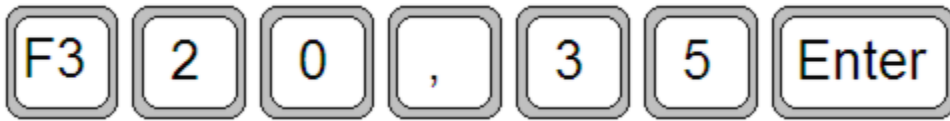
另请参见
[传递多个参数](#)

传递多个参数

可通过用逗号分隔每个参数的方式将多个参数传递给一个 `Cicode` 命令。每个参数按顺序传递给命令，并且在命令字段中引用为 `Arg1`, `Arg2`, `Arg3` 等。例如：

键序列	F3 ###, ### Enter
命令	SP1 = Arg1; SP2 = Arg2;

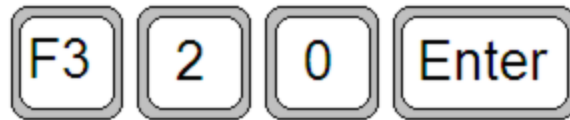
在这种情况下，操作员可以用一个命令设定两个变量，例如：



将变量 SP1 设为 20 而将 SP2 设为 35。

最多可以使用 8 个参数。不过，应避免传递过多的参数。

注意：没有办法检查每个参数的输入是否有效。如果操作员不为这些参数中的某一个输入任何数据（即，使用键序列：



），Arg2 会被设为 0，而且第二个变量也会被设为 0。如果无效的输入可能在工程或在更大系统中产生不希望的行为，请勿在一个命令中使用多个变量。



不希望的设备操作

除非所有可能的输入组合都已测试过且证明是安全的，否则，请勿使用按键序列将多个变量传递给 Cicode 命令。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

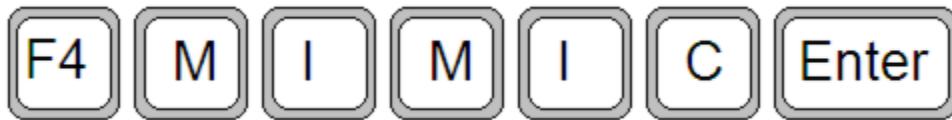
另请参见
[向函数传递键盘参数](#)

向函数传递键盘参数

您也可以在运行期间将参数直接传递给函数，如下例所示：

键序列	F4 ##### Enter
命令	PageDisplay(Arg1);

在本例中，操作员可以通过一个简单的命令选择任意的苯工程中定义的图形画面，例如：



选择“Mimic”画面。

注意：所有键盘参数都是按字符串值传递的。如果命令(或者函数)需要一个数字值，Cicode 会在使用它之前将字符串转化为数字。

如果使用变量数据，那么操作员只能向这个数据输入字母数字字符 (A-Z、a-z 和 0-9)。请不要将变量数据作为键序列的最后一项使用，以避免发生混淆；始终使用分隔符。

Chapter: 23 配置和处理报警

保护工厂设备是 Vijeo Citect 的一个主要作用。报警机制对设备数据进行持续的监测并且在设备出现任何意外事件（比如设备不响应输入命令）提醒操作人员。

支持两种类型的报警：

- **硬件报警**。诊断子程序持续运行，来检查 I/O 设备等所有外围设备。不希望出现的条件和事件都会被自动地向操作员报告。这种机制完全集成在产品内部，无需配置。
- **配置报警**。和硬件报警不同，用户必须对这些报警进行配置来对工厂中的意外条件进行通报（例如，当某个容器中的液面过高时或者当某个电机的温度过高时）。

另请参见

[配置报警](#)

[使用报警延时](#)

[使用自定义报警过滤器](#)

[报警分类](#)

[报警显示格式的设置](#)

[将报警属性作为标签使用](#)

[处理运行时的报警](#)

配置报警

可以使用七类配置的报警：

- [数字量报警](#)
- [多数字量报警](#)
- [带时间标签的报警](#)
- [模拟报警](#)
- [高级报警](#)
- [带时间标签的数字量报警](#)
- [带时间标签的模拟报警](#)

用户可以对每一个报警进行单独处理，或者将每一个报警归到不同的分类中，并在这些分类中赋予它们不同的优先级。然后用户可以对它们进行显示、确认、复位以及将所有的报警按照其特定的分类进行记录。

用户还可以使用 `SummarySort` 和 `SummarySortMode` 参数对在报警汇总页面中显示的报警顺序自行定义。（此顺序将会替代报警分类优先级顺序。）

为了帮助操作人员对报警进行处理，用户可以创建包含有关报警信息（例如操作人员为改正现状所应采取的措施）的图形页面。用户可以在报警出现时自动显示这些页面，或者仅当某个操作人员使用报警帮助键盘命令（Alarm Help Keyboard Command）时才进行显示。

报警属性还可以在任何可以使用普通变量标签的场合进行使用。例如，报警状态可用于改变一个图形画面上符号的颜色。

另请参见

[使用报警延时](#)

[报警分类](#)

使用报警延迟

可以通过“报警延迟”属性配置数字量、模拟量和高级报警，这样它们就不会激活，除非触发条件在指定周期保持 **true**。

对于模拟量报警，这表示报警激活之前，模拟量变量必须在报警延迟周期持续时间的指定范围内。对于数字量报警和高级报警，数字量变量或 **Cicode** 表达式的触发条件必须在延迟周期内保持 **true**。

例如，报警延迟可能在监视温度时非常有用。通过设置延迟周期，可以过滤掉由进入或超出不同范围的温度引起的瞬时报警。

另请参见

[使用自定义报警过滤器](#)

[报警分类](#)

使用自定义报警过滤器

可以为报警标签定义关键字，用于随后对报警数据执行自定义查询。

在“报警属性”对话框中，可以为系统中的每个报警最多定义八个自定义过滤器，这样可生成过滤特定信息报警数据的查询。

例如，您要对监视设备和状况（可能引起火灾）的所有报警赋予不同优先级。可以将每个相关标签的“自定义过滤器 1”设置为“火灾”。然后可调用 **Cicode** 函数，请求所有带有“自定义过滤器 1”字段（相当于“火灾”）的报警。最终结果会显示报警列表，通知任何潜在易燃环境下的操作员。

还可以留出“自定义过滤器 2”定义报警关联的设备类型和每个报警相应的标签（例如“泵”、“传送带”等）。然后可创建查询，列出与特定机器类型相关的所有报警，例如，与泵关联的所有报警。

实施使用自定义报警过滤器的查询

根据自定义报警过滤器实施查询需要执行两个步骤：

1. 创建执行您要实施的查询的 **Cicode** 函数。需要创建的查询函数的格式为：

```
INT FUNCTION Query (INT nRecordID, INT nVersion, STRING sInfo)
```

其中：

nRecordID	调用 AlarmGetFieldRec 时要使用的值； 例如，AlarmGetFieldRec(nRID, "CUSTOM1", nVer)。 (有关可用字段的详细信息，请参见 AlarmGetFieldRec 的帮助。)
nVersion	记录的版本；此参数应在每次更改记录时增加。
sInfo	用户定义的字符串，用来控制查询函数中的逻辑。

预期的返回参数值为 0 或 1，为 1 时表示将显示该记录。

2. 此查询通过函数 AlarmSetQuery 调用它来进行设置。继续操作之前，确认已经正确地为您的报警标签定义了自定义过滤器。

范例

您需要为您工厂的传送带定义调用当前报警的查询。这可以通过将关键字“传送带”应用到字段 **自定义过滤器 1**，从已经标识为与传送带关联的报警标签来绘制。

您要创建的查询函数如下：

```

INT
FUNCTION CheckCustom1(INT nRid, INT nVer, STRING sValue)
  STRING sCustom;
  // Get the information in CUSTOM1
  sCustom = AlarmGetFieldRec(nRid, "CUSTOM1", nVer);
  IF sCustom = sValue THEN
    // Lets display this
    RETURN 1;
  ELSE
    // Skip over this
    RETURN 0;
  END
END

```

假定您随后要在列出所有当前传送带报警的图形画面上创建一个按钮。可通过将 AlarmSetQuery 函数应用到该按钮来实现。

```
AlarmSetQuery(0, "CheckCustom1", "conveyor");
```

还可以创建一个重置按钮，通过取消查询来清除当前显示列表：

```
AlarmSetQuery(-1, "", "")
```

可以选择使用参数 "pumpcontrol" 来调用特定的 Cicode 函数，例如 "Customfeld1"；也可以使用更一般的方法，用参数 "CUSTOM1= pumpcontrol" 来调用函数 "Checkfield"。在此情况下，Cicode 函数会解析传递的字符串并检查指定的字段。

效率注意事项

Cicode 比预定义的过滤器需要更长的实现时间。为了避免对已经检测到的报警进行不必要的处理，需要缺省启用智能自定义过滤器，以尽可能地减少处理大型报警记录的数量。

使用智能自定义过滤器，在第一次运行某查询时，您的函数将对每个报警记录进行检查。随后的屏幕显示或记录变化都将仅调用更改记录或新记录的函数。

对于大多数查询而言，您可能希望您的系统行为如此。但是，对于一些特殊的查询而言，您可能希望禁用此行为，并检查每个警报记录。

例如，当您请求前 10 秒内所有已更改的或新的警报时，可能希望如此。在此情况下，“智能”过滤不会返回所需的报警，因为某些记录虽然不会被更改，但仍会包含在您的查询中，或被排除在您的查询之外。

要控制这种情况，可使用以下两种机制：

- 如果您将全局 ini 参数 [ALARM]EnableSmartCustomFilters 设置为 0，将禁止智能过滤器应用于所有查询。
- 以每个查询为基础，将第 4 个可选参数设置为 1 或 TRUE，例如：

```
AlarmsetQuery(0, "MyQueryTime", "10", TRUE);
```

当设置为 TRUE 时，将强迫查询运行。缺省值为 0 或 FALSE（允许智能查询）。

另请参见
[报警分类](#)

报警分类

可以为系统中的每个报警指定分类，并且可以将每个分类作为一个组进行处理。对于每个分类，可以设置报警显示详细信息（字体和页面类型）、日志记录详细信息（打印机或数据文件），以及在触发该分类的报警时采取的操作（例如，激活声音报警）。

每个分类都可以具有关联的优先级。报警优先级可用于对报警显示进行排序，为操作员提供有用的过滤功能。

您还可以使用 SummarySort 和 SummarySortMode 参数对报警在报警显示画面上显示的顺序进行自定义。（此顺序将覆盖报警分类优先级顺序。）

您最多可以配置 16376 个报警分类。如果没有为报警指定分类，则该报警与报警分类 0 具有相同的属性。如果没有定义报警分类 0，则将对该分类使用缺省值。

定义报警分类：

1. 选择**系统 | 报警分类**。此时将出现“报警分类”对话框。
2. 输入报警分类属性。
3. 点击**添加**以附加新记录，或者点击**替换**以修改现有记录。
使用“报警分类”对话框配置报警。

另请参见
[报警分类属性](#)

报警分类属性

[报警分类](#)具有下列属性：

分类编号

报警分类 (0-16375)。输入一个不超过 16 个字符的分类。

保留分类 254 是为了归纳用户创建的报警摘要条目。保留分类 255 是为了归纳硬件报警。

优先级

归到此报警分类 (0-255) 的所有报警所应用的优先级。报警优先级管理报警的显示顺序、确认顺序和启用顺序，等等。输入一个不超过 16 个字符的优先级。

优先级 1 是最高优先级，而优先级 255 是最低优先级。例如，如果显示了优先级为 1-8 的报警，则会首先按时间/日期顺序显示所有优先级为 1 的报警，接着显示优先级为 2 的报警，然后再显示优先级为 3 的报警，依此类推，直到显示了优先级为 8 的报警。

优先级 0（零）是缺省优先级，所有分类均具有零优先级（以及在此字段中输入的值）。优先级 0 用于引出所有优先级。例如，若要更改报警列表的显示参数，以便所有优先级的报警均可显示，请使用 **AlarmSetInfo**，并令 *类型*=7，令 *值*=0。

注意：当使用优先级 0 来显示所有优先级的报警时，会首先显示优先级仅为 0 的报警，再显示优先级为 1 的报警，然后显示优先级为 2 的报警，依此类推。您也可以使用 **SummarySort** 和 **SummarySortMode** 参数自定义报警在报警显示画面上的显示顺序。（此顺序将覆盖报警分类优先级顺序。）

在报警页显示

定义此分类的报警是否显示在报警页上。此字段的缺省值为 TRUE。输入一个不超过 6 个字符的值。

在摘要页显示

定义此分类的报警是否显示在摘要页上。此字段的缺省值为 TRUE。输入一个不超过 6 个字符的值。

未确认：报警结束字体

用于显示不再活动（但尚未得到确认）的报警的字体。此属性是可选的。如果不指定字体，则字体缺省为 10 磅的棕色字体。输入一个不超过 16 个字符的值。

确认：报警结束字体

用于显示不再活动并且已得到确认的报警的字体。此属性是可选的。如果不指定字体，则字体缺省为 10 磅的白色字体。输入一个不超过 16 个字符的值。

未确认：报警开始字体

用于显示尚未得到确认的表达式的字体。此属性是可选的。如果不指定字体，则字体缺省为 10 磅的黄色字体。输入一个不超过 16 个字符的值。

确认：报警开始字体

用于显示已得到确认的当前报警的字体。此属性是可选的。如果不指定字体，则字体缺省为 10 磅的青色字体。输入一个不超过 16 个字符的值。

禁止字体

用于显示禁止报警的字体。此属性是可选的。如果不指定字体，则字体缺省为 10 磅的白色字体。输入一个不超过 16 个字符的值。

启动时的行动

在触发此分类的报警时执行的 Cicode 命令（最多含 253 个字符）。例如：

报警发生时的动作

```
STOP_PROCESS = 1;
```

当触发此分类的报警时，数字量变量 STOP_PROCESS 设置为“开启”。

注意：不要在此字段中输入 Cicode 阻塞函数。报警系统在轮询过程中执行 ON、OFF 或 ACK 操作。阻塞函数会影响报警的轮询，而且有可能导致报警处理变慢或延迟。



警告

不希望的设备操作

不要在此字段中输入 Cicode 阻塞函数。阻塞函数会影响报警的轮询，而且有可能导致报警处理变慢或延迟。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

如果“报警发生时的动作”是自我参照的，并且采用诸如 TAG1 = TAG1 + 1 的形式，则此命令的情况特殊。由于出于性能方面的考虑而不会在处理“报警发生时的动作”前重新读取标签，因此此命令无法正确运行。因此，此特定命令最初会将 TAG1 的值设置为 1，而不会递增它。

若要在“报警发生时的动作”中正确运行此类型的命令，请使用 TaskNew() 运行您自己的 Cicode 函数以执行标签命令：

报警发生时的动作

```
TaskNew("MyFunc","Data",5);
```

关闭时的行动

在重置此分类的报警时执行的 Cicode 命令（最多含 253 个字符）。例如：

报警结束时的动作

```
ENABLE_PROCESS = 1;
```

当重置此分类的报警时，数字量变量 ENABLE_PROCESS 设置为“开启”。

注意：不要在此字段中输入 Cicode 阻塞函数。报警系统在轮询过程中执行 ON、OFF 或 ACK 操作。阻塞函数会影响报警的轮询，而且有可能导致报警处理变慢或延迟。



警告

不希望的设备操作

不要在此字段中输入 Cicode 阻塞函数。阻塞函数会影响报警的轮询，而且有可能导致报警处理变慢或延迟。忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

报警被确认时的动作

在确认此分类的报警时执行的 Cicode 命令（最多含 253 个字符）。

注意：不要在此字段中输入 Cicode 阻塞函数。报警系统在轮询过程中执行 ON、OFF 或 ACK 操作。阻塞函数会影响报警的轮询，而且有可能导致报警处理变慢或延迟。



警告

不希望的设备操作

不要在此字段中输入 Cicode 阻塞函数。阻塞函数会影响报警的轮询，而且有可能导致报警处理变慢或延迟。忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

报警格式

此分类的所有报警的屏幕显示格式（最多含 254 个字符）。报警显示格式指定分类中所有报警的数据如何显示在报警页上（仅限于屏幕上）。报警页上的每个报警都占用单独的一行，例如：

12:32:21RFP3 Raw Feed pump 3 Overload

显示格式属性是可选的。如果不指定报警显示格式，则格式缺省为：

格式	{Time,12} {Tag,10} {Name,20} {Desc,32}
----	--

有关每种字段类型的详细格式设置信息，请参见 [报警显示字段](#)。

您可以通过设置 [Alarm] DefDspFmt 参数，为所有报警更改此缺省报警显示格式。

注意： 如果某个报警值长于为它设定的显示字段的长度，会将它截断或者用 #OVR（“格式宽度溢出”）提醒信息来替代它。当将报警记录到某设备（例如，打印或写至文件或数据库中）时，为日志设备指定的格式会替代显示格式。

摘要格式

此分类的所有报警的摘要显示格式（最多含 254 个字符）。摘要显示格式指定报警摘要如何显示在报警摘要页上（仅限于屏幕上）。

确切地讲，显示格式是指报警显示格式。但是，您也可以使用其他数据字段。

此属性是可选的。如果不指定摘要显示格式，则格式缺省为：

格式	{Name,20} {OnTime,8} {OffTime,8} {DeltaTime,8} {Comment,22}
----	---

有关每种字段类型的详细格式设置信息，请参见 [报警显示字段](#)。

您可以通过设置 [Alarm] DefSumFmt 参数，为所有报警更改缺省摘要显示格式。

注意： 当将报警记录到某设备（例如，打印或写至文件或数据库中）时，为日志设备指定的格式会替代显示格式。

摘要设备

发送报警摘要的设备（最多含 16 个字符）。

如果已发送报警，并且报警的显示时间长于在 [Alarm] SummaryTimeout 参数中指定的时间，则会将报警记录到摘要设备中。当将报警记录到设备时，会将它从报警摘要页中删除。

将报警打印或写至文件或设备时，在设备中指定的格式会替代显示格式。

此属性是可选的。如果不指定摘要设备，则不会记录报警摘要。

日志设备

发送报警状态变化的设备（最多含 16 个字符）。

每当报警的状态（例如，开启、关闭、确认、启用和禁止）改变时，都会在日志设备中创建一个报警条目。

将报警打印或写至文件或设备时，在设备中指定的格式会替代显示格式。

此属性是可选的。如果不指定日志设备，则不会记录报警状态变化。

记录报警过渡次数：开启

记录报警变为活动状态时的报警详细信息（最多含 6 个字符）。此字段的缺省值为 TRUE。

记录报警过渡次数：关闭

记录报警变为非活动状态时的报警详细信息（最多含 6 个字符）。此字段的缺省值为 TRUE。

记录报警过渡次数：确认

记录报警得到确认时的报警详细信息（最多含 6 个字符）。此字段的缺省值为 TRUE。

注释

任何有用的注释（最多含 48 个字符）。

数字量报警

可以根据一个或两个数字量变量的状态将数字量报警配置为激活状态。当触发条件跨越指定的延迟周期时，报警将变成激活状态。

为数字量报警配置的变量以 Citect.ini 参数 [Alarm]ScanTime 设置的速率进行轮询。如果更改报警状态，则下次轮询变量时将出现通知。请注意，与报警状态关联的时间将表示变量轮询的时间，而不是报警条件出现的实际时间。

如何配置一个数字量报警：

1. 选择**报警 | 数字量报警**。将显示“数字量报警”对话框。
2. 填写属性。请参见 [数字量报警属性](#)。
3. 单击**添加**按钮附加一条新记录，如果已经修改了一条记录，请单击**替换**。

另请参见
[数字量报警属性](#)

数字量报警属性

[数字量报警](#)具有以下属性：

报警标签

报警的名称（最多为 79 个字符）。报警名称对于集群而言必须是唯一的。报警标签名称必须遵循 [标签名称语法](#)。如果该名称不唯一或语法不正确，则可能无法识别。如果具有许多标签，请使用命名约定（参见 [使用结构化标签名称](#)）。这样一来，可以更方便地查找和调试标签。

注意： 尽管集群名称可以保留为空，但该名称对于所有已定义的集群必须是唯一的。

集群名称

运行此报警的集群的名称（最多 16 个字符）。如果未设置集群名称，则此报警将在所有已定义的集群上运行。

报警名称

与该报警相关联的物理设备的名称（最多为 79 个字符）。这是一个可选属性，只有当报警的详细信息显示在屏幕上或记录到设备中时，才使用此属性。

报警描述

报警的描述（最多为 127 个字符）。可以包括变量数据。这是一个可选属性，只有当报警的详细信息显示在屏幕上或记录到设备中时，才使用此属性。

变量标签 A/变量标签 B

触发报警的数字量变量（标签）（最多为 254 个字符）。可以根据一个或两个数字量变量的状态将数字量报警配置为激活状态。

如果只使用一个变量来触发报警，则请使用**变量标签 A** 字段。例如：

变量标签 A	RFP3_TOL
--------	----------

在变量 RFP3_TOL 的状态更改为 ON (1) 时将触发报警。

或者，也可以将报警定义为当变量的状态更改成 OFF (0) 时触发，方法是在数字量地址之前使用逻辑操作符 NOT，例如：

变量标签 A	NOT RFP3_TOL
--------	--------------

在这种情况下，报警在变量 MCOL304 的状态更改为 OFF (0) 时被触发。

可以根据两个数字量变量的状态将数字量报警配置为激活状态，例如：

变量标签 A	RFP3_TOL
变量标签 B	NOT MCOL304

在这种情况下，报警在两个变量的状态都更改为激活状态时被触发：即变量 RFP3_TOL 的状态更改为 ON (1)，且变量 MCOL304 的状态更改为 OFF (0)。

注意：如果将**变量标签 B** 的属性保留为空，则只有变量标签 A 触发报警。

分类

报警分类号或标签（最多为 16 个字符）。这个属性是可选的。如果没有指定分类，则该报警的缺省值为分类 0。

延迟（小时：分钟：秒）

报警延迟周期（最多 16 个字符）。

当触发条件的状态在延迟周期阶段保持为真时，数字量报警变成激活。状态变为真时，激活报警具有 ON 时间。

这个属性是可选的。如果不指定延迟周期，则报警将在被数字量标签触发时立即激活。

注意：延迟期间必须采用 HH:MM:SS（时:分:秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

帮助

显示 AlarmHelp() 函数何时被用户自定义命令所调用的图形页面的名称（最多为 64 个字符）。这个属性是可选的。如果未指定帮助页面，则调用 AlarmHelp() 函数时不会出现任何动作。

注释

任何有用的注释（最多为 48 个字符）。

扩展形式的字段

以下字段通过扩展形式实现（按 F2）。

权限

操作员确认或禁用报警所需的权限（最多为 16 个字符）。

注意：如果您为报警分配了确认权限，请不要对确认报警的命令分配某个权限。如果对命令分配了另一个的权限，那么操作员必须同时拥有两个权限才能对该命令进行确认。更重要的是，此区域定义的内容可能会被忽略。

区域

报警所属的区域（最多为 16 个字符）。如果操作员无权访问某个区域，则报警在报警显示中不可见。例如，如果进入**区域 1**，则操作员必须具有对区域 1 的权限（以及任何所需的其他权限）才能确认或禁用此报警。

注意：此处定义的区域和权限字段必须配合使用。例如，在按钮上定义的权限将忽略报警定义的区域。

自定义过滤器 1...自定义过滤器 8

用于过滤当前报警的用户定义的字符串（最多为 64 个字符）。

在自定义 Cicode 查询中用作搜索条件。通过自定义报警过滤器，操作员可以识别并显示激活报警的子集。

注意：

- 只有当“数字报警”表单在“扩展”模式下打开时，自定义过滤器才可见。
- 这些字段不区分大小写，而且可以包含“A”...“Z”、“a”...“z”、“0”...“9”，以及下划线“_”。
- 自定义过滤器不能以数字开头。

呼叫

指示报警是否应被呼叫的读/写属性。如果其值为 1 (TRUE)，则呼叫该报警。缺省参数值为 0 (FALSE)。请参见 [报警呼叫属性](#)。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paging”来读取或修改。

呼叫组

指示报警所属呼叫组的只读文本字符串。最大长度为 80 个字符。有关如何使用呼叫组字符串的信息，请参见第三方呼叫系统的说明文档。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paginggroup”来读取。例如，将 PagingGroup 的值指定为变量：

```
myString = myCluster.Alarm_1.paginggroup
```

另请参见
[使用自定义报警过滤器](#)

多数字量报警

多数字量报警使用从三个数字量变量（例如：标签 A、B 和 C）的输出定义八个状态。这些状态表示变量可以具有的 true/false 参数值所有可能的组合。

每个状态中的标签参数值按标签 C、标签 B、标签 A 的顺序表示。true 参数值由标签字母表示，0（零）表示 false。

八个状态如下所示：

- **状态 000** - 3 个标签均为 false。
- **状态 00A** - 标签 C 和 B 为 false，标签 A 为 true。
- **状态 0B0** - 标签 C 和 A 为 false，标签 B 为 true。
- **状态 0BA** - 标签 C 为 false，标签 B 和 A 为 true。
- **State C00** - 标签 C 为 true，标签 B 和 A 为 false。
- **状态 C0A** - 标签 C 和 A 为 true，标签 B 为 false。
- **状态 CB0** - 标签 C 和 B 为 true，标签 A 为 false。
- **状态 CBA** - 3 个标签均为 true。

配置多数字量报警属性时，可以设置应触发报警的状态，并指定要在报警变为激活和非激活状态时调用的 Cicode 函数。

为多数字量报警配置的变量以 Citect.ini 参数 [Alarm]ScanTime 设置的速率进行轮询。如果报警状态发生变化，则下次轮询变量时将出现通知。与报警状态关联的时间将表示变量轮询的时间，而不是报警条件出现的实际时间。

范例

在下例中，标签 C 保留为空，为标签 A 和 B 指定了变量 BIT_12 和 BIT_1。如果指定状态 0BA 激活报警，则将标签 A 和 B 更改为 ON (1) 时，报警将激活。

变量标签 A

BIT_12

变量标签 B

BIT_1

变量标签 C

在此例中，为三个标签都指定了变量。如果指定状态 CBA 激活报警，则三个变量都更改为 ON (1) 时，报警将激活。

变量标签 A	RFP3_TOL
--------	----------

变量标签 B	BIT_1
--------	-------

变量标签 C	MCOL_304
--------	----------

如何配置多数字量报警：

1. 选择**报警 | 多数字量报警**。此时将出现“多数字量报警”对话框。
2. 输入报警属性。
3. 单击**添加**附加新记录，或单击**替换**修改现有记录。

另请参见

[多数字量报警属性](#)

多数字量报警属性

[多数字量报警](#)具有下列属性。

报警标签

报警的名称（最多 79 个字符）。报警名称对于集群必须是唯一的。报警标签名称必须遵循[标签名称语法](#)。如果该名称不唯一或语法不正确，则可能无法识别。如果具有许多标签，请使用命名约定（参见[使用结构化标签名称](#)）。这样就可以更方便地查找和调试标签。

注意： 尽管集群名称可以保留为空，但该名称对于所有已定义的集群必须是唯一的。

集群名称

运行此报警的集群的名称（最多 16 个字符）。如果未设置集群名称，则此报警将在所有已定义的集群上运行。

报警名称

与报警关联的物理设备的名称（最多 79 个字符）。此属性是可选属性，只有当报警的详细信息显示在屏幕中或记录到设备中时，才使用此属性。

报警描述

报警的描述（最多为 127 个字符）。可以包括变量数据。此属性是可选属性，只有当报警的详细信息显示在屏幕中或记录到设备中时，才使用此属性。

变量标签 A/变量标签 B/变量标签 C

用于定义八个状态的数字量变量（最多 254 个字符）。每个状态表示标签参数值的不同组合。

在以下范例中，数字量变量 RFP3_TOL、BIT_1 和 MCOL304 是为标签 A、B 和 C 指定的。

变量标签 A	RFP3_TOL
变量标签 B	BIT_1
变量标签 C	MCOL_304

状态和描述

以下八个状态代表所有可能的标签参数值组合（描述最多 8 个字符，状态 1 个字符）。这些标签按标签 C、标签 B、标签 A 的顺序表示。

- **状态 000** - 3 个标签均为 false。
- **状态 00A** - 标签 C 和 B 为 false，标签 A 为 true。
- **状态 0B0** - 标签 C 和 A 为 false，标签 B 为 true。
- **状态 0BA** - 标签 C 为 false，标签 B 和 A 为 true。
- **状态 C00** - 标签 C 为 true，标签 B 和 A 为 false。
- **状态 C0A** - 标签 C 和 A 为 true，标签 B 为 false。
- **状态 CB0** - 标签 C 和 B 为 true，标签 A 为 false。
- **状态 CBA** - 3 个标签均为 true。

对于每个状态，“多数字量报警”对话框中都包含两个字段。在第一个字段中，可以输入描述（例如，“健康”或“已停止”），最多八个字符。

在第二个字段中，可以指明状态是否应触发报警。参数值为 1 表示报警状态，参数值为 0 表示不触发报警。

重新报警（1 个字符）

表示从一个报警状态过渡到另一个报警状态时发生的情况。如果此字段中的参数值为 1，则会在状态发生变化时记录新时间和状态描述 {State_Desc,n}。如果参数值为 0，则只有第一个报警状态的时间和状态描述记录在报警汇总中。

报警开启函数（254 个字符）

多数字量报警变为激活状态时执行的 Cicode 函数，例如：

报警开启函数

```
STOP_PROCESS = 1;
```

触发报警时，数字量变量 STOP_PROCESS 将设置为“开启”。

注意：不要在此字段中输入 Cicode 阻塞函数。报警系统在轮询过程中执行 ON 或 OFF 操作。阻塞函数会影响报警的轮询，而且有可能导致报警处理变慢或延迟。



警告

不希望的设备操作

不要在此字段中输入 Cicode 阻塞函数。阻塞函数会影响报警的轮询，而且有可能导致报警处理变慢或延迟。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

自行引用报警开启函数（采用诸如 TAG1 = TAG1 + 1 的形式）时，此命令将出现情况特殊。由于处理报警开启函数之前不会重新读取标签（出于性能方面的原因），因此此命令无法正确运行。因此，这个特殊的命令最初会将 TAG1 的参数值设置为 1，而不是将其递增。

要在报警开启函数中正确运行此类型的命令，请使用 TaskNew() 运行自己的 Cicode 函数来执行标签命令：

报警开启函数

```
TaskNew("MyFunc","Data",5);
```

报警关闭函数（254 个字符）

多数字量报警变为非激活状态时执行的 Cicode 函数（最多 254 个字符）。例如，

报警关闭函数

```
ENABLE_PROCESS = 1;
```

重置此分类中的报警时，数字量变量 ENABLE_PROCESS 将设置为“开启”。

注意：不要在此字段中输入 Cicode 阻塞函数。报警系统在轮询过程中执行 ON 或 OFF 操作。阻塞函数会影响报警的轮询，而且有可能导致报警处理变慢或延迟。



警告

不希望的设备操作

不要在此字段中输入 Cicode 阻塞函数。阻塞函数会影响报警的轮询，而且有可能导致报警处理变慢或延迟。
忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

分类

报警分类号或标签（最多 16 个字符）。此属性是可选属性。如果未指定分类，则报警将缺省为分类 0。

帮助

显示 AlarmHelp() 函数何时被用户自定义命令所调用的图形页面的名称（最多为 64 个字符）。此属性是可选属性。如果未指定帮助页面，则调用 AlarmHelp() 函数时不会出现任何动作。

权限

操作员确认或禁止报警所需的权限（最多 16 个字符）。

注意：如果为报警分配了确认权限，则不要为确认报警的命令分配权限。如果为命令分配了不同的权限，则操作员必须同时拥有两个权限才能确认该命令。

区域

报警所属的区域（最多 16 个字符）。如果操作员无权访问某个区域，则报警在报警显示中不可见。例如，如果进入此处的区域 1，则操作人员必须具有对区域 1 的访问权限（以及任何所需的权限）才能确认或禁止此报警。

注释

任何有用的注释（最多 48 个字符）。

抑制

报警所属的抑制组的编号。此编号是 0-65535 之间的整型参数值。同一个组中的报警在此字段中显示同一个参数值。

此属性与抑制级别结合使用。

注意：要为抑制组分配名称，请使用整型参数值将名称定义为标签。

级别（抑制级别）

抑制组内报警的级别（整型参数值）。此参数值介于 0 与 255 之间，级别越低，优先级越高。

通过此属性，激活当前报警可以抑制同一个抑制组中优先级较低的报警。如果这种情况时，则只显示较高的优先级（较低的级别）。优先级较低（级别较高）的报警只能在优先级较高（级别较低）的报警变为非激活状态时激活和显示。

如果同时触发了同一个抑制组中不同优先级的两个报警，则报警列表中将显示两个报警。这是因为激活两个报警时，优先级较高的报警不是原来就处于激活状态，因此不能抑制优先级较低的报警。

如果希望优先级较高的报警始终在优先级较低的报警之前激活，而不管其他因素，则将优先级较高的报警存储在更接近报警数据库开始的位置。可以为触发的报警从头到尾扫描数据库，如果优先级较高的报警在数据库中离开始位置更近，则会首先激活并在抑制组内抑制任何优先级较低的报警。

扩展形式的字段

以下字段通过扩展形式实现（按 **F2**）。

自定义过滤器 1...自定义过滤器 8

用于过滤激活报警的用户定义的字符串（最多 64 个字符）。

在自定义 Cicode 查询中用作搜索条件。通过自定义报警过滤器，操作员可以识别并显示激活报警的子集。

注意：

- 只有“数字报警”表单在扩展模式下打开时，自定义过滤器才可见。
- 这些字段不区分大小写，而且可以包含“A”..“Z”、“a”..“z”以及下划线“_”。
- 自定义过滤器不能以数字开头。

呼叫

指示报警是否应被呼叫的读/写属性。如果其值为 1 (TRUE)，则呼叫该报警。缺省参数值为 0 (FALSE)。请参见 [报警呼叫属性](#)。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paging”来读取或修改。

呼叫组

指示报警所属呼叫组的只读文本字符串。最大长度为 80 个字符。有关如何使用呼叫组字符串的信息，请参见第三方呼叫系统的说明文档。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paginggroup”来读取。例如，将 PagingGroup 的值指定为变量：

```
myString = myCluster.Alarm_1.paginggroup
```

另请参见

[使用自定义报警过滤器](#)

带时间标签的报警

带时间标签的报警与数字量报警相似，不同的是，计数器在发生触发条件时可用于提供正确的时间戳，而不仅仅是变量轮询的时间。带时间标签的报警只能与单个数字量变量相关联。

报警变量的轮询频率由 [Alarm]ScanTime 设置，但是，计时器的值可用于定义与状态变更相关联的时间。

用户可以使用下面三种计数器或者计时器之一来对带时间标签报警的触发进行记录：

- **连续计数器**。Citect 通过对某个单元中的连续计数器进行读取来确定报警被触发的顺序。按照报警被触发时计数器的值（确切的时间并没有被记录）来对报警进行分类。
- **毫秒计数器**。如果单元支持毫秒计数器，用户可以对（单元中的）计数器编程来进行 24 小时的计数（按毫秒），接着再（在午夜）进行复位。通过对（单元中）该计时器变量中的值进行读取来确定报警被触发的确切时间。
- **LONGBCD 计时器**。通过采用 LONGBCD 计时器，用户可以记录某个带时间标签的报警被激活的确切时间。对该变量以及报警激活时的报警标签进行读取。

对某个带时间标签的报警进行配置：

1. 选择**报警 | 带时间标签的报警**。此时将出现“带时间标签的报警”对话框。
2. 输入报警属性。
3. 点击**添加**附加新记录，或点击**替换**修改现有记录。

另请参见

[带时间标签的报警属性](#)

带时间标签的报警属性

[带时间标签的报警](#)具有以下各项属性：

报警标签

报警的名称（最多 79 个字符）。报警名称对于集群必须是唯一的。报警标签名称必须遵循[标签名称语法](#)。如果该名称不唯一或语法不正确，则可能无法识别。如果具有许多标签，请使用命名约定（参见[使用结构化标签名称](#)）。这样就可以更方便地查找和调试标签。

注意： 尽管集群名称可以保留为空，但该名称对于所有已定义的集群必须是唯一的。

集群名称

运行此报警的集群的名称（最多 16 个字符）。如果未设置集群名称，则此报警将在所有已定义的集群上运行。

报警名称

与该报警相关联的物理设备的名称（最多为 79 个字符）。此属性是可选属性，只有当报警的详细信息显示在屏幕中或记录到设备中时，才使用此属性。

报警描述

报警的描述（最多为 127 个字符）。可以包括变量数据。此属性是可选属性，只有当报警的详细信息显示在屏幕中或记录到设备中时，才使用此属性。

变量标签

触发报警的数字量变量（标签）（最多为 254 个字符）。

计时器

表示在 I/O 设备中配置的计数器（或者毫秒计时器）的变量标签或者 Cicode 表达式（最多 254 个字符）。在 I/O 设备中该计数器必须通过程序来进行配置和维持，并且当报警被触发时是只读的。

用户可以使用下面三种计数器或者计时器之一来对带时间标签报警的触发进行记录：

- **连续计数器。** Citect 通过对某个单元中的连续计数器进行读取来确定报警被触发的顺序。按照报警被触发时计数器的值（确切的时间并没有被记录）来对报警进行分类。用户必须对计数器（单元中的）进行编程使得其连续计数直至达到其计数上限，再进行复位，然后再次开始计数直到计数上限。
- **毫秒计数器。** 如果单元支持毫秒计数器，用户可以对（单元中的）计数器编程来进行 24 小时的计数（按毫秒），接着再（在午夜）进行复位。通过对（单元中）该计时器变量中的值进行读取来确定报警被触发的确切时间。
- **LONGBCD 计时器。** 通过采用 LONGBCD 计时器，用户可以记录某个带时间标签的报警被激活的确切时间。对该变量以及报警激活时的报警标签进行读取。用户必须采用如下的格式来对 LONGBCD 变量进行编程：

BYTE	含义	范围
------	----	----

BYTE	含义	范围
1 st	小时	00-24 (最高有效位 (MSB))
2 nd	分钟	00-60
3 rd	秒	00-60
4 th	100 th /sec	00-100 (最高有效位 (MSB))

- 此字段也可用于与 PLC 代码进行握手通讯：PLC 将了解到已读取了计时器寄存器，且 PLC 将覆盖上一个值。例如，可以采用以下保存在 Cicode 文件中的代码：

```

INT
FUNCTION
AlarmTimerReset(INT iTimer, STRING sTimerTrigger)
TagWrite(sTimerTrigger, 0); //Reset the trigger
RETURN iTimer; //Return the timer value to the alarm system
END

```

用户可以对带时间标签的报警进行如下配置：

变量标签	AlmTrigger1
计时器	Timer(AlmTimer1, "AlmTrigger1")

其中 **AlmTimer1** 是保存报警时间的 PLC 寄存器，**AlmTrigger1** 是报警触发位。当 **AlmTrigger1** 被置为一（1）时，触发报警并调用 Cicode 函数。在调用函数时，将读取 **AlmTimer1** 寄存器。函数对该触发位进行复位（进行握手通讯），**AlmTimer1** 的值将被返回到报警系统中。

分类

报警分类号或标签（最多 16 个字符）。此属性为可选属性。如果未指定分类，则报警将缺省为分类 0。

帮助

显示 **AlarmHelp()** 函数何时被用户自定义命令所调用的图形页面的名称（最多为 64 个字符）。此属性为可选属性。如果未指定帮助页面，则调用 **AlarmHelp()** 函数时不会出现任何动作。

注释

任何有用的注释（最多为 48 个字符）。

扩展形式的字段

以下字段通过扩展形式实现（按 **F2**）。

权限

操作员确认或禁止报警所需的权限（最多 16 个字符）。

注意： 如果为报警分配了确认权限，则不要为确认报警的命令分配权限。如果对命令分配了另一个的权限，那么操作员必须同时拥有两个权限才能对该命令进行确认。

区域

报警所属的区域（最多为 16 个字符）。如果操作员无权访问某个区域，则报警在报警显示中不可见。例如，如果进入此处的区域 1，则操作人员必须具有对区域 1 的访问权限（以及任何所需的权限）才能确认或禁止此报警。

自定义过滤器 1...自定义过滤器 8

用于过滤激活报警的用户定义的字符串（最多 64 个字符）。

在自定义 Cicode 查询中用作搜索条件。通过自定义报警过滤器，操作员可以识别并显示激活报警的子集。

注意：

- 只有当“数字报警”表单在“扩展”模式下打开时，自定义过滤器才可见。
- 这些字段不区分大小写，而且可以包含“A”...“Z”、“a”...“z”、“0”...“9”，以及下划线“_”。
- 自定义过滤器不能以数字开头。

呼叫

指示报警是否应被呼叫的读/写属性。如果其值为 1 (TRUE)，则呼叫该报警。缺省参数值为 0 (FALSE)。请参见 [报警呼叫属性](#)。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paging”来读取或修改。

呼叫组

指示报警所属呼叫组的只读文本字符串。最大长度为 80 个字符。有关如何使用呼叫组字符串的信息，请参见第三方呼叫系统的说明文档。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paginggroup”来读取。例如，将 PagingGroup 的值指定为变量：

```
myString = myCluster.Alarm_1.paginggroup
```

另请参见

[使用自定义报警过滤器](#)

模拟量报警

当模拟变量更改超出一个或多个特定限制时，将触发模拟量报警。可以将每个报警配置为以下类型的任意组合。

- **高限和高高限报警** - 值达到非典型高限值
- **低限和低低限报警** - 值达到非典型低限值
- **偏差报警** - 值偏离预定义的设置点
- **变化速率报警** - 在指定的时间段内发生显著的值变化。

为模拟量报警配置的变量以 Citect.ini 参数 [Alarm]ScanTime 设置的速率进行轮询。如果触发报警状态，则下次轮询变量时将出现通知。请注意，与报警状态关联的时间将表示变量轮询的时间，而不是报警条件出现的实际时间。

配置模拟量报警：

1. 选择**报警 | 模拟量报警**。此时将出现“模拟量报警”对话框。
2. 输入报警属性。
3. 点击**添加**以附加新记录，或者点击**替换**以修改现有记录。

另请参见

[模拟量报警属性](#)

模拟量报警属性

[模拟量报警](#)具有以下属性：

报警标签

报警的名称（最多为 79 个字符）。报警名称对于集群而言必须是唯一的。报警标签名称必须遵循 [标签名称语法](#)。如果该名称不唯一或语法不正确，则可能无法识别。如果您具有许多标签，请使用命名约定（请参见 [使用结构化标签名称](#)）。这样一来，可以更方便地查找和调试标签。

注意： 尽管集群名称可以保留为空，但该名称对于所有已定义的集群必须是唯一的。

集群名称

运行此报警的集群的名称（最多 16 个字符）。如果未设置集群名称，则此报警将在所有已定义的集群上运行。

报警名称

与该报警相关联的物理设备的名称（最多为 79 个字符）。这是一个可选属性。只有当报警的详细信息显示在屏幕上或记录到设备中时，才使用此属性。

变量标签

触发报警的模拟量变量（标签）（最多为 254 个字符）。

设置点

确定是否触发偏差报警的模拟量变量标签或基准值（最多为 254 个字符）。这个属性是可选的。如果没有指定设置点，则缺省值为 0（零）。

高高限

用作高高限报警的触发条件的值（最多为 11 个字符）。如果在高高限延迟期间内变量标签的值超过此值，则高高限报警将成为当前报警。当标签超过高高限值时，当前报警处于打开状态。

由于高报警必须先于高高限报警，所以当高高限报警被触发时将会替代高报警。如果需要在报警页面上同时显示某个模拟量报警的不同状态，则应当对每一种状态单独配置一个报警。（每一个报警将对同一个标签进行监视。）

高高限延迟

高高限报警的延迟期间（最多 16 个字符）。只有在此时间段内满足报警的触发条件时，该报警才会激活。

此属性是可选的。如果没有设置值，则在该标签超出高高限值时将激活高高限报警。

注意：延迟期间必须采用 HH:MM:SS（时：分：秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

高限

用作高限报警的触发条件的值（最多为 11 个字符）。如果在高限延迟期间内变量标签的值超过此值，则高限报警将为当前报警。当标签超过高限值时，当前报警处于打开状态。

高限延迟

高限报警的延迟期间（最多 16 个字符）。只有在此时间段内满足报警的触发条件时，该报警才会激活。

此属性是可选的。如果没有设置值，则在该标签超出高限值时将激活高限报警。

注意：延迟期间必须采用 HH:MM:SS（时：分：秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

如果在高限延迟期间内标签值从高限值增加到高高限值，则延迟计时器将被重置。只有在该值始终处于延迟期间的高高限值范围内时，才会激活高高限报警。

如果在高限延迟期间过期后该值从高限值增加到高高限值，则高限报警将被激活，随后高高限报警的延迟期间即开始。

如果该标签值超出高高限值，随后在高高限延迟期间过期后又降至该值以下，则将立即触发高限报警。当标签值超过高高限值时，当前报警处于打开状态。

这些点还应用于在低限范围与低低限范围之间徘徊的标签值。

低限

用作低限报警的触发条件的值（最多为 11 个字符）。当变量标签的值降至此值之下且在低限延迟期间内始终低于该值时，低限报警将为当前警报。当标签降至低限值以下时，当前报警处于打开状态。

低限延迟

低限报警的延迟期间（最多 16 个字符）。只有在此时间段内满足报警的触发条件时，该报警才会激活。

此属性是可选的。如果没有设置值，则在该标签降至低限值以下时将激活低限报警。

注意：延迟期间必须采用 HH:MM:SS（时：分：秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

低低限

用作低低限报警的触发条件的值（最多为 11 个字符）。当变量标签的值降至此值之下且在低低限延迟期间内始终低于该值时，低低限报警将为当前警报。当标签降至低低限值以下时，当前报警处于打开状态。

由于低限报警必须先于低低限报警，所以当低低限报警被触发时将会替代低限报警。如果需要在报警页面上同时显示某个模拟量报警的不同状态，则应当对每一种状态单独配置一个报警。（每一个报警将对同一个标签进行监视。）

低低限延迟

低低限报警的延迟期间（最多 16 个字符）。只有在此时间段内满足报警的触发条件时，该报警才会激活。

此属性是可选的。如果没有设置值，则在该标签降至低低限值以下时将激活低低限报警。

注意：延迟期间必须采用 HH:MM:SS（时：分：秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

偏差

用作偏差报警的触发条件的值（最多为 10 个字符）。当变量标签的值在偏差延迟期间内始终处于偏差范围（由设置点确定）以外时，就会激活偏差报警。

这个属性是可选的。如果用户没有指定某个偏差量，将不会触发偏差报警。

偏差延迟

偏差报警的延迟期间（最多 16 个字符）。只有在此时间段内满足报警的触发条件时，该报警才会激活。

此属性是可选的。如果没有设置值，则在变量标签处于偏差范围以外时将立即激活偏差报警。

注意：延迟期间必须采用 HH:MM:SS（时：分：秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

速率

通过将此值除以报警期间，可以确定变量标签值变化的“最大速率”（最多为 10 个字符）。每次扫描时，都会检查该标签的值。如果其变化速率大于该“最大速率”，将会触发一个变化速率报警报警。

例如，要确认某个储存罐不会过快地填满，用户可以配置一个变化速率报警，采用的速率参数 Rate 为 300 升，报警周期参数 **[Alarm]Period** 为 60 秒，报警扫描时间参数 **[Alarm]ScanTime** 为 1 秒。这意味着所允许的最大变化速率为 5 升/秒（300 升/60 秒）。在每一个扫描时间间隔内对实际的变化速率进行计算。也就是说，每一秒都检查储存罐的当前液面并同前一秒所记录的液面进行比较。如果实际的变化速率是，比方说，8 升/秒，将会立即触发一个变化速率报警。

变量标签死区 % 必须小于按照变量标签的工程整定量划分的报警速率。否则，只有在变量标签的变化超出死区值时，变化速率报警才会关闭。

这个属性是可选的。如果用户没有指定某个数值，将不会激活变化速率报警。

死区

偏差报警处于不活动状态之前**变量标签**必须返回的值（最多为 10 的字符）。

格式

在图形页面中进行显示、写入到某个文件中或者传递到某个（等待的参数为某个字符串的）函数（变量的）显示格式（最多为 11 的字符）。

这个属性是可选的。如果用户没有指定某个格式，该格式的缺省值为对变量标签所指定的格式。

分类

报警分类号或标签（最多为 16 个字符）。这个属性是可选的。如果没有指定分类，则该报警的缺省值为分类 0。

帮助

显示 AlarmHelp() 函数何时被用户自定义命令所调用的图形页面的名称（最多为 64 个字符）。这个属性是可选的。如果未指定帮助页面，则调用 AlarmHelp() 函数时不会出现任何动作。

注释

任何有用的注释（最多为 48 个字符）。

扩展形式的字段

以下字段通过扩展形式实现（按 F2）。

权限

操作员确认或禁用报警所需的权限（最多为 16 个字符）。

注意：如果您为报警分配了确认权限，请不要对确认报警的命令分配某个权限。如果对命令分配了不同的权限，那么操作人员必须同时拥有两个权限才能对该命令进行确认。

区域

报警所属的区域（最多为 16 个字符）。如果某个操作人员无权访问某个区域，则报警在报警显示页面上不可见。例如，如果您进入此处的区域 1，则操作人员必须具有对区域 1 的权限（以及任何所需的其它权限）才能确认或禁用此报警。

自定义过滤器 1...自定义过滤器 8

用于过滤当前报警的用户定义的字符串（最多为 64 个字符）。

在自定义 Cicode 查询功能中用作搜索标准，借助自定义报警过滤器，操作人员可以识别并显示当前报警的子集。

注意：

- 只有在“数字报警”表单在扩展模式下打开时，自定义过滤器才可见。
- 这些字段不区分大小写，而且可以包含 "A"... "Z"、"a"... "z"、"0"... "9"，以及下划线 "_"。
- 自定义过滤器不能以数字开头。

呼叫

指示报警是否应被呼叫的读/写属性。如果其值为 1 (TRUE)，则呼叫该报警。缺省参数值为 0 (FALSE)。请参见 [报警呼叫属性](#)。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paging”来读取或修改。

呼叫组

指示报警所属呼叫组的只读文本字符串。最大长度为 80 个字符。有关如何使用呼叫组字符串的信息，请参见第三方呼叫系统的说明文档。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paginggroup”来读取。例如，将 PagingGroup 的值指定为变量：

```
myString = myCluster.Alarm_1.paginggroup
```

另请参见

[使用自定义报警过滤器](#)

高级报警

当 Cicode 表达式更改时，高级报警会成为当前报警。该表达式将以 Citect.ini 参数 [Alarm]ScanTime 设定的速率进行轮询，并用于测试结果中包含的更改。如果出现这种情况，就会出现报警状态通知。

配置高级报警：

1. 在**系统 | 高级报警**中配置。此时将出现“高级报警”对话框。
2. 输入报警属性。
3. 点击**添加**以附加新记录，或者点击**替换**以修改现有记录。

另请参见

[高级报警属性](#)

高级报警属性

[高级报警](#)具有下列属性。

报警标签

报警的名称（最多为 79 个字符）。报警名称对于集群而言必须是唯一的。报警标签名称必须遵循 [标签名称语法](#)。如果该名称不唯一或语法不正确，则可能无法识别。如果您具有许多标签，请使用命名约定（请参见 [使用结构化标签名称](#)）。这样一来，可以更方便地查找和调试标签。

注意： 尽管集群名称可以保留为空，但该名称对于所有已定义的集群必须是唯一的。

集群名称

运行此报警的集群的名称（最多 16 个字符）。如果未设置集群名称，则此报警将在所有已定义的集群上运行。

报警名称

与该报警相关联的物理设备的名称（最多为 79 个字符）。这个属性是可选的。只有当报警的详细信息显示在屏幕上或记录到设备中时，才使用此属性。

报警描述

报警的描述（最多为 127 个字符）。其中包括变量数据。这个属性是可选的。只有当报警的详细信息显示在屏幕上或记录到设备中时，才使用此属性。

表达式

触发报警的 Cicode 表达式（最多为 254 个字符）。每当该表达式的结果变为 TRUE 时，就会触发相应的报警。

分类

报警分类号或标签（最多为 16 个字符）。

这个属性是可选的。如果没有指定分类，则该报警的缺省值为分类 0。

延迟

高级报警的延迟期间（最多 16 个字符）。

如果触发报警的 Cicode 表达式的结果在延迟期间内始终为 TRUE，则高级报警就会变为当前报警。当前报警打开的时间是表达式为真时所返回的值。

此属性是可选的。如果没有指定值，则当触发表达式变为 True 后，报警就会马上变为当前报警。

注意：延迟期间必须采用 HH:MM:SS（时:分:秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

帮助

显示 AlarmHelp() 函数何时被用户自定义命令所调用的图形页面的名称（最多为 64 个字符）。这个属性是可选的。如果未指定帮助页面，则调用 AlarmHelp() 函数时不会出现任何动作。

注释

任何有用的注释（最多为 48 个字符）。

扩展形式的字段

以下字段通过扩展形式实现（按 F2）。

权限

操作人员确认或禁用报警所需的权限（最多为 16 个字符）。

注意：如果您为报警分配了确认权限，请不要对确认报警的命令分配某个权限。如果对命令分配了不同的权限，那么操作人员必须同时拥有两个权限才能对该命令进行确认。

区域

报警所属的区域（最多为 16 个字符）。如果某个操作人员无权访问某个区域，则报警在报警显示页面上不可见。例如，如果您进入此处的区域 1，则操作人员必须具有对区域 1 的权限（以及任何所需的其它权限）才能确认或禁用此报警。

自定义过滤器 1...自定义过滤器 8

用于过滤当前报警的用户定义的字符串（最多为 64 个字符）。

在自定义 Cicode 查询中用作搜索条件。通过自定义报警过滤器表达式，操作员可以识别并显示当前报警的子集。

注意：

- 只有在“数字报警”表单在扩展模式下打开时，自定义过滤器才可见。
- 这些字段不区分大小写，而且可以包含 "A"..."Z"、"a"..."z"、"0"..."9"，以及下划线 "_"。
- 自定义过滤器不能以数字开头。

呼叫

指示报警是否应被呼叫的读/写属性。如果其值为 1 (TRUE)，则呼叫该报警。缺省参数值为 0 (FALSE)。请参见 [报警呼叫属性](#)。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paging”来读取或修改。

呼叫组

指示报警所属呼叫组的只读文本字符串。最大长度为 80 个字符。有关如何使用呼叫组字符串的信息，请参见第三方呼叫系统的说明文档。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paginggroup”来读取。例如，将 PagingGroup 的值指定为变量：

```
myString = myCluster.Alarm_1.paginggroup
```

另请参见

[使用自定义报警过滤器](#)

带时间标签的数字量报警

带时间标签的数字量报警和带时间标签的模拟量报警与其他报警类型不同，它们不依赖于变量的轮询来确定报警条件。其工作方式为，使用 Cicode 函数 [AlarmNotifyVarChange](#) 通知报警服务器对指定变量的任意值进行了更改。

报警服务器将使用此信息更新监视该变量的报警。此过程允许使用正确的时间戳与报警条件相关联。

此过程将更新带时间标签的数字量报警的**变量标签 A** 和**变量标签 B**。

事件趋势可配合使用带有时间标签的数字量报警，为趋势和报警数据提供毫秒精确度。请参见 [TrnEventSetTable](#) 和 [TrnEventSetTableMS](#)。

对某个带时间标签的数字量报警进行配置：

1. 选择**报警 | 带时间标签的数字量报警**。此时将出现“带时间标签的数字量报警”对话框。
2. 以显示的格式完成属性。
3. 单击**添加**按钮附加新记录或单击**替换**（如果已对记录进行了修改）。

注意： 要使带时间标签的数字量报警正常工作，您必须配置 Cicode 函数 [AlarmNotifyVarChange](#)，它会通知报警服务器对相关变量的任意值进行了更改。它还可

以传递通知信息的时间标签。报警服务器维护一个通知信息的队列并运行单独的任务来检查此队列。

另请参见

[带时间标签的数字量报警属性](#)

带时间标签的数字量报警属性

[带时间标签的数字量报警](#)具有以下各项属性：

报警标签

报警的名称（最多 79 个字符）。报警名称对于集群必须是唯一的。报警标签名称必须遵循[标签名称语法](#)。如果该名称不唯一或语法不正确，则可能无法识别。如果具有许多标签，请使用命名约定（参见[使用结构化标签名称](#)）。这样就可以更方便地查找和调试标签。

注意： 尽管集群名称可以保留为空，但该名称对于所有已定义的集群必须是唯一的。

集群名称

运行此报警的集群的名称（最多 16 个字符）。如果未设置集群名称，则此报警将在所有已定义的集群上运行。

报警名称

与该报警相关联的物理设备的名称（最多为 79 个字符）。此属性为可选属性。只有当报警的详细信息显示在屏幕上或记录到设备中时，才使用此属性。

报警描述

报警的描述（最多为 127 个字符）。可以包括变量数据。此属性为可选属性。只有当报警的详细信息显示在屏幕上或记录到设备中时，才使用此属性。

变量标签 A/变量标签 B

触发报警的数字量变量（标签）（最多为 254 个字符）。可以根据一个或两个数字量变量的状态将带时间标签的数字量报警配置为激活状态。

如果只使用一个变量来触发报警，则请使用**变量标签 A** 字段。例如：

变量标签 A

RFP3_TOL

在变量 RFP3_TOL 的状态更改为 ON (1) 时将触发报警。

或者，也可以将报警定义为当变量的状态更改成 OFF (0) 时触发，方法是在数字量地址之前使用逻辑操作符 NOT，例如：

变量标签 A	NOT RFP3_TOL
--------	--------------

在这种情况下，报警在变量 MCOL304 的状态更改为 OFF (0) 时被触发。

可以根据两个数字量变量的状态将数字量报警配置为激活状态，例如：

变量标签 A	RFP3_TOL
变量标签 B	NOT MCOL304

在这种情况下，报警在两个变量的状态都更改为激活状态时被触发：即变量 RFP3_TOL 的状态更改为 ON (1)，且变量 MCOL304 的状态更改为 OFF (0)。

注意： 如果将**变量标签 B** 的属性保留为空，则只有变量标签 A 触发报警。

分类

报警分类号或标签（最多 16 个字符）。此属性为可选属性。如果没有指定分类，则该报警的缺省值为分类 0。

延迟（小时：分钟：秒）

报警延迟周期（最多 16 个字符）

当触发条件的状态在延迟周期阶段保持为真时，带时间标签的数字量报警变成激活。状态变为真时，激活报警具有 ON 时间。

此属性为可选属性。如果不指定延迟周期，则报警将在被数字量标签触发时立即激活。

注意： 延迟期间必须采用 HH:MM:SS（时:分:秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

帮助

显示 AlarmHelp() 函数何时被用户自定义命令所调用的图形页面的名称（最多为 64 个字符）。此属性为可选属性。如果未指定帮助页面，则调用 AlarmHelp() 函数时不会出现任何动作。

注释

任何有用的注释（最多 48 个字符）。

扩展形式的字段

以下字段通过扩展形式实现（按 **F2**）。

权限

操作员确认或禁止报警所需的权限（最多 16 个字符）。

注意： 如果为报警分配了确认权限，则不要为确认报警的命令分配权限。如果对命令分配了另一个的权限，那么操作员必须同时拥有两个权限才能对该命令进行确认。更重要的是，此区域定义的内容可能会被忽略。

区域

报警所属的区域（最多 16 个字符）。如果操作员无权访问某个区域，则报警在报警显示中不可见。例如，如果进入 **区域 1**，则操作员必须具有对区域 1 的权限（以及任何所需的其他权限）才能确认或禁用此报警。

注意： 此处定义的区域和权限字段必须配合使用。例如，在按钮上定义的权限将忽略报警定义的区域。

自定义过滤器 1...自定义过滤器 8

用于过滤激活报警的用户定义的字符串（最多 64 个字符）。

在自定义 Cicode 查询中用作搜索条件。通过自定义报警过滤器，操作员可以识别并显示激活报警的子集。

注意：

- 只有“数字报警”表单在扩展模式下打开时，自定义过滤器才可见。
- 这些字段不区分大小写，而且可以包含“A”..“Z”、“a”..“z”以及下划线“_”。
- 自定义过滤器不能以数字开头。

呼叫

指示报警是否应被呼叫的读/写属性。如果其值为 1 (TRUE)，则呼叫该报警。缺省参数值为 0 (FALSE)。请参见 [报警呼叫属性](#)。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paging”来读取或修改。

呼叫组

指示报警所属呼叫组的只读文本字符串。最大长度为 80 个字符。有关如何使用呼叫组字符串的信息，请参见第三方呼叫系统的说明文档。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paginggroup”来读取。例如，将 PagingGroup 的值指定为变量：

```
myString = myCluster.Alarm_1.paginggroup
```

另请参见
[使用自定义报警过滤器](#)

带时间标签的模拟量报警

带时间标签的数字量报警和带时间标签的模拟量报警与其他报警类型不同，它们不依赖于变量的轮询来确定报警条件。其工作方式为，使用 Cicode 函数 [AlarmNotifyVarChange](#) 通知报警服务器对指定变量的任意值进行了更改。

然后，报警服务器将使用此信息更新监视该变量的所有报警。此过程允许使用正确的时间戳与报警条件相关联。

此过程用于更新带有时间标签的模拟量报警的**变量标签**和**设置点**。

事件趋势可配合使用带有时间标签的模拟量报警，为趋势和报警数据提供毫秒精确度。请参见 [TrnEventSetTable](#) 和 [TrnEventSetTableMS](#)。

对某个带时间标签的模拟量报警进行配置：

1. 选择**报警 | 带时间标签的模拟量报警**。此时将出现“带时间标签的模拟量报警”对话框。
2. 输入报警属性。
3. 点击**添加**附加新记录，或点击**替换**修改现有记录。

注意：要使带时间标签的模拟量报警正常工作，您必须配置 Cicode 函数 [AlarmNotifyVarChange](#)，它会通知报警服务器对相关变量的任意值进行了更改。它还可以传递通知信息的时间标签。报警服务器维护一个通知信息的队列并运行单独的任务来检查此队列。

另请参见
[带时间标签的模拟量报警属性](#)

带时间标签的模拟量报警属性

[带时间标签的模拟量报警](#)具有以下各项属性：

报警标签

报警的名称（最多 79 个字符）。报警名称对于集群必须是唯一的。报警标签名称必须遵循 [标签名称语法](#)。如果该名称不唯一或语法不正确，则可能无法识别。如果具有许多标签，请使用命名约定（参见 [使用结构化标签名称](#)）。这样就可以更方便地查找和调试标签。

注意：尽管集群名称可以保留为空，但该名称对于所有已定义的集群必须是唯一的。

集群名称

运行此报警的集群的名称（最多 16 个字符）。如果未设置集群名称，则此报警将在所有已定义的集群上运行。

报警名称

与该报警相关联的物理设备的名称（最多为 79 个字符）。此属性为可选属性。只有当报警的详细信息显示在屏幕上或记录到设备中时，才使用此属性。

变量标签

触发报警的模拟量变量（标签）（最多为 254 个字符）。

设置点

触发偏差报警时所确定的模拟量变量标签或基值（最多 254 个字符）。此属性为可选属性。如果没有指定设置点，则缺省值为 0（零）。

高高限

用作高高限报警的触发条件的值（最多为 11 个字符）。如果在高高限延迟期间内变量标签的值超过此值，则高高限报警将成为当前报警。当标签超过高高限值时，当前报警处于打开状态。

由于高报警必须先于高高限报警，所以当高高限报警被触发时将会替代高报警。如果需要在报警画面上同时显示某个模拟量报警的不同状态，则应当对每一种状态单独配置一个报警。（每一个报警将对同一个标签进行监视。）

高高限延迟

高高限报警的延迟期间（最多 16 个字符）。只有在此时间段内满足报警的触发条件时，该报警才会激活。

此属性为可选属性。如果没有设置值，则在该标签超出高高限值时将激活高高限报警。

注意：延迟期间必须采用 HH:MM:SS（时:分:秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

高限

用作高限报警的触发条件的值（最多为 11 个字符）。如果在高限延迟期间内变量标签的值超过此值，则高限报警将为当前报警。当标签超过高限值时，当前报警处于打开状态。

高限延迟

高限报警的延迟期间（最多 16 个字符）。只有在此时间段内满足报警的触发条件时，该报警才会激活。

此属性为可选属性。如果没有设置值，则在该标签超出高限值时将激活高限报警。

注意：延迟期间必须采用 HH:MM:SS（时:分:秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

如果在高限延迟期间内标签值从高限值增加到高高限值，则延迟计时器将被重置。只有在该值始终处于延迟期间的高高限值范围内时，才会激活高高限报警。

如果在高限延迟期间过期后该值从高限值增加到高高限值，则高限报警将被激活，随后高高限报警的延迟期间即开始。

如果该标签值超出高高限值，随后在高高限延迟期间过期后又降至该值以下，则将立即触发高限报警。当标签值超过高高限值时，当前报警处于打开状态。

这些点还应用于在低限范围与低低限范围之间徘徊的标签值。

低限

用作低限报警的触发条件的值（最多为 11 个字符）。当变量标签的值降至此值之下且在低限延迟期间内始终低于该值时，低限警报将为当前警报。当标签降至低限值以下时，当前报警处于打开状态。

低限延迟

低限报警的延迟期间（最多 16 个字符）。只有在此时间段内满足报警的触发条件时，该报警才会激活。

此属性为可选属性。如果没有设置值，则在该标签降至低限值以下时将激活低限报警。

注意：延迟期间必须采用 HH:MM:SS（时:分:秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

低低限

用作低低限报警的触发条件的值（最多为 11 个字符）。当变量标签的值降至此值之下且在低低限延迟期间内始终低于该值时，低低限警报将为当前警报。当标签降至低低限值以下时，当前报警处于打开状态。

由于低限报警必须先于低低限报警，所以当低低限报警被触发时将会替代低限报警。如果需要在报警画面上同时显示某个模拟量报警的不同状态，则应当对每一种状态单独配置一个报警。（每一个报警将对同一个标签进行监视。）

低低限延迟

低低限报警的延迟期间（最多 16 个字符）。只有在此时间段内满足报警的触发条件时，该报警才会激活。

此属性为可选属性。如果没有设置值，则在该标签降至低低限值以下时将激活低低限报警。

注意：延迟期间必须采用 HH:MM:SS（时:分:秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

偏差

用作偏差报警的触发条件的值（最多为 10 个字符）。当变量标签的值在偏差延迟期间内始终处于偏差范围（由设置点确定）以外时，就会激活偏差报警。

此属性为可选属性。如果用户没有指定某个偏差量，将不会触发偏差报警。

偏差延迟

偏差报警的延迟期间（最多 16 个字符）。只有在此时间段内满足报警的触发条件时，该报警才会激活。

此属性为可选属性。如果没有设置值，则在变量标签处于偏差范围以外时将立即激活偏差报警。

注意：延迟期间必须采用 HH:MM:SS（时:分:秒）格式输入。该值必须介于 0 秒 (00:00:00) 与 24 小时 (24:00:00) 之间。

速率

通过将此值除以报警期间，可以确定变量标签值变化的“最大速率”（最多为 10 个字符）。每次扫描时，都会检查该标签的值。如果其变化速率大于该“最大速率”，将会触发一个变化速率报警报警。

例如，要将某个储存罐不会过快地填满的机会最小化，用户可以配置一个变化速率报警，采用的速率为 300 升，[Alarm]Period 为 60 秒，[Alarm]ScanTime 为 1 秒。这意味着所允许的最大变化速率为 5 升/秒（300 升/60 秒）。在每一个扫描时间间隔内对实际的变化速率进行计算。也就是说，每一秒都检查储存罐的当前液面并同前一秒所记录的液面进行比较。如果实际的变化速率是（比方说）8 升/秒，将会立即触发一个变化速率报警。

此属性为可选属性。如果用户没有指定某个数值，将不会激活变化速率报警。

死区

偏差报警处于不活动状态之前**变量标签**必须返回的值（最多为 10 的字符）。

格式

在图形页面中进行显示、写入到某个文件中或者传递到某个（等待的参数为某个字符串的）函数（变量的）显示格式（最多为 11 的字符）。

此属性为可选属性。如果用户没有指定某个格式，该格式的缺省值为对变量标签所指定的格式。

分类

报警分类号或标签（最多为 16 个字符）。此属性为可选属性。如果未指定分类，则报警将缺省为分类 0。

帮助

显示 AlarmHelp() 函数何时被用户自定义命令所调用的图形页面的名称（最多为 64 个字符）。此属性为可选属性。如果未指定帮助页面，则调用 AlarmHelp() 函数时不会出现任何动作。

注释

任何有用的注释（最多 48 个字符）。

扩展形式的字段

以下字段通过扩展形式实现（按 F2）。

权限

操作员确认或禁止报警所需的权限（最多 16 个字符）。

注意： 如果为报警分配了确认权限，则不要为确认报警的命令分配权限。如果为命令分配了不同的权限，则操作员必须同时拥有两个权限才能确认该命令。

区域

报警所属的区域（最多 16 个字符）。如果操作员无权访问某个区域，则报警在报警显示中不可见。例如，如果进入此处的区域 1，则操作人员必须具有对区域 1 的访问权限（以及任何所需的权限）才能确认或禁止此报警。

自定义过滤器 1...自定义过滤器 8

用于过滤激活报警的用户定义的字符串（最多 64 个字符）。

在自定义 Cicode 查询中用作搜索条件。通过自定义报警过滤器，操作员可以识别并显示激活报警的子集。

注意：

- 只有“数字报警”表单在扩展模式下打开时，自定义过滤器才可见。
- 这些字段不区分大小写，而且可以包含“A”..“Z”、“a”..“z”以及下划线“_”。
- 自定义过滤器不能以数字开头。

呼叫

指示报警是否应被呼叫的读/写属性。如果其值为 1 (TRUE)，则呼叫该报警。缺省参数值为 0 (FALSE)。请参见 [报警呼叫属性](#)。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paging”来读取或修改。

呼叫组

指示报警所属呼叫组的只读文本字符串。最大长度为 80 个字符。有关如何使用呼叫组字符串的信息，请参见第三方呼叫系统的说明文档。此属性可以使用报警标签来读取，或在启用了标签属性的情况下使用名称“myCluster.myAlarm.paginggroup”来读取。例如，将 PagingGroup 的值指定为变量：

```
myString = myCluster.Alarm_1.paginggroup
```

另请参见
[使用自定义报警过滤器](#)

设置报警显示格式

显示格式指定了报警在报警以及报警汇总画面中的显示格式。有关如何更改报警汇总画面所列报警顺序的详细信息，请参见 [更改报警汇总显示顺序](#)。

另请参见
[包括数据](#)
[包括固定文字](#)
[显示列表和表格](#)
[报警信息中的变量数据](#)
[报警显示字段](#)
[报警汇总字段](#)

包括Vijeo Citect数据

通过为每个要显示的字段指定字段名和宽度可将数据包括在报警显示中。用户必须将每一个字段包括在一对大括弧 {} 中，并且按照下面的语法格式：

```
{<field name>, [width[, justification]]}
```

例如：

```
格式          {Tag,8}{Name,32}
```

在这个例子中，数据显示在两个字段中：**Tag**，8 个字符宽度；**Name**，32 个字符宽度。对宽度的指定是可选的；如果用户没有使用该属性，字段的宽度将取决于括弧中字符的个数。

格式	报警的名称:{Name }
----	---------------

在这个例子中，Name 后面跟有四个空格；数据 {Name} 将采用 8 个字符的宽度进行显示。

注意：在报警画面中显示的总字符数（以及由此产生的字段数）是由用户计算机的屏幕分辨率决定的。

另请参见
[包括固定文字](#)

包括固定文字

用户可以通过对文本信息进行指定来将需要显示的固定文本进行显示；例如：

格式	报警的名称：
----	--------

用户在一个文本字符中使用的任何空格都将被包含在显示中。

另请参见
[显示列表和表格](#)

显示列表和表

可以采用对齐标识符的方法来对每一个字段中的文本进行对齐。您可以使用三个对齐字符，L（左）、R（右）和 N（无）；例如：

格式	报警的名称:{名称,32,R} {标签,8,L}
----	--------------------------

对齐标识符是可选的；当忽略该属性时，字段为左对齐。在使用对齐标识符时，您必须同时使用宽度标识符。

要在列中显示字段文本，请使用 TAB 字符 (^t)，例如：

格式	{标签,8}^t{名称,32}^t{描述,8}
----	-------------------------

在使用比例字体显示报警时，此格式会对报警的标签、名称和描述字段进行对齐。

另请参见
[报警消息中的变量数据](#)

报警信息中的变量数据

数字、高级和时间戳记报警的**报警描述**字段可以被用来显示变量数据。可以将某个表达式（变量标签、函数等）嵌入到**报警描述**字段的文本中去。在报警执行的过程中将会计算该表达式的值，并且当产生报警时及时地返回该时刻所有变量标签的数值。

把表达式包含在括弧中以将变量数据同静态文本区分开来。例如：

报警描述	Line Broken Alarm at Line Speed {LineSpeed1}
------	--

当 *LineSpeed1* 为某个变量标签时，该表达式将会在报警显示或者报警日志中产生如下的输出：

Line Broken Alarm at Line Speed 1234

下面的报警条目中采用的是一个表达式而不是标签：

报警描述	High Level at Total Capacity {Tank1+Tank2+Offset()}
------	---

当 *Tank1* 和 *Tank2* 都为变量标签，而 *Offset* 为某个 *Cicode* 函数时，该表达式将产生下面的输出：

最大总容量 4985 升

注意：结果的格式是根据该表达式中第一个变量标签所指定的格式来决定的。可以采用标准变量格式标示符来对数字变量的格式进行定义，并对变量标签中指定的默认格式进行替换。

另请参见
[报警显示字段](#)

报警显示字段

您可以使用下列任何字段或 [报警摘要字段](#) 来设置报警显示（请参阅 [报警类别](#)）和报警记录设备（请参阅 [设置报警显示格式](#)）的格式：

域名	描述
{Tag,n}	报警标签 注意： 如果将 标签 字段配置为支持长名称（最多为 79 个字符），可能会导致报警显示出现重叠情况。如果预计使用长名称，请使用较小的显示字段。
{TagEx,n}	带有群集名称前缀的报警标签 注意： 如果将 TagEx 字段配置为支持长名称（最多为 79 个字符），可能会导致报警显示出现重叠情况。如果预计使用长名称，请使用较小的显示字段。
{AlarmType,n}	报警类型（字符串），未本地化。值为：数字量、模拟量、高级、多-数字量、多状态模拟量、带时间戳的、带时间戳的数字量、带时间戳的模拟量。
{TypeEnum,n}	报警类型编号（改用 AlarmType 获取字符串值）。值为： -1 无效 0 数字量 1 模拟量 2 高级 3 多-数字量 4 多状态模拟量 5 用户事件 6 带时间戳的 7 硬件 8 带时间戳的数字量 9 带时间戳的模拟量
{AlmComment,n}	在报警属性对话框的“注释”字段中输入的文本信息。
{Cluster,n}	集群名称
{CUSTOM1,n} {CUSTOM2,n} {CUSTOM3,n} {CUSTOM4,n} {CUSTOM5,n} {CUSTOM6,n} {CUSTOM7,n} {CUSTOM8,n}	配置的报警自定义字段。
{Name,n}	报警名称 注意： 如果将 名称 字段配置为支持长名称（最多为 79 个字符），可能会导致报警显示出现重叠情况。如果预计使用长名称，请使用较小的显示字段。
{Native_Name,n}	表达式中的报警名称 注意： 如果将 Native_Name 字段配置为支持长名称（最多为 79 个字符），可能会导致报警显示出现重叠情况。如果预计使用长名称，请使用较小的显示字段。
{Desc,n}	报警描述
{Native_Desc,n}	报警描述采用本地语言

域名	描述
{Category,n}	报警分类
{Help,n}	帮助页面
{Area,n}	区域
{Priv,n}	权限
{Priority,n}	报警分类的优先级
{Type,n}	报警或条件的类型: ACKNOWLEDGED CLEARED DISABLED UNACKNOWLEDGED
{LocalTimeDate,n}	采用以下格式的报警日期和时间: "yyyy-mm-dd hh:mm:ss[.ttt]"
{Time,n}	报警状态更改的时间 (hh: mm: ss)。(设置 [Alarm]SetTimeOnAck 参数以便在确认报警的时间内使用此字段。)
{Date,n}	报警状态更改的日期 (dd: mm: yyyy)。请注意, 您可以通过参数 [ALARM]ExtendedDate 来更改使用的格式。
{DateExt,n}	采用扩展格式的报警状态更改的日期。
{State,n}	报警的当前状态。该字段只能用于报警显示。而不适用于报警汇总。 ON OFF
{Millisec,n}	向 {Time,n} 字段中添加毫秒数
{High,n}	高限报警触发值
{HighHigh,n}	高高限报警触发值
{Low,n}	低限报警触发值
{LowLow,n}	低低限报警触发值
{Rate,n}	触发值的变化速率
{Deviation,n}	偏差报警触发值
{Deadband,n}	死区
{Format,n}	变量标签的显示格式
{Value,n}	模拟量变量的当前值
{State,n}	报警的当前状态。该字段只能用于报警显示。而不适用于报警汇总。 DEVIATION RATE

域名	描述
	LOW LOWLOW HIGH HIGHHIGH CLEARED
{ErrDesc,n}	与某种协议（通讯）错误相关联的文本字符串。该字段仅与硬件错误相关联并且包含与该错误相关联的额外信息（例如，如果该错误与某个设备相关联，将返回该设备的名称；如果该错误与某个 Cicode 函数相关联，将返回该函数的名称；如果该错误与某个 I/O 设备相关联，将返回该 I/O 设备的提醒信息）。
{ErrPage,n}	与该报警相关联的页面、设备等。
{LogState,n}	报警所经过的上一个状态。（将报警记录到某个设备时，此状态非常有用。）
{State_desc, n}	特定状态的配置描述（例如，运行状况或已停止）。配置多-数字量报警属性时输入此描述
{Paging,n}	指示是否应该呼叫该报警。如果其值为 TRUE ，则呼叫该报警。缺省参数值为 FALSE 。请参见 报警呼叫属性 。
{PagingGroup, n}	指示报警所属的呼叫组。最大长度为 80 个字符。

n 指定显示字段的大小。

注意：

- 任何类型的报警都可显示上述任一字段。对于不适用的特殊报警类型，则会显示零或空白字符。
- 当某个报警值的长度大于要在其中显示的字段长度（例如，n）时，它将会被截断或者用 #OVR（“格式宽度溢出”）提醒信息来替代。
- 对于汇总页面，则应使用 {SumState}。要将状态记录到某个设备，使用 {LogState}。State 是报警的当前状态，SumState 是报警发生时的状态，Log State 是报警在过渡时期的状态。

另请参见
[报警汇总字段](#)

报警汇总字段

可以使用下面列出的任何字段（或组合）来对报警汇总显示以及报警汇总设备的格式进行设置。

通过在“报警分类属性”对话框中的**摘要格式**字段中指定字段名称，对整个报警分类的报警汇总进行格式设置。

还可以使用 [Alarm]DefSumFmt 参数来设置报警汇总的格式，尤其是在所有的报警汇总格式都相同的时候。

字段名	描述
{UserName,n}	已登录并对报警执行了某些操作（例如确认报警或禁用报警等）的用户的名称（用户名）。请注意，当报警第一次被激活时，用户名被设置为“系统”（因为操作人员未对报警实施跟踪）。
{FullName,n}	已登录并对报警执行了某些操作（例如确认报警或禁用报警等）的用户的全名（用户全名）。请注意，当报警第一次被激活时，用户名被设置为“系统”（因为操作人员未对报警实施跟踪）。
{UserDesc,n}	与用户事件相关联的文本。
{OnDate,n}	报警被激活的日期
{OnDateExt,n}	报警被激活的日期（采用扩展的格式 dd/mm/yyyy）
{OffDate,n}	报警返回到其正常状态的日期
{OffDateExt,n}	报警返回到其正常状态的日期（采用扩展的格式 d/mm/yyyy）
{OnTime,n}	报警被激活的时间
{OffTime,n}	报警返回到其正常状态的时间
{DeltaTime,n}	OnDate/OnTime 与 OffDate/OffTime 之间的时间差（秒）
{OnMilli,n}	向报警激活时间中添加毫秒数。
{OffMilli,n}	向报警返回到其正常状态的时间中添加毫秒数。
{AckTime,n}	报警被确认的时间
{AckDate,n}	报警被确认的日期
{AckDateExt,n}	报警被确认的日期（采用扩展的格式 dd/mm/yyyy）
{SumState,n}	对报警发生时的状态进行描述
{SumDesc,n}	对报警汇总的描述
{SumType,n}	报警汇总类型（类似于报警“类型”）。值为 ACKNOWLEDGED、CLEARED、DISABLED、UNACKNOWLEDGED
{Native_SumDesc,n}	报警汇总描述（采用本地语言）
{Comment,n}	操作人员在运行时间向“报警汇总”条目中添加的注释。该注释是使用 AlarmComment() 函数来指定的。
{Native_Comment,n}	操作人员在运行时间向“报警汇总”条目中添加的本地语言注释。

其中 n 指定显示字段的大小。

注意：您还可以在“报警汇总”中包含**状态**之外的其它任何报警显示字段。

另请参见
[更改报警汇总显示的顺序](#)

改变报警摘要的显示顺序

可以使用 `SummarySort` 和 `SummarySortMode` 参数自定义报警在报警摘要画面上的显示顺序。`SummarySort` 参数允许按照 `OnTime`、`OffTime` 和 `AckTime` 显示报警。`SummarySortMode` 确定报警是以升序还是以降序排列。（使用这些参数设定的顺序将会替代在报警优先级中设定的顺序。）

另请参见
[报警显示的格式设置](#)

将报警属性作为标签使用

可以使用变量标签的场合也同样可以使用报警属性（报警描述除外）。例如，报警 `CV110_FAULT` 激活时，可以为操作员提供直观的指示。报警处于激活状态时，`CV110_FAULT.On` 为 `TRUE`；报警处于非激活状态时，`CV110_FAULT.On` 为 `FALSE`。例如，`CV110_FAULT.On` 在图形对象中作为填充颜色表达式输入。传送带出现故障时，图形对象会改变颜色。

要将报警属性用作标签，必须按以下格式对其进行设置：报警标签（例如 `CV100_STOP`），后跟句号 (.)，然后是属性（例如 `分类`）。完整的报警属性会变为 `CV100_STOP.Category`。

注意：如果打算将时间戳数字量或时间戳模拟量报警属性用作变量标签，则需要确保正确设置这些报警属性，将所需数据通过 `Cicode` 函数 `AlarmNotifyVarChange` 推入相关变量。

有关如何操作这些报警的详细信息，请参见 [时间戳数据量报警](#) 和 [时间戳模拟量报警](#)。

另请参见
[支持的报警属性](#)
[报警属性的写入](#)
[设置报警](#)

支持的报警属性

以下的属性可用于所有报警类型。请记住，返回的数值是同描述相关的。例如，对于某个数字量，如果返回值为 1，则意味着描述为 `TRUE`，而 0（零）意味着 `FALSE`。

属性	描述	返回类型
.On*	报警激活	数字量
.确认	报警被确认	数字量
.Custom1 .Custom2 .Custom3 .Custom4 .Custom5 .Custom6 .Custom7 .Custom8	自定义字段。	字符串（64 个字节）
.Disabled	报警被禁用（参见下面的注释）	数字量
.Time	报警被触发时的 32 位时间值	长型
.Tag	报警标签	字符串（80 个字节）
.Name	报警名称	字符串（80 个字节）
.Category	报警分类	整型
.Paging	报警呼叫	布尔
.PagingGroup	报警所属的呼叫组。	字符串（80 个字节）
.Priority	报警优先级	整型

* 当与某个报警标签相关的任何报警被激活时，模拟量报警的 .On 属性将变为 true。

注意：报警一旦被禁用，只有使用 AlarmEnable() 或 AlarmEnableRec() 函数才能将其重新启用。

对于数字量报警、带时间标签的报警和高级报警，还可以使用以下属性：

属性	描述	返回类型
.Desc	报警描述	字符串（128 个字节）
.Delay	报警延迟	长型

注意：所有报警类型都有描述，但描述无法为虚拟量报警或带时间标签的虚拟量报警返回有意义的数据。描述将返回一个指示读取是否成功的空字符串；写入表示该标签已被解析，该写入请求已发送。

对于虚拟量报警和带时间标签的虚拟量报警，还可以使用以下属性：

属性	描述	返回类型
.Value	报警标签数值	实型
.设定值	设置点	实型
.HighHigh	高高限	实型
.High	高限	实型
.LowLow	低低限	实型
.Low	低限	实型
.DeadBand	死区	实型
.Rate	速率	实型
.Deviation	偏差	实型
.HHDelay	高高限延迟	长型
.HDelay	高限延迟	长型
.LDelay	低限延迟	长型
.LLDelay	低低限延迟	长型
.DevDelay	偏差延迟	长型

对于下面的数字量属性，每个报警在任何时刻只有一个可以为 **true**。它们的优先级从最低到最高排列如下所示：

.DVL	偏差报警被触发（低限）	数字量
.DVH	偏差报警被触发（高限）	数字量
.R	变化速率报警被触发	数字量
.L	低限报警被触发	数字量

.H	高限报警被触发	数字量
.LL	低低限报警被触发	数字量
.HH	高高限报警被触发	数字量

注意：只有当偏差 > 0 的时候才计算 DVL 和 DVH 的值。只有在速率 > 0 的时候才计算 R 的值。

某些报警属性返回配置数据。如果用户还未定义此信息，系统将提供以下默认值：

属性	默认值
.Setpoint	0
.HighHigh	3.4e+38
.High	3.4e+38
.LowLow	-3.4e+38
.Low	-3.4e+38
.Rate	0
.Deviation	0
.Deadband	0
.Category	0
.Priority	0

另请参见
[写入报警属性](#)
[设置报警](#)

报警属性的写入

如果拥有相应的权限，用户可以对下列报警权限进行写入。（请记住，用户所写入属性的数值是同描述相关的。例如，如果用户将某个数字量报警的属性设置为 1，相当于将其描述变为 TRUE。如果用户将其设置为 0（零），相当于将其描述变为 FALSE。）

属性	描述	输入类型
.确认	报警已被确认（一旦经确认将不会变为“未被确认”）	数字量
.死区	报警死区	实型
.偏差	与设定值之间的偏差	实型
.已被禁用	报警被禁用	数字量
.高高限	高高限	实型
.高限	高限	实型
.低低限	低低限	实型
.低限	低限	实型
.高高限延迟	高高限延迟	长型
.高限延迟	高限延迟	长型
.长型延迟	低限延迟	长型
.长长型延迟	低低限延迟	长型
.偏差延迟	偏差延迟	长型
.Custom1 .Custom2 .Custom3 .Custom4 .Custom5 .Custom6 .Custom7 .Custom8	自定义字段	字符串
.Paging	报警呼叫	布尔
.PagingGroup	报警所属的呼叫组。	字符串

模拟量报警的阈值可以通过 `AlarmSetThreshold()` 函数进行改变。

请注意以下几点：

- 报警标签必须是唯一的。
- 在报警服务器以及控制客户机上所有包含工程中的报警数据库必须是相同的。

另请参见

[支持的报警属性](#)

[设置报警](#)

设置报警属性

要使用报警属性，必须在警报服务器上启用它们。

1. 在“工程编辑器”中，选择**服务器 | 警报服务器**。
2. 按 F2 显示扩展表单。
3. 将“发布报警属性”设置为 TRUE，并指定警报服务器将监听的端口。

另请参见

[警报服务器定义](#)
[支持的报警属性](#)
[报警属性的写入](#)

处理运行时的报警

一旦被触发，报警将进入活动状态。数字量报警的活动状态为“开启”，而模拟量报警的活动状态将根据报警类型的不同（例如“高限”、“低低限”、“速率”等）而有所不同。当操作人员对报警进行确认后，其状态变为“已确认”。当报警被复位后（引发报警的条件被修正后），其状态将变为“关闭”。

报警显示在标准报警显示画面上。要确认报警，操作员可以用鼠标选择报警并点击鼠标左键，也可以将光标移动到报警上并按 **Enter** 键。此外，操作员可以通过点击**报警确认**确认所有报警。一旦被确认，报警显示在屏幕上的颜色将改变。得到确认的报警继续保留在屏幕上，直至其状态变为“关闭”。

如果报警的工作与设计不符，或被认为没有必要，则操作员可以禁用它。在被重新启用前，已禁止的报警将被忽略。（您必须定义一个使用 `AlarmDisable()` 函数禁止报警的命令。）

为保存报警活动历史，将保留所有报警的事件日志。在该日志中储存了每个报警被激活、确认以及复位的时间。您可以在报警汇总画面上显示该事件日志中的所有的报警（包括被禁止的报警）。

创建一个基于 `AlarmSummary` 模板的名为**摘要**的页面。您应将页面命名为“摘要”以便报警摘要按钮（在比如菜单页面这样的页面上）工作正常（`PageSummary()` 函数只能打开名为“摘要”的页面）。

操作人员可以为汇总日志中的任何报警添加注释。（要向报警中添加注释，您必须使用 `AlarmComment()` 函数来定义命令。）

注意：如果报警画面或报警汇总画面上有许多报警，则使用上一页和下一页来滚动浏览此列表。

要创建报警画面，请执行以下操作：

1. 在 **Citect 浏览器**中，双击**图形 | 画面**文件夹中的**创建新画面**。

- 或者 -

2. 在**图形编辑器**中，选择**文件 | 新建**，然后点击**画面**。
3. 选择要使用的报警模板。使用**报警**模板创建用于显示可配置报警的画面，使用**汇总**模板创建用于显示汇总报警的画面，使用**已禁止**模板创建用于显示已禁止报警的画面，使用**硬件**模板创建用于显示硬件报警的画面。
4. 选择**文件 | 保存**。
5. 在画面标题域中指定一个名称。新画面的名称应与模板名称一致。例如，将新硬件报警画面称为**硬件**。
6. 点击**确定**。

注意：您还可以创建自己的（非标准）报警画面。最简单的办法是在标准报警模板中进行复制和修改。

要在运行时显示报警画面，请执行以下操作：

1. 如果尚未在工程中创建报警（或硬件报警）画面，则创建一个。应将可配置的报警画面称为**报警**，将硬件报警画面称为**硬件**。
2. 创建一个新的键盘命令或按钮，以在运行时调用该画面。您还可以向现有屏幕对象中添加触摸命令。
3. 在命令域中，输入 **PageAlarm()**（以显示可配置报警画面）或 **PageHardware()**（以显示硬件报警画面）。
4. 根据需要配置其它属性。
5. 点击**添加**附加新记录，或点击**替换**修改现有记录。

注意：如果使用的是标准画面模板，通常无需创建画面显示命令：这些命令已内置在其中。

要显示自定义报警画面（使用非标准名称），使用 `PageDisplay()` 函数显示该画面，如果需要，可以在后面加上 `AlarmSetInfo()` 函数。

使用系统字体

可以通过报警分类和 `Cicode` 函数使用预定义的字体（即系统字体）显示文本。也可以自行配置字体；有关详细信息，请参见 [配置自定义颜色字体](#)。

注意：如果与 I/O 关联的动画在启动时没有正确初始化或出现脱机，而系统仍在运行，则关联的动画将在相关图形页面中灰显（因为参数值无效）。可以通过 `[Page]ComBreak` 参数禁止此功能。请参见 [处理报表中的通讯错误](#)。

如何定义系统字体：

1. 在工程编辑器或图形编辑器中，选择**系统 | 字体**。将显示“字体”对话框。
2. 输入字体属性。
3. 单击**添加**附加新记录，或单击**替换**修改现有记录。

另请参见
[字体属性](#)

字体属性

使用“字体”对话框定义字体属性。字体具有下列属性：

字体名

字体的名称（最多 16 个字符）。与背景文本、字符串和数字（这些可以使用标准的 Windows 字体）不同，动画文本必须要定义字体。

字体类型

Windows 所支持的任何文本字体（最多 31 个字符）。字体类型从菜单中选择。

还可以指定加粗、斜体和下划线文本。要指定以上任意选项，只需为字体类型附加适当的定义符，例如：

字体类型	Courier,B
------	-----------

指定加粗字符。

字体类型	Helv,I
------	--------

指定斜体字符。

字体类型	TmsRmn,U
------	----------

指定带下划线的文本。

还可以指定多个选项，例如：

字体类型	Courier,B,I,U
------	---------------

指定加粗、斜体和带下划线的字符。

注意：要使用菜单中未显示的字体，必须先要在计算机上安装该字体，才能在系统中使用。要查看、安装或移除 Windows 字体，请使用 Windows 控制面板（字体选项）。请参考 Windows 说明文档了解详细信息。（如果您的系统使用了网络，所有计算机均必须安装该字体。）

像素大小

所显示文本的大小（最多 16 个字符）。可以用像素或者点数来指定文本字体。

要指定点大小，请在**像素大小**域中输入负数，例如：

字体大小

-10

指定一个 10 点的字体大小。请注意点大小只能采用整数来指定。

如果没有在系统中安装字体类型（或者像素大小），系统将采用同指定的字体以及/或者大小最接近的字体。

如果使用的是点大小，该字体的大小在所有的屏幕分辨率下都保持不变。在低分辨率屏幕上，字体会比在高分辨率屏幕上看起来大，因此可能会导致动画不对齐。在相同分辨率的计算机屏幕上应使用点大小。

前景颜色

所显示文本的前景颜色（即，文本字符的颜色）（最多 16 个字符）。可以使用预定义的颜色标签（通过菜单访问）、用户定义的自定义工具，或由函数 `MakeCitectColour` 生成的 RGB 编码的数字（请参见 *Cicode 参考指南*）。

注意：请勿混淆预定义标签与颜色选择器关联的颜色 **名称** 功能。不能通过此对话框使用颜色名称，这样会产生编译错误。

前景闪烁

对文本字符使用闪烁颜色时应用到字体的二级颜色（最多 16 个字符）。可以使用预定义的颜色标签（通过菜单访问）、用户定义的自定义工具，或由函数 `MakeCitectColour` 生成的 RGB 编码的数字（请参见 *Cicode 参考指南*）。

如果不指定颜色，文本将保持实线。

背景颜色

所显示文本的背景颜色（最多 16 个字符）。可以使用预定义的颜色标签（通过菜单访问）、用户定义的自定义工具，或由函数 `MakeCitectColour` 生成的 RGB 编码的数字（请参见 *Cicode 参考指南*）。

这个属性是可选的。如果不指定背景颜色，默认为透明色。

背景闪烁

当使用闪烁颜色时应用到背景的二级颜色（最多 16 个字符）。可以使用预定义的颜色标签（通过菜单访问）、用户定义的自定义工具，或由函数 `MakeCitectColour` 生成的 RGB 编码的数字（请参见 *Cicode 参考指南*）。

如果不指定颜色，文本将保持实线。

注释

任何有用的注释（最多为 48 个字符）。

配置自定义颜色字体

从 Vijeo Citect 版本 6.0 开始，颜色将通过使用红绿蓝 (RGB) 值的十六进制代码进行引用。通过在“图形编辑器”中选择工具 | **编辑喜爱的颜色**，您可以查看选定颜色的 RGB 值（以十进制数形式显示）。

例如，如果要使用的颜色的值是 R = 128, G = 170, B = 213，您可以将这些值转换为对应的十六进制形式（R = 0x80, G = 0xAA, B = 0xD5）。

如果为颜色定义了标签，则可以使用十进制或十六进制值。例如：

- **标签名称:** Plate_Blue
- **表达式:** 0x80AAD5（或 8432341）

定义了标签之后，便可以创建在工程中使用的字体，以用于报警分类、Cicode 函数和按钮对象。例如，使用“字体”对话框定义具有下列属性的字体：

- **字体名:** Plate_Font
- **字体类型:** Arial
- **像素大小:** 12
- **前景颜色:** Plate_Blue

有关详细信息，请参考知识库文章 Q4024。

另请参见

[使用系统字体](#)

Chapter: 24 配置事件

您可以使用事件来触发一个动作，例如一个命令或一组命令。例如，当某个生产过程完成之后通知操作人员，或者当生产过程达到某个特定的步骤时执行一系列的指令。

事件必须被激活后才能运行。使用 *计算机设置向导*（自定义设置）来激活事件。如果使用网络，则可以在任何计算机（或所有计算机）上处理事件。

注意： 事件系统不冗余。即，不可能分配主要和备用事件系统。如果需要冗余的事件报告系统，则应使用报表。参见 [报表信息](#)。

如何定义一个事件：

1. 选择**系统 | 事件**。将出现“事件”对话框。
2. 输入事件属性。
3. 点击**增加**加入新的记录，或点击**替换**修改现有的记录。

注意： 必须首先激活事件服务器，然后事件才能起作用。

另请参见
[事件属性](#)
[运行事件](#)

事件属性

[事件](#)具有以下属性：

名称

对于一个单一计算机系统，将事件名指定为**全局**：

名称	全局
----	----

如果使用网络，并希望所有计算机上运行一个事件，请为该事件名称指定**全局**。如果只希望在特定计算机上运行一个事件，请指定事件名称，并使用“计算机设置向导”（自定义设置）来指定哪些计算机将运行此事件。事件名称不一定是唯一的，您可以使用相同名称指定多个事件。

输入不超过 16 个字符的值。

注意：事件名称必须符合 [标签名称语法](#)。使用任何其它字符（如空格）都将导致编译器出错。使用下划线字符 (`_`) 代替空格。

集群名称

在事件操作和事件触发 Cicode 运行时要使用的缺省集群（最多 16 个字符）。如果这些 Cicode 表达式中的任何标签在表达式内没有指定其集群，它们都将被解析为此缺省集群。此字段在单集群系统中可以保留空白，或表达式中的每一个标签都声明为 `<ClusterName>.<TagName>` 格式，此字段也可保留空白。

时间

一天中同步**周期**的时间，格式为 `hh:mm:ss`（小时:分钟:秒）。如果没有指定时间，**周期**将在 `00:00:00` 同步（也就是午夜）。输入不超过 32 个字符的值。

周期

检查事件的周期，格式为 `hh:mm:ss`（小时:分钟:秒）。输入不超过 32 个字符的值。或者您可以指定：

- 输入一周中的某一天，例如星期一、星期二、星期三等，在这一天检查事件，并以此指定每一周的周期。
- 输入一个月中的某一天，例如 1 号、2 号、3 号、4 号、5 号等，在这一天检查事件，并以此指定每一个月的周期。
- 输入一年中的某月和日期，例如与 1 月 1 号和 2 月 25 号等等，在这一天检查事件，并以此指定每年的周期。月份与日期必须用空格分开。

如果没有指定周期或时间，缺省的周期将是一秒。如果没有指定周期，但指定了时间，缺省的周期将是一天。

触发器

用来确定是否执行事件操作的 Cicode 表达式（或变量标签）。每秒钟检查一次此表达式。输入不超过 254 个字符的值。

动作

要执行的命令。输入不超过 64 个字符的值。这些命令将在下列情形中执行：

- 当指定的**时间**和**周期**发生，**触发器**条件为 `TRUE` 或空白。
- 当**触发器**变为 `TRUE`，而**时间**和**周期**域为空白。触发器必须变为 `FALSE`，然后重新变为 `TRUE`，这样动作才能重新执行。

注释

任何有用的注释。输入不超过 48 个字符的值。

运行事件

必须首先激活事件服务器，然后事件才能工作。可以在以下情况下自动运行事件：

- 在指定时间和周期中。
- 触发器条件变为 TRUE 时。
- 触发器条件在指定时间和周期中为 TRUE。

另请参见

[指定时间和周期](#)

[使用触发器](#)

指定时间和周期

周期确定事件何时运行。用户可以采用 `hh:mm:ss`（小时:分钟:秒）形式指定周期，例如：

周期	注释
1:00:00	每小时运行事件
6:00:00	每六小时运行事件
72:00:00	每三天运行事件
星期一	每星期一运行事件
15 号	每月十五日运行事件
6 月 25 日	6 月 25 日运行事件

还可指定同步事件的具体时间，例如：

周期	注释
6:00:00	在早晨 6:00 点同步事件
12:00:00	在中午 12:00 点同步事件

时间同步每天运行事件的时间，周期确定事件运行的时间，例如：

时间	6:00:00
周期	1:00:00

在这个例子中，报表每小时运行一次。如果早晨 7: 25 启动 Runtime 系统，事件将在早晨 8: 00 运行，然后每隔一小时运行一次。

另请参见
[使用触发器](#)

使用触发器

您可以使用任何 Cicode 表达式（或变量标签）作为事件的触发器。如果表达式结果（在**触发器**字段中）变为 TRUE，并且**时间**和**周期**字段为空，事件将运行。例如：

时间	
周期	
触发器	RCC1_SPEED<10 AND RCC1_MC

只有当表达式（触发器）变为 TRUE 时，此事件才会运行，也就是说，当数字标签 RCC1_MC 是 ON 而模拟标签 RCC1_SPEED 小于 10 时。表达式必须首先变为 FALSE 然后再变为 TRUE，此时事件才能再次运行。

如果使用“时间”和/或“周期”字段，触发器将在指定的“时间”和/或“周期”被选中，例如：

时间	6:00:00
周期	1:00:00
触发器	RCC1_SPEED<10 AND RCC1_MC

此事件每小时运行一次，但前提必须是表达式（触发器）是 TRUE（也就是说，数字标签 RCC1_MC 是 ON 而模拟标签 RCC1_SPEED 小于 10）。

另请参见
[运行事件](#)

Chapter: 25 使用累积器

累积器用于跟踪递增的运行时数据，如发动机运行时间、功率消耗及关闭时间。可设置触发器（例如电机开）对三个计数器递增：

- 累积器的触发次数（例如，发动机的启动次数）。
- 以秒为单位的运行时间。
- 根据定义的增量得到的累计值（例如，当前值）。

累积的数据作为变量标签存储在 I/O 设备中。变量标签在启动时读取，并在触发器激活时定期更新。可以通过动画显示、趋势显示或记录变量标签来监视和显示累积的数据。

注意： 可以使用 `AccControl()` Cicode 函数在运行时控制（重新读取或重置）任何累积器。

如何配置累积器：

1. 选择**系统 | 累积器**。将显示“累积器”对话框。
2. 输入累积器属性。
3. 单击**添加**附加新记录，或单击**替换**修改现有记录。

可以使用“累积器属性”对话框配置累积器。

另请参见

[累积器属性](#)

累积器属性

累积器具有下列属性：

名称

累积器的名称。输入一个不超过 79 个字符的值。

集群名称

运行累积器的集群的名称（最多 16 个字符）。如果未设置集群名称，则此累积器将在所有已定义的集群上运行。

触发器

Cicode 表达式（或变量标签）可触发累积器。如果此字段中的表达式的结果为 `TRUE`，则累积启动。如果此字段中的表达式的结果变成 `FALSE`，则累积停止。输入一个不超过 254 个字符的值。

触发器检查频率由 `[Accumulator]WatchTime` 参数加以控制。

运行时间

包含运行时间（以秒为单位）的变量（标签）。输入一个不超过 254 个字符的值。启动时，将读取此值。然后，此变量的本地副本将递增（当**触发器**为真），且变量将按照 [Accumulator]UpdateTime 参数确定的频率写回 I/O 设备。

启动次数

包含启动次数（即触发器从 FALSE 改为 TRUE 的次数）的变量（标签）。输入一个不超过 254 个字符的值。启动时，将读取此值。然后，此变量的本地副本将递增，且变量将按照 [Accumulator]UpdateTime 参数确定的频率写回 I/O 设备。

累加器增量

“**触发器**”条件为 TRUE 时加到“**累加器**”变量的任意 Cicode 表达式（或变量标签）。输入一个不超过 254 个字符的值。

累加器

包含累加值的变量（标签）。输入一个不超过 254 个字符的值。

启动时，将读取此值。每次检查触发器且触发器为 TRUE 时，“**累加器增量**”字段中的值都被添加到此“**累加器**”变量的本地副本中。新的“**累加器**”变量按照 [Accumulator]UpdateTime 参数确定的频率写回 I/O 设备。

例如,如果您为一个发动机配置了一个累积器, 并且**累加器增量**是发动机使用的电流（安培），则**运行时间**将包含发动机运行的时间（以秒为单位），**启动次数**将包含发动机启动的次数，**累加器**将包含发动机使用的电流总计。发动机使用的平均电流为**累加器/运行时间**。

注释

任何有用的注释。输入不超过 48 个字符的值。

扩展形式的字段

以下字段通过扩展形式实现（按 **F2**）。

权限

操作员对累积器执行操作所需的权限（通过使用累积器函数）。输入不超过 16 个字符的值。

区域

累积器所属的区域。只有有权访问此区域且具备所有必需权限的用户可以对此累积器执行操作。例如，如果您进入了此处的区域 1，则操作员必须有权访问区域 1 且具备所有必需的权限）才能对此累积器执行操作。输入不超过 16 个字符的值。

注意：

- 运行时间、启动次数和累加值存储在 I/O 设备的变量中。这样一来可以方便地访问这些变量，而且由于这些变量保存在 I/O 设备中，因此在项目关闭时将保存这些变量的

值。通过这些变量存储在磁盘 I/O 设备中，可以提高系统性能。（如果将这些变量存储在外部 I/O 设备中，则更新这些变量时将消耗通讯带宽。

- 如果使用网络，则可以使用冗余的磁盘（虚拟）I/O 设备来保护这些变量。
- 累积器服务器作为报表服务器的一部分运行。如果您有一个冗余的报表服务器，则必须使用主/从配置来停止在两个报表服务器上运行的累积器。使用“计算机设置向导”来定义报表服务器。
- 您可以使用 `AccControl() Cicode` 函数在运行时间控制（重新读取或重置）任何累积器。

Chapter: 26 记录和趋势化数据

自 Vijeo Citect 版本 6.0 起，过程分析器（一个内置的趋势可视化工具）已经取代了趋势图的功能。不过，仍然支持趋势图表。请注意，如果打算使用 SPC 趋势，则应该使用趋势图表，因为过程分析器不支持 SPC。

有关详细信息，请参过程分析器帮助。

另请参见

[趋势化数据](#)

[趋势图表](#)

[打印趋势数据](#)

[导出趋势数据](#)

[使用趋势历史文件](#)

[使用路径替换](#)

[调试趋势](#)

趋势化数据

趋势系统可以帮助您更好地了解工厂和设备的性能。趋势化可以用来进行可视性动态分析（趋势以及 SPC 图线）、生产记录、或者为了提高效率以及预防性维护而定期记录设备状态。

使用趋势标签，可以指定要从 I/O 设备变量收集的数据。可以定期间隔（周期趋势）记录此信息，或只在发生事件（事件趋势）时记录。事件趋势用于对不依赖于时间的数据进行趋势记录，例如，当某个产品离开生产线时。趋势数据通常被保存在磁盘中以对其进行分析，或者在某个趋势图线中进行显示。

趋势系统通常是基于实时采样的。每次对数据进行采样时，趋势系统将获得一个数据点。尽管数据点之间的空隙可以被填充，用户应当确认现场设备能够按照指定的速度返回数据（尤其当用户采用的采样周期小于 100ms 时）。

可以收集并存储任意数量的数据。用户计算机的硬盘大小是所能存储的数据量的唯一限制（一种非常有效的数据存储方法就是确保用户硬盘空间利用率为最大）。对于长期的数据保存，用户可以将数据归档到磁盘或者磁带上（而不必影响正在运行的系统）。为了提高存储效率，用户应当采用压缩的形式对趋势文件进行保存。

注意： 如果通过网络来对数据进行趋势化（分布式处理机制），我们推荐用户使用 [时间同步](#) 来启用时间同步。这可确保将所有趋势都同步到同一时间。

用户还可以采用 [Trend]StaggerRequestSubgroups 参数来对趋势采样请求进行交错排列。

另请参见
[配置趋势标签](#)

配置趋势标签

可以使用“趋势标签”对话框配置您的趋势标签。

如何配置趋势标签：

1. 选择**标签 | 趋势标签**。将出现“趋势标签”对话框。（按 **F2** 键查看扩展的趋势标签表格。）
2. 输入趋势标签属性。
3. 点击**增加**加入新的记录，或点击**替换**修改现有的记录。

另请参见
[趋势标签属性](#)

趋势标签属性

趋势标签名称

分配给趋势数据的名称（最多 79 个字符）。如果用趋势标签来对某个变量进行记录，应使用同与其相关的 32 个字符的变量名称相类似的 16 个字符的名称。这样用户就能够很容易地看出这两者之间的联系。此名称对于集群必须是唯一的。趋势标签名称必须遵循 [标签名称语法](#)。如果该名称不唯一或语法不正确，则可能无法识别。如果具有许多标签，请使用命名约定（参见 [使用结构化标签名称](#)）。这样就可以更方便地查找和调试标签。

注意： 尽管集群名称可以保留为空，但该名称对于所有已定义的集群必须是唯一的。

注意： 趋势标签名称应唯一，而且不与运行此趋势的集群中的任何 SPC 标签名称相同。访问同一文件的两个标签可能导致包含趋势/SPC 数据丢失或损坏的系统错误。



警告

不希望的设备操作

如果使用“文件名”属性，请确保趋势标签使用唯一的名称。访问同一文件的两个标签可能导致包含趋势/SPC 数据丢失或损坏的系统错误。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

集群名称

运行此趋势的集群的名称（最多 16 个字符）。如果未设置集群名称，则此趋势将在所有已定义的集群上运行。

表达式

趋势标签中记录的数值（最多 254 个字符）。用户可以利用变量标签对每个变量进行记录。例如：

表达式	LT131
注释	记录变量标签 LT131

过程变量 LT131 的值被写入日志。

用户还可以对任何 Cicode 表达式或者函数进行记录，例如：

表达式	LT131/COUNTER
注释	记录变量标签 LT131 除以变量标签 COUNTER。

注意： 当在趋势标签属性的表达式字段中使用变量标签时，该标量标签的**工程零点刻度**和**工程满刻度**字段设置必须适当，否则数据将会丢失，因为趋势将负值记录为无效。

触发器

对数据记录进行触发的 Cicode 表达式（或者变量标签）（最多 254 个字符）。例如：

触发器	LT131<50
-----	----------

在这个例子中，当变量标签（LT131）的值下降到小于 50 时，日志记录将开始进行。

对于周期性趋势，只有当触发器的值为 TRUE 的时候数据才被写入日志。（当触发器的值为 FALSE 时，趋势图线仍然会滚动显示，此时将会显示 <GATED> 字样）。在以上例子中，LT131 的值一直低于 50，所以连续记录数据。当值上升至（或高于）50 时，停止记录。只有当 LT131 值低于 50 时，才再次开始记录。

用户不必为某个周期性趋势指定一个触发器。如果用户没有对某个周期性趋势指定触发器，则日志记录将会连续进行。

对于事件趋势，一旦触发器的值从 FALSE 变化到 TRUE，数据将被记录到日志中。在上面的例子中，当 LT131 的值第一次下降到 50 以下时，一个采样值被写入到日志中。而在

LT131 上升到（或者高于）50 并且再次下降到 50 以下之前，不会对另一个采样值进行记录。

采样周期

数据的采样周期（最多 16 个字符）。用户可以输入自己的周期，或者从菜单中选择一个。

当采样周期大于一秒时，键入时应当采用 hh:mm:ss（小时：分钟：秒）的形式。如果用户键入一个不带冒号（:）的单个数字，该数字将被作为秒数来对待。例如，如果键入 2，该数字将被看作是 2 秒。

小于一秒的采样周期必须被作为小数的形式输入。例如，要键入 200 毫秒的周期，用户应当键入 0.2。

I 如果采样周期小于一秒，则一秒必须能够被该周期整除（商是整数）。例如，0.05 的采样周期是有效的，因为 $1/0.05=20$ ，而采样周期为 0.3 是无效的，因为 $1/0.3 = 3.333\dots$ 。

请注意以下几点：

- 用户的 I/O 设备必须能够按照指定的速度来提供数据，否则在趋势数据中将会出现空隙并且/或者将会引起**趋势采样丢失**的硬件报警。用户可以利用 [Trend]GapFillTime 参数来填充文件以及图线中的空隙。在图线中的空隙只能采用 TrnsetDisplayMode() 函数进行填充。
- 如果某个采样周期小于一秒的趋势被网络上的多个客户机（分布式处理机制）所共享，用户应使用 [时间同步](#) 配置应用程序来启用时间同步。这可以确保各个趋势之间保持同步。

每个示例周期都将检查**触发器**。如果触发器的值是 TRUE（或者在事件趋势的情况下从 FALSE 变为 TRUE），**表达式**的值将写入到日志中。

范例

采样周期	注释
30	每 30 秒将数据写入日志
10:00	每 10 分钟将数据写入日志
10:00:00	每 10 小时将数据写入日志
2:30:00	每 2.5 小时将数据写入日志

画面上最快的趋势的采样周期将作为该画面的显示周期默认值。

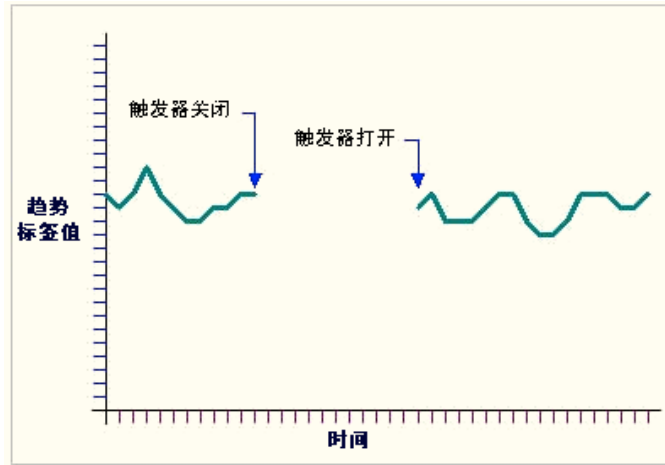
此属性为可选属性。如果用户没有对采样周期进行指定，则采样周期的默认值是 10 秒。

注意： 如果您在现有工程中编辑此属性，则必须先删除相关的趋势文件，然后再运行新的运行系统（有关趋势文件的位置，请参阅“文件名称”）。

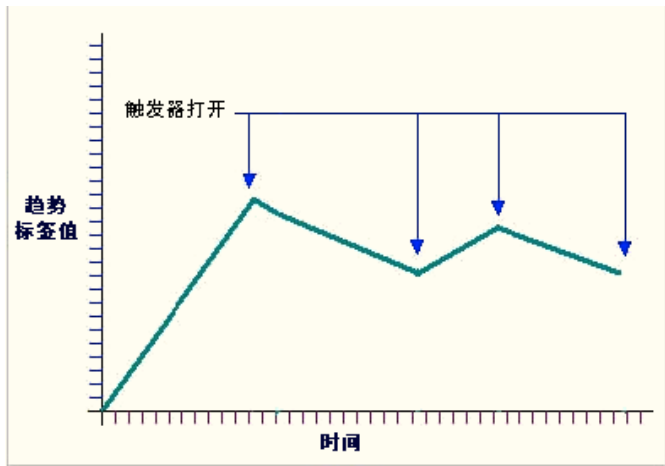
类型

趋势类型（最多 32 个字符）：

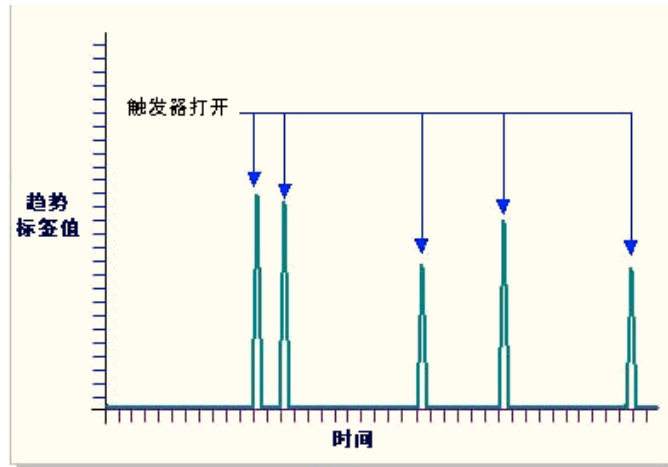
- **TRN_PERIODIC** - 按指定周期进行连续采样的趋势。您也可以定义一个触发条件来停止和启动趋势（当工厂内发生特定条件时）。



- **TRN_EVENT** - 每次触发器的值从 FALSE 更改为 TRUE 时进行采样的趋势。内插发生在每个数据点间，以创建连续图线。



- **TRN_PERIODIC_EVENT** - 每次触发器的值从 FALSE 更改为 TRUE 时进行采样的趋势。这些点间不发生内插，因此将在每个数据点添加图线。



注释

任何有用的注释（最多为 48 个字符）。

扩展形式的字段

以下字段通过扩展形式实现（按 **F2**）。

文件名

保存数据所用的文件（最多 253 个字符）。必须指定完整的路径或者使用替代路径。

从现场收集数据时，会将其存储在计算机硬盘上的文件中，该计算机稍后将用于显示趋势或 SPC 图线（将针对每个趋势标签使用独立文件）。

默认情况下，Vijeo Citect 将该文件保存在安装 Vijeo Citect 的硬盘上的 [DATA] 目录中。文件的缺省名称是趋势标签名称。但是，用户可以另外指定一个文件名称，如：

文件名	[DATA]:TANK131
-----	----------------

其中 [DATA] 指定了数据所在的磁盘以及路径。路径替换使得用户的工程更加“便捷”。

注意：

- 您无法在 bin、runtime、backup 或 user 目录或者任何替代目录中存储趋势文件。如果用户有已经存在的 3.xx 或者 4.xx 版本的工程使用这些目录保存趋势文件，则这些趋势文件的路径必须被改变为 DATA 目录。
- 趋势系统将会先缓冲采集到的数据，然后再保存到某个文件中。[Trend]CacheSize 参数可确定存放返回数据的缓冲区大小的。
- 文件名称属性是可选的。如果未指定文件名，文件名默认位于安装 Vijeo Citect 的硬盘上的 [DATA]:<Name> 下。其中 <Name> 是趋势标签名称。

如果使用“文件名”属性，请确保没有任何其他 SPC 标签或趋势标签使用相同的文件名。访问同一文件的两个标签可能导致包含趋势/SPC 数据丢失或损坏的系统错误。



不希望的设备操作

如果使用“文件名”属性，请确保趋势标签使用唯一的文件名称。访问同一文件的两个标签可能导致包含趋势/SPC 数据丢失或损坏的系统错误。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

- 指定文件名称时，请不要使用文件扩展名。如果用户在某个已经存在的工程中对该属性进行编辑（改动文件名称或者路径），所有已经存在的趋势数据将会被忽略。

存储方式

选择**标度**或**浮点**（64 个字符）。标度为 2 字节数据存储方式，浮点使用 8 字节。

同标度存储方式相比，浮点存储方式明显地扩大了数据的范围，使得数值具有更高的分辨率。然而，用户需要考虑的是这种方式比前者需要更大的磁盘空间。标度应当被用于需要使用同 V5.31 之前版本相兼容的历史文件的场合。

如果用户没有对存储方式进行指定，默认的设置是**标度**。

注意： 如果您在现有工程中编辑此属性，则必须先删除相关的趋势文件，然后再运行新的运行时系统。（有关趋势文件的位置，请参见“文件名称”。）

权限

操作人员显示趋势中的趋势数据所需要的权限（最多 16 个字符）。

区域

趋势数据所属的区域（最多 16 个字符）。

工程单位

记录到日志中的变量/表达式的工程单位（最多 8 个字符）。工程单位是在趋势标度以及显示趋势光标时需要使用的。

格式

记录到日志中的变量/表达式的格式（最多 11 个字符）。格式是趋势标度以及显示趋势光标时所需要的。

此属性为可选属性。如果没有指定格式，缺省格式是####.#。

文件数量

在用户的硬盘中所保存的历史文件的个数（对于该标签而言）（最多 4 个字符）。用户可以为每个趋势标签所指定的文件的最大数量为 999。注意，如果每个趋势拥有大量历史文件，则性能和存储将受到严重影响。

如果用户没有对文件个数进行指定，用户的硬盘上将会保存 2 个历史文件。

注意： 如果您在现有工程中编辑此属性，则应先删除相关的趋势文件，然后再运行新的运行系统。（有关趋势文件的位置，请参见“文件名称”。）

时间

一天中对历史文件的开头进行同步的时间，采用 hh:mm:ss（小时:分钟:秒）的形式（最多 32 个字符）。如果您没有指定时间，文件将在 0:00:00 时刻进行同步（即午夜时刻）。

注意： 如果您在现有工程中编辑此属性，则应先删除相关的趋势文件，然后再运行新的运行系统。（有关趋势文件的位置，请参见“文件名称”。）

周期

历史文件的周期，采用 hh:mm:ss（小时:分钟:秒）的形式（最多 32 个字符）。用户还可以采用：

- 输入一周中的某一天，例如星期一、星期二、星期三等，在这一天开始记录历史文件，并以此指定每一周的时间段。
- 输入一个月中的某一天，例如 1 号、2 号、3 号、4 号、5 号等，在这一天开始记录历史文件，并以此指定每一个月的时间段。
- 输入一年中的某月某日，例如 1 月 1 日、2 月 25 日等，在这一天开始记录历史文件。日期和月份之间必须用空格分隔。

如果没有指定时间段，缺省的时间段将是星期日（每周）。

确定周期设置后，请注意，趋势查看器（现有 Vijeo Citect 客户机或过程分析器）的性能可能会受到趋势文件大小的影响。显示基于事件的趋势数据时，尤其会出现这一情况。

注意： 如果您在现有工程中编辑此属性，则应先删除相关的趋势文件，然后再运行新的运行系统。（有关趋势文件的位置，请参见“文件名称”。）

趋势图表

趋势图表可以很直观地表示过去和当前的现场数据，描述怎样更改变量（如产品输出、级别、温度等）或怎样操作设备或进程。用户可以在当前活动发生时对其进行监视，也可以通过滚动条来查看趋势历史。

当变量的数值随时间发生变化时，或者当有事件发生时，画面上的图线也随之移动。最新值按缺省进行显示。用户可以回调历史数据来显示变量（或者生产过程）过去的数值。

用户可以对任何单个变量或者 Cicode 表达式进行趋势记录。用户可以在屏幕上同时显示任意数量的趋势，即使这些趋势具有各不相同的采样周期。用户还可以在任何趋势窗口中显示多达八条的趋势标签（画笔）。

趋势图表只能与一个群集进行通信，因此，您不能将趋势与单个趋势图表上的多个群集相混淆。如果要绘制多个群集的趋势，您需要使用多个趋势图表或过程分析器（没有此类限制）。

即使在不进行显示的时候，历史数据的采集仍然在继续进行。用户可以在画面之间进行切换而不会影响到趋势图线的显示。即使在不进行显示的时候，趋势数据的采集以及存储（在趋势历史文件中）仍然在继续进行。

用户可以采用以下的标准趋势：

- 单个全画面趋势中，一个图形画面中只显示一个趋势窗口。
- 双全画面趋势，一个图表画面中显示两个趋势窗口。
- 可放大趋势，两个趋势窗口以及添加的放大功能。
- 弹出式趋势，用户可以在计算机屏幕的任何位置“弹出”（在另一个窗体中）。
- 用户自定义趋势，可以放在任何图形页面的任意位置上。

注意： 变量标签可以通过 SPC 控制图表来进行趋势显示。统计过程控制（SPC, Statistical Process Control）是一种使得用户能够对材料、生产产品、服务等的质量进行控制的机制。这种质量控制是通过对采集的数据进行汇总、编排、分析、以及测试并对其一致性或者质量的欠缺进行检测来获得的。

另请参见

[创建趋势画面](#)

创建趋势画面

可以使用任何预定义的趋势模板来创建趋势画面，也可以利用预定义模板来生成自己的趋势模板。可以在趋势上绘制趋势背景（如网格线）。

配置趋势画面：

1. 点击**新建画面**，或者选择**文件 | 新建**。
2. 选择**类型:画面**。
3. 选择趋势画面的**屏幕分辨率**（大小）。
4. 为趋势画面选择一个**趋势模板**：
 - **Singletrend** - 画面上只有一个趋势
 - **Doubletrend** - 画面上有两个趋势
 - **Eventtrend** - 画面上有一个事件趋势
 - **Zoomtrend** - 画面上有两个趋势（其中一个窗口用于缩放）
 - **Poptrend** - 画面上只有一个趋势（在弹出窗口中显示）

5. 点击**确定**。

要创建多个趋势画面，您可以：

- 为要显示在运行期间系统中的每个趋势集创建一个趋势画面。
- 创建单个趋势画面并使用 PageTrend() 函数根据需要显示趋势。利用此函数，可以在具有单一趋势画面的系统中显示所有趋势。
- 用图形编辑器创建趋势画面，并将所有的笔名称设置为空白。然后可通过调用函数来显示画面，并传递所需的趋势标签（最多 8 个）。从趋势画面的菜单中调用 PageTrend() 函数。

另请参见
[趋势插值](#)

趋势内插

当采样与显示周期无法保持同步或者当采样值丢失时，采用趋势内插来确定趋势图线的形状。

例如，对于某个趋势来说，在每两次对趋势图线进行刷新之间可能已经进行了五次采样。由于每次刷新时只能有一个数值被显示，因此必须采用最能够代表这五个采样值的数值；这个数值可能是最大值、最小值、或者是一个平均值。

要定义 Vijeo Citect 对该数值的计算方式，用户必须专门设置一个趋势内插显示方法。

下表中给出了几种常用的内插显示方法，分为**压缩式方法**（适用于显示周期长于采样周期的情况）和**扩展式方法**（适用于显示周期小于或者等于采样周期的情况）。

压缩式方法	扩展式方法
平均值 （默认） - 显示在前一个显示周期中获得的采样值的平均值。	分步式 （默认） - 这种方法只显示最近一次采样的值。
最小值 - 显示在前一次显示周期中所出现的最小的采样值。	斜率 - 这种方法采用在发出请求前后最近的时刻以及采样值所形成的斜率来插入一个“直线”值。
最大值 - 显示在前一次显示周期中所出现的最大值。	原始数据 - 该方法显示实际的原始数据值。

内插显示方法是通过 TrmSetDisplayMode() 函数来进行设置的。用户还可以使用 [Trend]GapFillMode 参数，但是这种方法除了在趋势图线中进行内插之外，还将在实际的趋势文件中进行内插。

打印趋势数据

您可以使用以下函数打印趋势数据：

函数	用途
TrnPrint	打印屏幕上所显示的趋势。
TrnPlot	打印一个或者多个趋势标签的图线。
TrnComparePlot	打印两个趋势（一个同另一个相重叠），每个趋势最多可以到四个趋势标签。
WinPrint	打印当前激活的窗口。

标准的趋势模板中有用于调用这些函数（以打印数据）的按钮。

当您使用 TrnPrint 函数进行打印时，将会显示“绘图设置”对话框。使用此对话框执行以下操作：

- 对趋势的标题进行指定。
- 添加一条显示在标题下方的注释。
- 指定是用黑白还是彩色方式对趋势进行打印。您在此做出的选择将会成为 [General]PrinterColorMode 参数的设置。
- 对打印机设置进行定义。用户在这里所选定的打印机将在 [General]TrnPrinter 参数中被设置为默认打印机。
- 指定下一次使用该函数时是否要显示该表格。此复选框设置 [General]DisablePlotSetupForm 参数。

另请参见

[导出趋势数据](#)

导出趋势数据

您可以使用下列函数将趋势数据导出到报表以及数据库中：

函数	用途
TrnGetTable	检索趋势信息并将其保存到某个 Cicode 数组中
TrnExportClip	将趋势数据复制到剪贴板上
TrnExportCSV	将趋势数据复制到 CSV 文件中
TrnExportDBF	将趋势数据复制到 DBF 文件中

标准的趋势模板有按钮来调用这些函数对数据进行导出。

注意：您还可以选择趋势图形中的一部分（点击并且拖动），并将其所对应的数值复制到 Windows 剪贴板。您可以随后将这些数据粘贴到 Excel 电子数据表格中。（如果要粘贴毫秒级数据，您需要在时间（TIME）栏中创建一种用户自定义的格式以正确地显示这些数据。要完成这项工作，请选中该列并选择**格式 | 单元格**。在**数字**分页中，选择**用户自定义**类别，并键入 **h:mm:ss.000 AM/PM**。）

另请参见
[使用趋势历史文件](#)

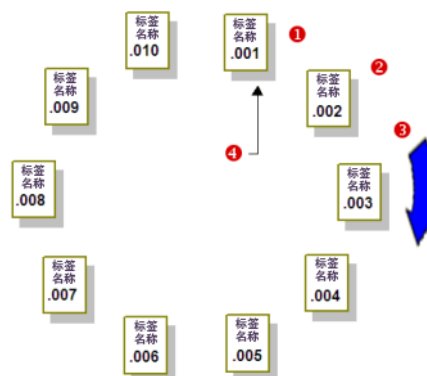
使用趋势历史文件

当 Vijeo Citect 首次启动时，它将创建运行时系统中的每个趋势标签所需要的所有趋势文件。（您可以使用 [Trend]AllFiles 参数更改此缺省设置。）

一个旋转历史文件系统将用于存储趋势数据。数据被记录在几个文件中而不是在一个大的文件中。

缺省情况下，有两个文件（对于每个趋势标签）。用户可以通过对需要使用的文件数量进行指定来更改该默认设置。例如：

文件数量	注释
10	对于数据使用十个文件，如下面的示意图所示。



1. 当 Citect 开始记录时，数据被写入第一个文件。
2. 在下周日的午夜，Citect 写入第二个文件。
3. 在下一个周日的午夜，Citect 写入第三个文件，等等。
4. 在第十周后，第一个文件被新数据覆盖。

对于每一个趋势标签可以指定的文件数量最大可以达到 270。

用户还可以对文件之间的周期进行指定，也就是说什么时候开始使用新的历史文件。例如：

周期	注释
1:00:00	每小时启用一个新的文件
6:00:00	每六个小时启用一个新的文件
72:00:00	每三天启用一个新的文件
星期一	在每周的星期一开始启用一个新的文件
15 号	在每个月的 15 号开始启用一个新的文件
6 月 25 日	在每年的 6 月 25 日开始启用一个新的文件

用户还可以指定一天中对历史文件进行同步的时间，例如：

时间	注释
6:00:00	早上 6:00 对文件进行同步
12:00:00	中午 12:00 对文件进行同步
18:30:00	晚上 6:30 对文件进行同步

另请参见
[存储方式](#)

存储方式

可以选择用于存储趋势标签和 SPC 标签的方式。有**基本范围**或**浮点**两种方式可供选择。基本范围表示一种 2 字节数据存储方式；浮点则使用 8 个字节。

与基本范围存储相比，浮点存储方式明显地扩展了数据的范围，这可以提高值的准确度，但会占用较大的磁盘空间。基本范围应当被用在需要使用同 V5.31 之前版本相兼容的历史文件的场合。

用户可以通过趋势标签或者 SPC 标签属性表来对所需的存储方式进行设置（按 **F2** 键来查看扩展表格）。存储方式的默认设置是基本范围。

另请参见
[计算磁盘存储](#)

计算磁盘存储

利用下面的公式可以计算对指定时间区间内的趋势进行保存所需要的总磁盘空间。

注意： 由于对趋势所采用的存储方式（**基本范围**或**浮点**）会影响每个采样所需的字节数，因此为了正确计算所需的磁盘空间，在计算过程中使用正确的公式非常重要。

在扩展的“趋势标签属性”对话框中可找到对特定趋势所采用的存储方式。（缺省情况下采用**基本范围 (Scaled)** 存储方式。）

基本范围

每个数据采样需要 2 个字节的存储空间。因此，可以使用以下公式计算每个趋势所需的全部磁盘存储空间：

$$\text{每个趋势所需的字节数} = 464 \times \text{文件数} + 176 + \left(\frac{\text{文件周期 (秒)} \times (\text{文件数}) \times 2}{\text{采样周期 (秒)}} \right)$$

例如，如果趋势记录在一周中每十秒钟产生一个采样，并且用户要使用五个数据文件（五周），则所需的字节数为：

$$\text{所需字节数} = 464 \times 5 + 176 + \left(\frac{(7 \times 24 \times 60 \times 60) \times 5 \times 2}{10} \right)$$

$$\text{所需字节数} = 607,296 \text{ 个字节}$$

浮点

每一个采样数据需要八个字节的存储空间。那么前面的公式将变为：

$$\text{每个趋势需要的字节数} = 704 \times \text{文件数} + 160 + \left(\frac{\text{文件周期 (秒)} \times (\text{文件数}) \times 8}{\text{采样周期 (秒)}} \right)$$

因此，上面范例中所需要的字节数为：

$$\text{需要的字节数} = 704 \times 5 + 160 + \left(\frac{(7 \times 24 \times 60 \times 60) \times 5 \times 8}{10} \right)$$

$$\text{需要的字节数} = 2,422,976 \text{ 个字节}$$

注意： 上面的计算没有考虑对每一个趋势的历史文件进行保存所需要的空间。这是因为这些文件始终保持一个固定的大小，因此对于所需要的总的磁盘空间大小没有大的影响。

注意： 为了高效地存储趋势，请使用 Windows 文件压缩。使用这种方法，可以经常将文件大小压缩 10%；实际的压缩量取决于数据的变化率。

另请参见
[重新配置历史文件](#)

重新配置历史文件

如果要对某现有工程中的趋势历史文件的配置进行修改，或者要更改某个影响数字、时间、或趋势文件周期的趋势标签，用户必须删除所有已经存在的趋势文件，然后再运行新的系统。

如果用户修改了某个已经存在的工程中的趋势历史文件，所有已经存在的趋势数据都将被忽略。

注意： 当系统正在运行时，您不能从硬盘中删除（由 Vijeo Citect 创建的）历史文件，因为趋势服务器会尝试重新创建这些文件，此操作可能会导致系统性能问题。

使用路径替换

可以使用路径替换，而不是在系统中指定数据文件的完整路径。

在路径替换中，可定义用来替换完整路径的名称。然后可以使用以下格式的替换命令：

文件名	[SUBSTITUTION]:<文件名>
-----	----------------------

例如，如果打算将趋势数据文件 MYFILE 存储在 C:\CITECT\DATA\MYTRENDS 目录中，则可以指定文件的完整路径，例如：

文件名	C:\CITECT\DATA\MYTRENDS\MYFILE
-----	--------------------------------

或者定义路径替换（例如，MYDATA）并将该路径指定为：

文件名	[MYDATA]:MYFILE
-----	-----------------

路径替换能够更好地控制数据存储。可以通过更改数据路径的定义（而不是逐个查找和更改每个出现的数据路径）来更改所有数据文件的位置。

另请参见
[缺省路径定义](#)

缺省路径定义

Vijeo Citect 具有以下预定义的替代路径：

路径名称	平台	缺省目录
[Bin]	Vista 之前和 Vista	C:/Program Files/Schneider Electric/Vijeo Citect/Vijeo Citect 7.10/Bin
[User]	Vista 之前	Documents and Settings/All Users/Application Data/Schneider Electric/Vijeo Citect 7.10/User
	Vista	ProgramData/Schneider Electric/Vijeo Citect 7.10/User
[Data]	Vista 之前	Documents and Settings/All Users/Application Data/Schneider Electric/Vijeo Citect 7.10/Data
	Vista	ProgramData/Schneider Electric/Vijeo Citect 7.10/Data
[Run]	Vista 之前和 Vista	当前的工程目录
[Copy]	Vista 之前和 Vista	当前的复制工程目录
[Back]	Vista 之前和 Vista	当前的备份工程目录

调试趋势

提供了 TrendDebug citect.ini 参数，以在记录和趋势数据期间为您提供帮助。

例如：

```
[Trend]
TrendDebug=n
```

其中 *n* 可以是以下调试选项的组合：

- 1 - 记录客户机、服务器和冗余消息类型，同时记录以正常方式收集来的被写入趋势服务器的采样。

- 2 - 记录有关当前活动 **backfill** 过程的信息，包括写入档案文件的冗余采样。
- 4 - 记录 **TrendSetTable** 函数的详细信息。
- 7 - 记录所有趋势活动。
- 8 - 仅记录当前活动 **backfill** 过程的摘要信息。

可以将这些设置一起加入，以将各记录层组合在一起。例如

```
[Trend]  
TrendDebug=6
```

记录详细的 **backfill** 过程和 **TrendSetTable** 函数。

动态地读取这些设置，就是说，用户可以在 **Vijeo Citect** 运行时改变这些设置，并且此改变从那点起开始生效。

Chapter: 27 了解统计过程控制

统计质量控制 (SQC) 有助于追踪和提高产品或服务的质量。统计过程控制 (SPC) 是一种主要的 SQC 工具，可用于收集、编排和内插与产品相关的过程变量，从而达到质量的一致性。

每个过程都会有变化，因此始终应进行控制以改进过程一致性和质量。所以需采用一个连续性的改良策略。重复以下循环可执行一个预防策略。这是采用 SPC 的关键。考虑以下步骤：

- 分析过程
- 我们都了解这个过程的哪些可变性？
- 这个过程在统计上是否能控制（可预言性）？
- 这个过程是否符合设置要求？
- 哪些注意事项对达到过程质量标准最为重要？
- 维护（控制）过程
- 改良过程

SPC 包括过程变差、统计控制以及过程性能，通常使用 XRS 控制图表、性能图表以及 Pareto 图表等工具。

另请参见

[过程变差](#)

[统计控制](#)

[过程性能](#)

[XRS 控制图表](#)

[性能图表](#)

[配置性能图表](#)

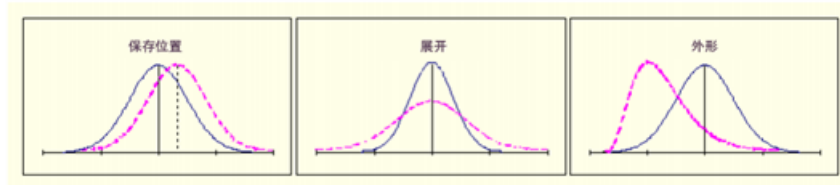
[Pareto 图表](#)

[使用统计过程控制 \(SPC\) Vijeo Citect](#)

过程变差

要有效地使用 SPC，您应该理解变差的概念。当一个产品的特性被反复估量，每次估量和上一次会有所不同。这是因为过程包含了可变的来源。

当数据被组合放入一个频率柱状图，它会形成一种模式。模式作为一个概率分布来参考，并通过以下三种方式表现：



注意：大多数 SPC 技术假设收集的数据拥有正态分布。

变差以以下两种类型中的一种分类：

- **常见：**指随时间的变化，变差是可预测和重复的。分布特性是稳定的。常见变差可能是因为不精确的过程或类似原因造成的。

统计学指出 85% 的过程质量要求偏差是因为常见变差。这些偏差通常需要管理阶层来解决。

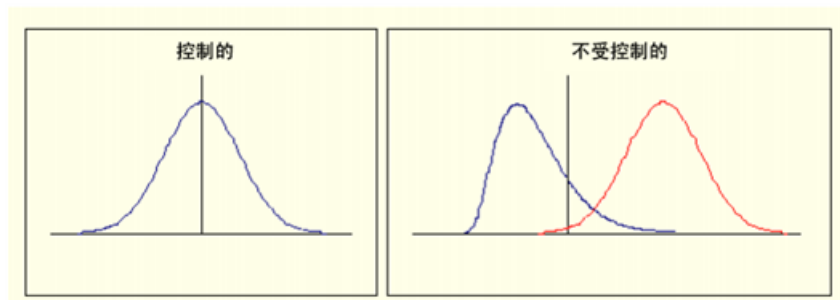
- **特殊：**指变差不是经常发生。当特殊变差发生它会改变分布特性。而且随时间变化，分布不会稳定。

统计学指出 15% 的过程质量要求偏差是因为特殊变差。通常这些偏差要求本地处理（设备修理等）来解决。

另请参见
[过程变差](#)

统计控制

一个过程属于统计控制，通常是指变差来源是由常见变差引起的。一个统计控制过程是用户想要的，因为它是可预测的，而一个非控制统计过程将造成不可预测的分布。



即使一个过程可能不受统计控制，它仍然能符合要求。相反的，一个可控制的过程可能不能符合要求。这种说法被阐明为考虑过程性能。

另请参见
[过程变差](#)

过程性能

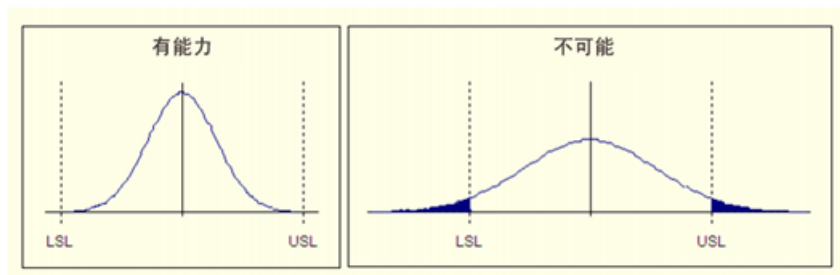
一个过程被称为性能优良是在说明书规定的极限以外的样本的百分比低于一个预定义值。以下假设也必须是真实的：

- 过程在统计上来说是稳定的（只存在常见导致的变差）。
- 单独测量和正态分布一致。
- 测量变差（由于测量工具）是小的。

说明书限制是客户要求的反映并可选择的。样本百分比必须在说明书限制之内，这是从标准偏差（ Σ ）计算得来的“3- Σ ”在平均值的任何一边。

注意：“3 西格玛”是指落在中心的任何一边 3 标准偏差位置的边界。对于一个正态分布，99.74% 的样本应在这个边界之内。

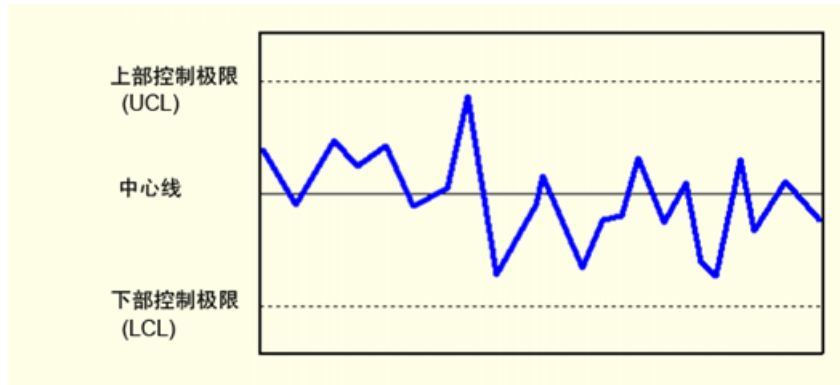
最终性能确定过程在统计上是否符合说明书。减少常见变差的影响会使过程性能更佳，如下所示：



另请参见
[XRS 控制图表](#)

XRS 控制图表

XRS 图表是 SPC 的主要工具，并传送有关变差和控制的信息。图表是趋势图分别显示平均数（X 轴）、差距 (R)、标准偏差 (s 或 sigma)。图中每一点代表子群数据，不是单独样本。



CL、UCL 和 LCL

每个图表都有中心线、控制上限 (UCL) 线和控制下限 (LCL) 线。控制限是根据一个近似正态分布的 "3- Σ " 限制的位置来估计的。在实际操作中，这个限制是从测量的 X 双轴、R 轴和 s 轴计算得到的，图表的值是以使用的子组大小为根据的。

因为这些线是基于过程的自然可变性，所以它们被作为分析的一个标线。这些限制不是规格限。

解释图表

控制图表可提供有关过程变异的重要信息。查阅特殊模式和事件可得出如何控制过程的结论。

一个统计上受控制的（普通）过程会依照一个正态分布来分布数据。数据随机分布，但以中心线为中心。出现在控制极限附近的数据是少有的但是所期望的。出现在控制极限以外的数据表明过程在统计上是不受控制的，所以要采取措施找出原因。同样，出现不变数据或模式指出了不稳定性。

以下模式和事件被考虑为触发报警的原因：

- 控制限制之外的点
- 几个连续点只在平均值的一边
- 几个连续点一直增加或减少
- 多于 2/3 的绘图点接近平均值
- 少于 2/3 的绘图点接近平均值

这些类型模式和事件的存在具有不同意义，这取决于分析的是哪种类型的图表。将三种控制图表一起使用(X 轴, R 和 s)会有更好的理解。

另请参见
[性能图表](#)

性能图表

过程性能图表是所有过程平均值数据的频率柱状图和分布图，用于分析过程性能。此图表根据适当的中心线、规格下限和规格上限指标绘制而成。性能图表通常用 C_p 、 C_{pk} 、峰度和偏斜度指标来解释。

USL 和 LSL

上部说明书极限 (USL) 和下部说明书极限 (LSL) 指出产品的要求 (客户自定义)。理想值应是 USL 和 LSL 之间的中间点。

C_p 指数

固有过程能力 (C_p) 是公差与 6σ 的比率。此指数实质上表示分布是否位于规格上限 (USL) 和规格下限 (LSL) 之间。它的含义定义如下：

- $C_p > 1.0$ - 表示过程变异将在指定的限制 (USL 和 LSL) 内，因此符合能力要求。
- $C_p < 1.0$ - 表示过程不符合能力要求。

C_{pk} 指数

基于最差情况的过程能力，与 C_p 类似。不过，此指数指示平均值相对于规格上限和规格下限而言所在的位置。它从数学上解释了 C_p 。它的含义定义如下：

- $C_{pk} < 0$ - 表示过程平均值在指定限 (USL 和 LSL) 外。
- $C_{pk} = 0$ - 表示过程平均值等于其中一个指定限。
- $C_{pk} > 0$ - 表示过程平均值位于指定限内。
- $C_{pk} = 1.0$ - 表示 6σ 限的一侧位于一个规格限上。
- $C_{pk} > 1.0$ - 表示 6σ 限完全位于指定限内。

另请参见

[Pareto 图](#)

Pareto 图

Pareto 图是用来分析规格或成功标准中偏差相关频率的工具。Pareto 图是一个频率柱状图，按从高到低排序。Pareto 图不同于控制图和性能图，它不使用统计索引。

Pareto 图强调 Pareto 的经验定律，任何事件分类均由少数主要元素和许多次要元素组成。在 SQC 中，从许多不重要的机会中挑选出少数重要的机会来进行改良很重要。Pareto 图对于维修成本和偏差频率的对比分析非常有用。

除柱状图外，通常还会提供一个累计百分比。从顶部到底部，百分比表示到该点所有值的和与图表中所有值的和的比率。

使用统计过程控制 (SPC)

Vijeo Citect 显示有关三种图表类型的统计过程控制信息：XRS 控制图、过程性能图和 Pareto 图。

如何配置 SPC 图：

1. 单击**新建页面**，或选择**文件 | 新建**。
2. 选择**类型：画面**。
3. 选择 SPC 页面的**分辨率**（大小）。
4. 为 SPC 页面选择一个 SPC **模板**：
 - **SPCXRSChart** - XRS 控制图
 - **SPCCpk** - 与平均图结合的性能 (Cpk) 图

注意： Pareto 图的配置稍有不同，因此此处不做介绍。

5. 单击**确定**。
6. 双击图形显示。
7. 在精灵弹出式菜单中输入变量标签。
8. 单击**确定**。
9. 保存页面。

另请参见

[SPC 标签](#)

[SPC 控制图](#)

SPC 标签

SPC 标签指定为在 SPC 操作中使用而收集的数据。当数据被定义后，能在运行时间动态分析（作为 SPC 图表和报警）这些数据。SPC 标签类似于趋势标签。

和趋势类似，Vijeo Citect 能收集和存储任何数量的 SPC 数据。对于能够保存的数据量的唯一限制是用户计算机上硬盘空间的大小。(Vijeo Citect 使用高效的数据存储方法来优化计算机硬盘的存储空间利用。)对于长期的数据保存，用户可以将数据归档到磁盘或者磁带上

（不会影响 Runtime 系统）。用户还可以按周期记录数据（周期趋势），或仅在事件发生时记录（事件趋势），均采用与趋势标签相同的方式。

注意： SPC 不支持周期-事件趋势，它是周期和事件趋势属性的一个组合。另外，如果使用基于事件的 SPC 标签，您的显示屏幕刷新率可能比应用程序的要求或实际刷新率要慢。

如果要执行性能分析，必须在定义 SPC 标签时填充规格上限 (USL) 和规格下限 (LSL)。这些值需精确的表现用户的需求，目标值应介于二者之间。如果这些域保留空白，性能分析就没有意义。

要配置 SPC 标签，请执行以下操作：

1. 选择**标签 | SPC 标签**。将出现“SPC 标签”对话框。
2. 输入 SPC 标签的属性。
3. 点击**添加**附加新记录，或点击**替换**修改现有记录。

另请参见

[SPC 标签属性](#)

SPC 标签属性

使用“SPC 标签”对话框配置 SPC 标签属性。统计过程控制 (SPC) 标签有以下属性：

SPC 标签名称

分配给 SPC 数据的名称（最多 16 个字符）。如果要记录变量，您应该为 SPC 标签使用用于变量标签的名称。

注意： SPC 标签名称应唯一，而且不与任何趋势标签名称相同。访问同一文件的两个标签可能导致包含趋势/SPC 数据丢失或损坏的系统错误。



警告

不希望的设备操作

如果使用“文件名”属性，请确保 SPC 标签使用唯一的名称。访问同一文件的两个标签可能导致包含趋势/SPC 数据丢失或损坏的系统错误。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

集群名称

SPC 标签在其中运行的集群的名称（最多 16 个字符）。

如果未设置集群名称，则 Vijeo Citect 将考虑在所有已定义的集群上运行此 SPC 标签。

表达式

SPC 标签的记录值。输入一个不超过 254 个字符的值。用户可以利用变量标签对每个变量进行记录，例如：

表达式	PT104
注释	记录变量标签 PT104

记录过程变量 PT104 的值。应将变量 PT104 定义为变量标签。您还可以记录任何 Cicode 表达式或函数，例如：

表达式	PT104/COUNTER
注释	记录变量标签 PT104 除以变量标签计数器。

触发器

触发数据记录的 Cicode 表达式（或变量标签）。输入一个不超过 254 个字符的值。例如：

触发器	PT104<500
-----	-----------

在本范例中，当变量标签 (PT104) 的值低于 500 时开始记录。

对于周期性 SPC 趋势，只有当触发值为 TRUE 时数据才被记录。在以上范例中，数据在 PT104 值低于 500 时被持续记录，当值上升至 500（或更高）时记录停止。在 PT104 值降低到低于 500 之前，不会再次开始记录。

并非必须为每个周期性 SPC 趋势都指定触发器。如果您没有为周期性 SPC 趋势指定触发器，记录会连续发生。

对于事件 SPC 趋势，当激活值从 FALSE 变成 TRUE 时，数据被记录一次。在以上范例中，当 PT104 值首次降低至小于 500 时，一个采样被记录。只有当 PT104 值上升至 500（或更高），然后又下降至 500 以下时，才会记录另一个采样。

采样周期（16 个字符）

hh:mm:ss（小时：分钟：秒）格式的数据采样周期。Vijeo Citect 每个周期检查一次**触发**。如果触发为 TRUE 或刚从 FALSE 变为 TRUE（在事件 SPC 趋势中），Vijeo Citect 会记录**表达式**的值。

范例

采样周期	注释
30	每 30 秒记录一次数据
10:00	每 10 分钟记录一次数据
10:00:00	每 10 小时记录一次数据
2:30:00	每 2.5 小时记录一次数据

此属性为可选属性。如果用户没有指定采样周期，默认采样周期是 10 秒。

注意： 如果您在现有工程中编辑此属性，则应先删除相关的趋势文件，然后再运行新的运行系统。

类型（32 个字符）

SPC 趋势类型：

1. TRN_PERIODIC
2. TRN_EVENT

注意： SPC 不支持周期-事件趋势，它是周期和事件趋势属性的一个组合。另外，如果使用基于事件的 SPC 标签，您的显示屏幕刷新率可能比应用程序的要求或实际刷新率要慢。

规格下限（16 个字符）

规格下限 (LSL)。此值用作确定过程性能的规格下限。当与 USL 结合时使用，它可为过程提供公差。

如果您不熟悉过程性能和性能索引，请咨询专家意见。不要将此域留空，应该尝试至少一次估计。为此标签输入一个您认为的最低可接受值。如果将此域留空，只有性能分析会受影响。

规格上限（16 个字符）

规格上限 (USL)。此值用作确定过程性能的规格上限。当与 LSL 结合时使用，它可为过程提供一个公差。

如果您不熟悉过程性能和性能索引，请咨询专家意见。不要将此域留空，您应该尝试至少一次估计 - 为此标签输入一个您认为的最高可接受值。如果将此域留空，只有性能分析会受影响。

注释（48 个字符）

任何有用的注释。

扩展形式的字段

以下字段通过扩展形式实现（按 **F2**）。

文件名（253 个字符）

存储数据所用的文件。必须指定完整的路径或使用路径替换。

当 Vijeo Citect 从用户的工厂级采集数据时，数据被保存在用户计算机硬盘上的某个文件中。在随后用数据显示 SPC 趋势时，它会从这个文件中读取数据。（Vijeo Citect 为每个 SPC 标签使用单独的文件。）

默认情况下，Vijeo Citect 将该文件保存在安装 Vijeo Citect 的硬盘上的 [DATA] 目录中。文件的缺省名称是 SPC 标签名称。但是，用户可以另外指定一个文件名称，如：

文件名	[DATA]:TANK131
-----	----------------

其中 [DATA] 指定了数据所在的磁盘以及路径。路径替换使得用户的工程更加“便捷”。

文件名称属性是可选的。如果未指定文件名，文件名默认位于安装 Vijeo Citect 的硬盘上的 [DATA]:<Name> 下。其中 <Name> 是 SPC 标签名称。

注意： 如果使用“文件名”属性，请确保没有任何其他 SPC 标签或趋势标签使用相同的文件名。访问同一文件的两个标签可能导致包含趋势/SPC 数据丢失或损坏的系统错误。



警告

不希望的设备操作

如果使用“文件名”属性，请确保 SPC 标签使用唯一的名称。访问同一文件的两个标签可能导致包含趋势/SPC 数据丢失或损坏的系统错误。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

注意： 不要在指定文件名时使用文件扩展名。如果您在某个已经存在的工程中对该属性进行编辑（改动文件名称或者路径），所有已经存在的 SPC 数据将会被忽略。

存储方式

选择**基本范围**或**浮点**作为 SPC 数据的存储方法（64 个字符）。这两种选项的关键区别在于基本范围是一种双字节的存储方式，而浮点则要使用八个字节。

同标度变换存储方式相比，浮点存储方式明显地扩大了数据的范围，使得数值具有更高的分辨率。然而，用户需要考虑的是这种方式比前者需要大得多的磁盘空间。基本范围应当被用在需要使用同 V5.31 之前版本相兼容的历史文件的场合。

如果用户没有对存储方式进行指定，默认的设置是**基本范围**。

注意： 如果您在现有工程中编辑此属性，必须先删除关联的趋势文件，然后再运行新的 Runtime 系统。

权限（16 个字符）

操作员在 SPC 页面上显示 SPC 数据所需的权限。

区域（16 个字符）

SPC 数据所属的区域。只有能访问这个区域的用户才可在 SPC 页面上显示 SPC 数据。例如，如果用户输入区域 1，操作员必须能访问区域 1（加上任何所需的权限）来显示 SPC 数据。

工程单位（8 个字符）

记录到日志中的变量/表达式的工程单位。SPC 趋势比例尺和 SPC 趋势指针显示使用工程单位。

格式（11 个字符）

记录到日志中的变量/表达式的格式。SPC 趋势比例尺和 SPC 趋势指针显示使用格式。

此属性为可选属性。如果没有指定格式，默认格式是 ####.#。

文件个数（4 个字符）

在用户的硬盘中所保存的历史文件的个数（对于该标签而言）。

如果用户没有对文件个数进行指定，用户的硬盘上将会保存 2 个历史文件。可指定的最大文件数量为 270。

注意： 如果您在现有工程中编辑此属性，则应先删除相关的 SPC 趋势文件，然后再运行新的运行系统。

子集大小（8 个字符）

每个子集的大小。默认情况下，此值为 5。1 - 25 范围内的值均有效。

时间（32 个字符）

一天中对历史文件的开头进行同步的时间，采用 hh:mm:ss（小时:分钟:秒）的形式。如果您没有指定时间，文件将在 0:00:00 时刻进行同步（即午夜时刻）。

注意： 如果您在现有工程中编辑此属性，则应先删除相关的 SPC 趋势文件，然后再运行新的运行系统。

周期（32 个字符）

历史文件的周期，采用 hh:mm:ss（小时:分钟:秒）的形式。用户还可以采用：

- 输入一周中的某一天，例如星期一、星期二、星期三等，在这一天开始记录历史文件，并以此指定每一周的时间段。
- 输入一个月中的某一天，例如 1 号、2 号、3 号、4 号、5 号等，在这一天开始记录历史文件，并以此指定每一个月的时间段。

- 输入一年中的某月某日，例如 1 月 1 日、2 月 25 日等，在这一天开始记录历史文件。日期和月份之间必须用空格分隔。如果没有指定时间段，默认的时间段将是星期日（每周）。



不希望的设备操作

如果编辑某个 SPC 标签的“期间”属性，则在恢复使用系统之前，应删除所有关联的 SPC 趋势文件。
忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

注意： 如果您在现有工程中编辑此属性，则应先删除相关的 SPC 趋势文件，然后再运行新的运行系统。

过程平均值（16 个字符）

过程平均数的计算覆盖（X 双轴）。如果在这里指定的一个值将会用于所有 SPC 计算，而不是 Vijeo Citect 计算的值。这会影响控制极限的计算，控制极限通常是收集的数据样本的一个函数。

除非用户对 SPC 非常熟悉，不然不要使用这个域。

标准偏差（16 个字符）

过程标准偏差的计算覆盖。如果在这里指定的一个值将会用于所有 SPC 计算，而不是 Vijeo Citect 计算的值。这会影响控制极限的计算，控制极限通常是收集的数据样本的一个函数。

除非用户对 SPC 非常熟悉，不然不要使用这个域。

范围（16 个字符）

过程范围的计算覆盖（R 轴）。如果在这里指定的一个值将会用于所有 SPC 计算，而不是 Vijeo Citect 计算的值。这会影响控制极限的计算，控制极限通常是收集的数据采样的一个函数。

除非用户对 SPC 非常熟悉，不然不要使用这个域。

SPC 控制图表

Vijeo Citect 使用以下类型的控制图表：

- [XRS 控制图表](#)
- [性能图表](#)
- [Pareto 图表](#)

另请参见
[控制图表线常量](#)

XRS 控制图表

XRS 图表显示子集平均值，差距，和标准偏差的趋势。操作 XRS 类似于标准趋势，但带有附加的 SPC 额外特征。每个子集作为一个图表上的单一节点和由一条线连接的连续节点显示。

每个控制图有一条中心线和上部极限 (UCL) 和下部极限控制线 (LCL)。Vijeo Citect 自动在运行时间计算这些 SPC 值。如果用户想要覆盖 UCL 和 LCL，可在 SPC 标签中输入过程平均数、差距和标准偏差。

另请参见
[配置 XRS 图表](#)

配置 XRS 图表

精灵可以简化添加新 SPC 画面的任务。如何创建一个新图表：

- 定义 SPC 标签。
- 使用一个 XRS 模块创建画面。

注意：如果您希望开发自己的 XRS 模块，可通过复制并修改现有模块来实现。

性能图表

过程性能图表是当前显示（在平均值图表上）的所有样本数据的频率柱状图和分布图。Vijeo Citect 会自动获得趋势性数据，构建一个分布，添加规格下限和规格上限。它还会计算 Cp、Cpk、峰度和偏斜度指标。

过程性能的定义涉及给定 SPC 标签的规格上限 (USL) 和规格下限 (LSL)。这些值在 SPC 标签中定义，而且应准确表示用户的要求。

另请参见
[配置性能图表](#)

配置性能图表

精灵可以简化添加新 SPC 画面的作业。要创建新图表，请执行下列操作：

1. 定义 SPC 标签和指定 LSL 及 USL。
2. 使用性能 (Cpk) 模板建立画面。

Pareto图

Pareto 分析是一种用于辨别问题和条件相对重要性的方法。Pareto 图是一个从高到低排列的频率柱状图 – Vijeo Citect 在运行时间自动排列各个直条。柱状图中每个直条的数据代表一个 Vijeo Citect 变量（在变量标签中定义的）。不要使用 SPC 标签。

注意：通常，柱状图的频率为整数类型，但需要时可以使用浮点型。负数值无效。

另请参见

[配置 Pareto 图](#)

配置排列图

精灵能够简化添加新排列图的任务。

如何创建一个新图表：

1. 定义变量标签（排列图不使用 SPC 标签）。
2. 用一个排列图模块建立页面。

如何配置一个排列图：

1. 点击“新建页面”或选择文件 | 新建。
2. 选择**类型：画面**。
3. 选择 SPC 页面的**分辨率**（大小）。
4. 选择 SPC 图表的 **SPCPareto 模块**。
5. 点击**确定**。
6. 双击显示（模块提示）。
7. 在**标签名称**域中输入变量标签。（在此处可以使用变量标签，不必一定要使用 SPC 标签。）
8. 在**标签描述**框中输入变量描述。
9. 点击**确定**。
10. 保存页面。

SPC 报警

Vijeo Citect 自动监视数种特定于 SPC 数据的特殊类型的条件。当一个特殊模式或事件针对 SPC 标签发生时，它会设定适当的报警。通常，这些报警与 XRS 控制图相关联，并与该图一起使用。

SPC 报警的配置方式与标准数字量报警不同，专为此额外功能提供。必须使用高级报警表配置 SPC 报警。可以使用 SPCAlarms Cicode 函数检查报警的条件：

注意：SPC 报警只能在与包含 SPC 标签的趋势服务器位于同一集群中的报警服务器上定义（尽管 SPC 标签中引用的变量标签可能位于不同的 IOserver 集群中）。

按下述方式填写高级报警表：

高级报警	
报警标签	Feed_Above_UCL
报警描述	不受控制的变差
表达式	SPCAlarms("Feed_SPC", XAboveUCL)
注释	数个采样是上面的 UCL

SPC（趋势）服务器检查任何指定报警条件。一旦发现报警条件，它会通知报警服务器有报警发生。了解 SPC 图表中显示的子集数量，以及用于 SPC 报警计算的数量（由 [SPC]AlarmBufferSize 参数设置）是非常重要的。如果这两个值不同，SPC 报警可能与用户的 SPC 图表相关联。

以下列表列举了有效的报警类型：

名称	描述
XFreak	单点和中心线大不相同（ 2σ ）。
XOutsideCL	过程平均数超出任何一个控制极限（UCL 或 LCL）
XAboveUCL	过程平均数超出上部控制极限（UCL）。
XBelowLCL	过程平均数低于下部控制极限（LCL）。
XOutsideWL	过程平均数超出提醒极限，这是 UCL 和 LCL 的 67%。
XGradualUp	过程平均数逐渐上升到一个新水平，这由超过平均数的一些连续点来指出。
XGradualDown	过程平均数逐渐下降到一个新水平，这由低于平均数的一些连续点来指出。
XUpTrend	数个点的值连续上升。
XDownTrend	数个点的值连续下降。
XErratic	大于控制极限的大的浮动。
XStratification	恒久的假象。一些连续点接近（ 1σ 之内）中心线。
XMixture	一些连续点远离（ 1σ 之外）中心线。
ROutsideCL	过程差距超出任何一个控制极限（UCL 和 LCL）。
RAboveUCL	过程差距超出上部控制极限（UCL）。

名称	描述
RBelowLCL	过程差距低于下部控制极限 (LCL)。

注意：所有以上报警依靠 n 个连续点来产生报警。可以通过 SPC 参数为每种报警类型设置 n 值。

另请参见
[SPC 公式和常量](#)

SPC 公式和常量

SPC 计算是基于子集中收集的采样。每个子集包含相同数量的采样（一般为 4 个）。子集的大小是在 **SPC 标签** 属性表中设置的。

每个子集的采样数量可从 1 到 25，包括 1 和 25。

当每个子集中的采样数为 1 时

子集平均值 (\bar{X}_i):

是集成的单一采样值，由以下公式定义：

$$\bar{X}_i = X_i$$

其中 X_i 是子集中的单一采样值。

移动范围 (MR):

是各连续采样值之间的区别，定义方式如下：

$$MR_i = X_i - X_{i-1} \quad i \geq 2$$

其中 X_i 是当前采样值， X_{i-1} 是上一个采样值。过程中的移动范围数量总比子集数量小一。

子集标准偏差 (s):

对绝对变差或散布的测量。它描述了采样值和它们平均值的差距，由以下公式估测：

$$s_i = \frac{MR_i}{D2} \quad i \geq 2$$

过程中子集的标准变差的数量总比子集数量小一。

过程平均数 ($\bar{\bar{X}}$):

$$\bar{\bar{X}} = \frac{\bar{X}_1 + \bar{X}_2 + \dots + \bar{X}_m}{m}$$

其中 \bar{X}_1 、 \bar{X}_2 和 \bar{X}_m 是子集平均值， m 是过程中子集的总数。

过程范围 (\bar{R}):

$$\bar{R} = \frac{MR_2 + MR_3 + \dots + MR_m}{m-1}$$

其中 MR_2 、 MR_3 和 MR_m 是子集的移动范围， m 是过程中子集的总数。

过程标准偏差 (\bar{s}):

$$\bar{s} = \frac{\bar{R}}{D_2}$$

当每个子集中的采样数量大于 1 时

子集平均值 (\bar{X}):

是集中采样的平均数（不是中值或中心），由以下公式定义：

$$\bar{X} = \frac{X_1 + X_2 + \dots + X_n}{n}$$

其中 X_1 、 X_2 和 X_n 是子集中的采样值， n 是子集中采样的总数。

子集范围 (R):

是集中最高和最低采样间的差距，由以下公式定义：

$$R = X_{\max} - X_{\min}$$

其中 X_{\max} 是集中最大的采样值， X_{\min} 是集中最小的采样值。

子集标准偏差 (s):

对绝对变差或散布的测量。它描述了采样值和它们平均值的差距，由以下公式定义：

$$s = \sqrt{\frac{\sum (X_i - \bar{X})^2}{n-1}}$$

其中 X 是集成的采样值， \bar{X} 是集成的平均数， n 是集成的采样数量。

过程平均数 ($\bar{\bar{X}}$):

$$\bar{\bar{X}} = \frac{\bar{X}_1 + \bar{X}_2 + \dots + \bar{X}_m}{m}$$

其中 \bar{X}_1 、 \bar{X}_2 和 \bar{X}_m 是子集平均值， m 是过程中子集的总数。

过程范围 (\bar{R}):

$$\bar{R} = \frac{R_1 + R_2 + \dots + R_m}{m}$$

其中 R_1 、 R_2 和 R_m 是子集的范围， m 是过程中子集的总数。

过程标准偏差 (\bar{s}):

$$\bar{s} = \frac{s_1 + s_2 + \dots + s_m}{m}$$

其中 s_1 、 s_2 和 s_m 是子集的标准偏差， m 是过程中子集的总数。

平均控制极限 (UCL_x 和 LCL_x):

指定近似的 3 个 Σ 界线。对于一个正态分布，99.74% 的样本将在这个边界之内。

$$UCL_x = \bar{\bar{X}} + A_2 * \bar{R}$$

$$LCL_x = \bar{\bar{X}} - A_2 * \bar{R}$$

其中 \bar{R} 是过程范围， A_2 是一个常量（在控制图表线常量表中提供）。

范围控制极限 (UCL_R 和 LCL_R):

指定近似的 3 个 Σ 界线。对于一个正态分布，99.74% 的样本将在这个边界之内。

$$UCL_R = D_4 * \bar{R}$$

$$LCL_R = D_3 * \bar{R}$$

其中 \bar{R} 是过程范围， D_3 和 D_4 是常量（在控制图表线常量表中提供）。

标准偏差控制极限 (UCL_s 和 LCL_s):

指定近似的 3 个 Σ 界线。对于一个正态分布，99.74% 的样本将在这个边界之内。

$$UCL_s = B_4 * \bar{s}$$

$$LCL_s = B_3 * \bar{s}$$

其中 \bar{s} 是过程标准偏差， B_3 和 B_4 是常量（在控制图表线常量表中提供）。

过程性能 (Cp):

是一个过程符合一个特殊公差的能力。如果位于规格上限和规格下限之间的一个过程的变量的样本百分比大于指定值，该过程被认为是合格过程。

继承过程能力定义如下：

$$C_p = \frac{(USL - LSL)}{6s}$$

- $C_p > 1.0$ - 表示过程变异将在指定的限制（USL 和 LSL）内，因此符合能力要求。
- $C_p < 1.0$ - 表示过程不符合能力要求。

过程性能基于最坏情况数据被定义为：

$$C_{pk} = \frac{\min((USL - \bar{X}), (\bar{X} - LSL))}{3s}$$

- $C_{pk} < 0$ - 表示过程平均值在指定限制（USL 和 LSL）外。
- $C_{pk} = 0$ - 表示过程平均值等于其中一个指定限制。
- $C_{pk} > 0$ - 表示过程平均值位于指定限制内。
- $C_{pk} = 1.0$ - 表示 6σ 限的一侧位于一个规格限制上。
- $C_{pk} > 1.0$ - 表示 6σ 限完全位于指定限制内。

偏斜 (Sk):

是一个频率分布的不对称程度（通常和一个普通分布相关联）。

$$sk = \frac{\sum (X_i - \bar{X})^3}{Ns^3}$$

其中 N 是整个过程的采样数（即子集大小*子集数量）。

- 偏斜 > 0 - 表示柱状图的平均值（尾部）偏右。
- 偏斜 < 0 - 表示柱状图的平均值（尾部）偏左。

峰度 (Ku):

是一个频率分布的巅峰度（通常是和一个普通分布相关联）。

$$Ku = \frac{\sum (X_i - \bar{X})^4}{Ns^4}$$

其中 N 是整个过程的采样数（即子集大小*子集数量）。

- Kurtosis < 3 - 表示窄的分布和相对高的峰度。
- Kurtosis > 3 - 表示宽的分布和相对平的峰度。

控制图线条常量

控制图线条常量见下表：

样本 组	平均数	差距			标准偏差	
	A2	D2*	D3	D4	B3	B4
1	2.660	1.128	0	3.267	0	3.267
2	1.880		0	3.267	0	3.267
3	1.023		0	2.574	0	2.568
4	0.729		0	2.282	0	2.266
5	0.577		0	2.114	0	2.089
6	0.483		0	2.004	0.030	1.970
7	0.419		0.076	1.924	0.118	1.882
8	0.373		0.136	1.864	0.185	1.815
9	0.337		0.184	1.816	0.239	1.761
10	0.308		0.223	1.777	0.284	1.716
10	0.308		0.223	1.777	0.284	1.716
11	0.285		0.256	1.744	0.321	1.679
12	0.266		0.283	1.717	0.354	1.646
13	0.249		0.307	1.693	0.382	1.618
14	0.235		0.328	1.672	0.406	1.594
15	0.223		0.347	1.653	0.428	1.572
16	0.212		0.363	1.637	0.448	1.552
17	0.203		0.378	1.622	0.466	1.534
18	0.194		0.391	1.608	0.482	1.518
19	0.187		0.403	1.597	0.497	1.503
20	0.180		0.415	1.585	0.510	1.490
21	0.173		0.425	1.575	0.523	1.477
22	0.167		0.434	1.566	0.534	1.466
23	0.162		0.443	1.557	0.545	1.455
24	0.157		0.451	1.548	0.555	1.445

样本	平均数	差距	标准偏差		
25	0.153	0.459	1.541	0.565	1.435

*当一个子集有一个样本时，D2 只用于估计标准偏差。

请参考 ANSI Z1.1-1985, Z1.2-1985 & Z1.3-1985: 美国国家标准, *质量控制图表指南, 分析数据的控制图表的方法, 生产过程质量控制的控制图表方法。*

提示

双击 SPC 页面的图表区域来显示 SPC 精灵和修改 SPC 变量。

在此过程中，工具和菜单条目将自动打开 Vijeo Citect 窗体。移动光标直到它变成手行，然后查找这些工具或选项。

Chapter: 28 报表信息

用户可要求有关现场状况的规律性报表，报表提供有关现场特殊条件的信息。报表可在指定期间或在某特定事件发生（如位地址状态的更改）期间在请求的基础上运行。报告的输出由设备控制。当报告运行或保存到磁盘以便于稍后打印时，将打印报告。用户可使用一个文本编辑器或文字处理器来查看、编辑或打印报表，或在 Vijeo Citect 中作为画面一部分来显示。

报表也可以包括报表运行时执行的 Cicode 语句。

配置报表有两个阶段：

- 报表属性
- 报表格式文件

如果报表数据和一个在启动时没有正确初始化的 I/O 设备相关联，或在 Vijeo Citect 运行时不在线，相关数据不被写入报表（因为是无效参数值）。而是将错误代码写入报表。

另请参见

[配置报表](#)

[运行报表](#)

[报表格式文件](#)

[处理报表中的通信错误](#)

配置报表

如何设计、配置和使用报表：

1. 为报表输出配置一个设备（例如，如果在报表运行时需将报表保存至一个文件，需设置一个 ASCII_DEV 设备）。
2. 配置报表属性。
3. 编辑报表格式文件注意，对于一个 RTF 报表，报表格式文件必须保存成 RTF 格式（即，带有 .RTF 文件扩展名）。
4. 使用**计算机设置向导**将您的 PC 定义为报表服务器。

如何配置报表属性：

1. 选择**系统 | 报表**以显示“报表”对话框。
2. 输入报表属性。点击**编辑**以编辑报表格式文件。

使用“报表”对话框配置报表。

另请参见

[报表对话框](#)

报表对话框

报表表格有以下属性：

名称

报表的名称。名称最多为 79 个字符。名字可由分号 (;) 或单引号 (') 之外的任何字符组成。报警名称对于集群必须是唯一的。

注意： 尽管集群名称可以保留为空，但该名称对于所有已定义的集群必须是唯一的。

集群名称

运行此报表的集群的名称（最多 16 个字符）。如果未设置集群名称，则 Vijeo Citect 将在所有已定义的集群上运行此报表。

时间

同步报表的时间是 hh:mm:ss（小时：分钟：秒）格式。如果用户没有指定一个时间，报表同步时间是 0:00:00（即，午夜）。输入一个不超过 32 个字符的参数值。

周期

报表时间段是 hh:mm:ss（小时：分钟：秒）格式。输入一个不超过 32 个字符的参数值。或者可以：

- 输入周的某一天报表开始运作来指定一个周阶段，例如，星期一，星期二，星期三等。
- 输入月的某一天报表开始运作来指定一个月阶段，例如，一日，二日，三日，四日，五日等。
- 输入日和月报表开始运作来指定一个年阶段。例如，一月一日，二月二十五等。日和月间需有一个空格。

如果用户没有指定一个时间段，报表将每天运行。

触发器

任何 Cicode 表达式（或“变量”标签）可激活报表。输入一个不超过 254 个字符的值。如果表达式（在此字段中）结果是 TRUE，以及时间和时间段字段是空白的，报表将运行。只有当表达式成为 TRUE 时，报表才运行。在报表重新运行前，表达式必须从 FALSE 变为 TRUE。

报表格式文件

报表格式文件名。输入一个不超过 253 个字符的参数值。如果用户没有指定一个文件扩展名，默认为 .RPT。可使用任何有效文件名；但是，您不能在此字段中使用路径替换。如果用户指定了一个没有路径的文件名，则文件将被保存在预定义为“Run”的目录中。除非使用 RTF 扩展名，否则报表将（由 Vijeo Citect 编辑器）被假设为 ASCII 文件。

注意： 报表格式文件名最多为 64 个字符，或 253 个字符（包括路径）。它由除单引号（'）和分号（；）外的任何字符组成。

输出设备

用于发送报表的设备。输入一个不超过 16 个字符的值。

对于需保存为一个文件的 RTF 报表，选择一个 ASCII_DEV 类型的设备。由于内容特性的不同，不建议使用相同的 ASCII 设备来记录 RTF 和非 RTF 报表。

注意： 如果两个或多个报表同时运行，并将输出内容发送到同一打印机，则每个报表输出会混合在一起。用户需在每个报表中用信号量控制访问打印机。请参见 [SemOpen Cicode 函数](#)。如果报表只包含 Cicode 语句（而且没有输出数据），这个属性是可选择的。

注释

任何有用的注释。输入一个不超过 48 个字符的参数值。

扩展形式的字段

以下字段通过扩展形式实现（按 F2）。

权限

如果报表是一个命令驱动报表，则操作员需要一个权限来运行这个报表。输入一个不超过 16 个字符的参数值。

如果报表是由时间或事件驱动的，这个属性是可忽略的。

注意： 如果用户给一个报表分派一个确认权限，用户不可给运行该报表的命令分派一个权限。如果给命令分派一个不同的权限，则操作员需拥有两者的权限来运行报表。

区域

这个报表所属的区域。输入一个不超过 16 个字符的参数值。只有拥有此区域访问权限（和任何被要求的权限）的用户才能运行这个报表。例如，如果输入区域 1，则操作员必须具有访问区域 1 的权限（以及任何被要求的权限）才能运行这个报表。

运行报表

可以通过以下方法运行报表：

- Vijeo Citect 启动时自动运行
- 在指定时间和周期中自动运行
- 触发事件时自动运行
- 通过使用命令
- 结合上述方法

另请参见

[启动时运行报表](#)

[指定时间和周期](#)

[使用触发器](#)

[使用命令](#)

启动时运行报表

可以在启动时运行报表。Vijeo Citect 会在启动时搜索名为“Startup”的报表，如果 Vijeo Citect 找到此报表，则会自动运行。可以通过 *计算机设置向导* 更改缺省报表的名称。

另请参见

[指定时间和周期](#)

指定时间和周期

周期确定报表何时运行。用户可以采用 hh:mm:ss（小时：分钟：秒）形式指定周期，例如：

周期	注释
1:00:00	每小时运行报表
6:00:00	每六小时运行报表
72:00:00	每三天运行报表
星期一	每星期一运行报表
15 ^号	每月十五日运行报表
6月25日	6月25日运行报表

还可指定同步报表的时间，例如：

时间	注释
6:00:00	在早晨 6:00 点同步报表
12:00:00	在中午 12:00 点同步报表

时间同步每天运行报表的时间，周期确定报表运行的时间，例如：

时间	周期
6:00:00	1:00:00

在这个例子中，报表每隔一小时运行一次。如果早晨 7: 25 启动 Runtime 系统，报表将在早晨 8: 00 运行，然后每隔一小时运行一次。

另请参见
[使用触发器](#)

使用触发器

您可以使用任何 Cicode 表达式（或变量标签）作为报表的触发器。如果表达式结果（在**触发器**字段中）变为 TRUE，并且**时间**和**周期**字段为空，报表将运行。例如：

时间	
周期	
触发器	RCC1_SPEED<10 AND RCC1_MC

只有当表达式（触发器）变为 TRUE 时，此报表才会运行，也就是说，当数字标签 RCC1_MC 是 ON 而模拟标签 RCC1_SPEED 小于 10 时。表达式必须首先变为 FALSE 然后再变为 TRUE，此时报表才能再次运行。

如果使用**时间**和/或**周期**字段，触发器将在指定的时间和/或周期被选中，例如：

时间	6:00:00
周期	1:00:00
触发器	RCC1_SPEED<10 AND RCC1_MC

此报表每小时运行一次，但前提必须是表达式（触发器）是 TRUE（也就是说，数字标签 RCC1_MC 是 ON 而模拟标签 RCC1_SPEED 小于 10）。

另请参见
[使用命令](#)

使用命令

如果**时间**、**周期**和**触发器**字段为空白，则报表只能通过调用 Cicode Report() 函数的命令运行。

报表格式文件

报表格式文件指定报表中的数据是如何被格式化的。用户可在任何报表中使用固定文本、Cicode 表达式和数据库变量。

用户可以使用一个由 Windows 支持的文本编辑器创建（和修改）报表格式文件。如果报表格式文件是 RTF（富文本格式），您应使用 Microsoft Wordpad。

包括固定文本

用户可在报表中包括固定文本，指定文本可如用户制定的完全一样（例如，报表名，描述等）。

包括 OLE（只限于 RTF 文件）

在 RTF 报表格式文件中，可链接或嵌入对象，然而，Vijeo Citect 不能显示或打印这些对象。

使用字体（只限于 ASCII 格式）

如果用户的格式文件是 ASCII 格式的，可在报表中使用 Windows 支持的任何文本字体。可使用 PrintFont() 函数指定一个字体。因为 RTF 格式文件使用主机文字处理器的格式特点，所以不需要这个函数。

包括 Cicode 表达式和变量

用户可使用括号 {} 来包括 Cicode 表达式和变量（和可选择的格式规格）-例如：

```
{TIME(1) }
{PV12:#####.##}
{PV12:4.2}
```

每个域的大小（字符数）是由格式规格或括号间的字符数决定的。在上面的例子中，变量 PV12 的格式是用小数点之前四个字符和之后的两个字符来完成。

用户不需要包括格式，例如：

```
{PV12}
```

在这个例子中，变量只是由四个字符（或者说，括号间的字符数）组成的。

将一个报表记录到数据库设备时遵循以下规则：

- 此格式（针对报表字段）指定的字段大小不可大于设备中指定的相关字段。
- 每个字段规格间不允许空格，如：

```
{TIME(1) }{PV12:#####.##}{PV12:4.2}
```

包括 Cicode 段

用户可使用以下格式包括 Cicode 段：

```
{CICODE}
Statements;
{END}
```

Cicode 段是以命令 {CICODE} 和 {END} 来划定界线。{END} 命令后，报表将转换回 WYSIWYG 模式。如果整个报表是 Cicode，或最后部分是 Cicode，则不需要 {END} 命令。

除非用户使用 Print() 或 PrintLn() 函数，否则 Cicode 段不会向设备发送任何输出内容。如果使用以上任何一个函数，参数将被打印到设备。

Cicode 变量

用户可在 Cicode 段中声明变量。所有变量都应在文件的开始（即，在任何报表格式或 Cicode 前）声明。先添加一个 {CICODE} 段，例如：

```
{CICODE}
INT nVar1;
STRING sVar2;
Statements;
{END}

Remainder of report
```

包括注释

通过使用括号包括注释字符“!”来包含注释，例如：

```
{!This is a Comment}
```

报表正文中的注释不同于 Cicode 段的注释：Cicode 段的注释不需括号。

包括其他报表元素

下表说明在报表中包括的其他元素。

目的	方法
发出换页	使用换页指示符：{FF}
包括绘图	使用绘图函数。
包括趋势数据	使用 TrnGetTable() 函数。
包括趋势图形	使用 TrnPlot() 函数。

另请参见
[报表示例](#)

报表示例

下面是一个报表格式文件示例（针对一个打印机或 ASCII 文件设备）：

```

-----
SHIFT REPORT
-----
{Time(1) } {Date(2) }
Shift Production {Shift_Prod:###.##} tons
Total Production {Total_Prod} tons
{! The following Cicode displays "Shift Report Complete" on the
screen}
{CICODE}
Print("End of Report")
Shift_Prod = 0; ! Reset the Shift production tonnage

Prompt("Shift Report Complete");
{END}
{FF}

```

这个报表可向设备生成以下输出内容，在图形画面显示“转换报表完毕”。

```

-----
SHIFT REPORT
-----
6:00am    12/3/92
Shift Production 352.45 tons
Total Production 15728 tons
End of Report

```

如何编辑一个报表格式文件：

1. 从报表属性表格选择相关报表，点击**编辑**。或者，点击**编辑报表**。
2. 选择要编辑的报表
3. 点击**编辑**。

注意： 如果报表格式文件已存在，会被加载到编辑器供用户编辑。如果文件不存在，Vijeo Citect 将创建一个新的文件。

如何改变报表格式文件编辑器：

1. 点击**选项**工具，或选择**文件 | 选项**。
2. 输入新的**编辑器**名称。

3. 点击**确定**。

处理报表中的通信错误

您可以通过以下两种方法处理与 I/O 设备间的通信错误（例如，I/O 设备启动时没有正确初始化或在 Vijeo Citect 运行时离线）：

- 可以将通信错误和无效数据作为错误代码写入报表。
- 如果与 I/O 设备的通信丢失，您也可以禁止通过 I/O 设备触发的报表的运行。

另请参见：

[报告 I/O 设备数据错误](#)

[压缩报表](#)

报告 I/O 设备数据错误

如果检测到一个通信错误（针对 I/O 设备），或者如果数据是无效的，以下其中一个提醒信息将被写入报表（而不是值）：

错误	意义
#ASS	参数值关联（替换字符串或精灵）错误。
#COM	与 I/O 设备的通信丢失
#DIV/0	尝试除以 0（零）
#ERR	出现了某种非常见的错误。（使用函数 <code>IsError</code> 来查找该错误。）
#MEM	超出内存或要求超过 64 kb 的字节。
#PEND	来自设备的数据正在挂起初始更新，以显示参数值。
#RANGE	返回参数值超出范围
#STACK	参数值引起一个堆栈溢出
#WAIT	来自调度设备的数据不可用，因为它没有达到其调度间隔，且没有缓存参数值。

例如：

报表格式：

```
{PV_1} {SP_1} {OP_1}
```

如果当 PV_1 值（例如：101.5）超出范围，SP_1 是 42.35，OP_1 是 60.0 时运行以上报表，报表输出是：

报表输出：

```
#RANGE 42.35 60.0
```

在某些情况下，当报表写入一个数据库设备，您可能希望停止提醒讯息，并将参数值写入报表（即使参数值是无效的）。使用 ERR_FORMAT_OFF 命令停止所有提醒讯息，并将所有数据作为参数值写入报表。

例如：

报表格式：

```
{ERR_FORMAT_OFF}
{PV_1} {SP_1} {OP_1}
If the above report is run when the value of PV_1 is out of range (for example
101.5), SP_1 is
  42.35 and OP_1 is 60.0, the output of the report is:
```

报表输出：

```
42.35 60.0
```

重新启用提醒讯息，使用 ERR_FORMAT_ON 指定符。

注意： 如果一个 I/O 设备不在线，而且您已停止了通信错误，打印到报表的参数值为 0（零）或当报表最后一次运行时从 I/O 设备读取的参数值。以上任何情况下的参数值都是无效的。

另请参见
[压缩报表](#)

压缩报表

如果检测到通讯错误，可以使用 [Report]ComBreak 参数压缩触发自 I/O 设备的报表。例如，可以在位开启时将报表配置为每小时运行一次。与该位关联的 I/O 设备将离线。如果 [Report]ComBreak 参数是 0，此报表不会运行。如果参数为 1，而且从位读取的最近有效值为 1，报表将运行。

这个参数只用于激活报表，不用于报表中的数据。

Chapter: 29 使用标签

可以通过标记在系统中使用一系列命令（或表达式），而无需在每次使用时重复这些命令（或表达式）。对工程进行编译时，在标记中定义的所有命令（或表达式）都由出现的标记符替代。

标记与编程语言（如 Basic 或“C”）中使用的宏类似。

在不同命令中经常使用相同的语句组合。例如，当操作员确认报警时，可能希望将详细信息记录到文件中。这样的命令会需要多个语句：

```
命令  FileWrite(AlrmFile, Time()); FileWrite(AlrmFile, Date()); . . .
```

可以定义标签，然后使用标签代替语句，而不是在命令中需要时输入相同的语句。编译工程时，出现的每个标签都被解析，也就是说，标签名称将被标签中的表达式替代。例如：



注意辨别标签的下划线的使用。

定义标签后，可在命令中将其用作语句，例如：

系统键盘	
键序列	F5 Enter
命令	<u>_Log_Alarms</u> ;
权限	1

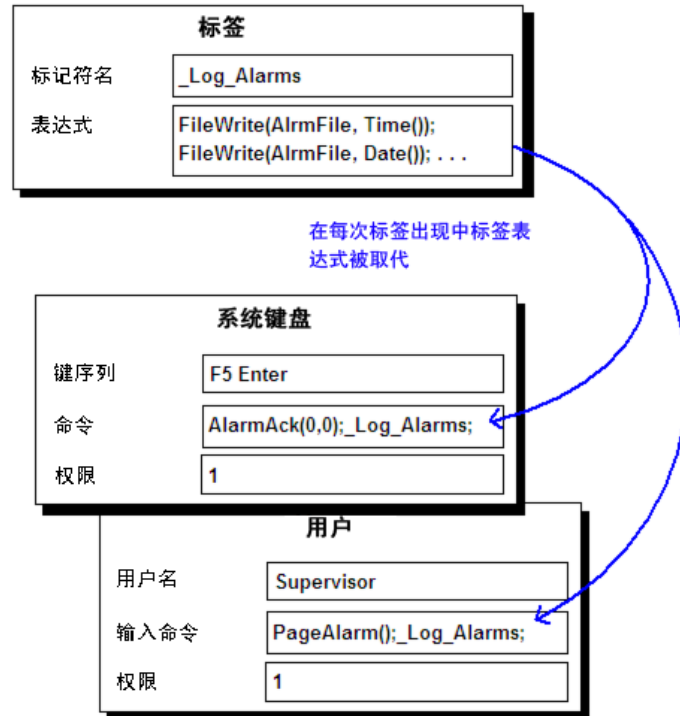
操作员发出此命令时，标签中定义的表达式将在命令中被替代。

标签	
标记符名	<code>_Log_Alarms</code>
表达式	<code>FileWrite(AlrmFile, Time()); FileWrite(AlrmFile, Date()); ...</code>

当操作员发出命令时，表达式中的语句执行

系统键盘	
键序列	<code>F5 Enter</code>
命令	<code>_Log_Alarms;</code>
权限	<code>1</code>

也可以将标签与其它语句组合使用，例如：



标签的主要优点是可以全局定义，并可以在整个 Vijeo Citect 系统中识别。如果需要更改某些内容（在上例中，可能是更改文件名或数据记录方式），则只需在一处进行更改，即在标签定义中。出现的所有其它标签名称将影响所做的更改。

另请参见

[在标签中使用参数](#)

[将参数值转换为字符串](#)

[替代字符串](#)

[定义标签](#)

在标签中使用参数

可以定义接受以括号 () 括起的参数的标签。下例显示按特定参数值使变量递增的标签：

标签名称	Inc(X, STEP)
表达式	X = X + STEP

此处，“X”是要递增的变量，“STEP”用于确定增量的多少。如下例所示，随后可以在命令中使用此标签：

键序列	FastInc
命令	Inc(SP12, 10);

操作员可以使用此命令使 SP12 的参数值以 10 递增。

指定默认值

可以在定义标记符时为参数指定默认值，如下例所示：

标记符名	Inc(X, STEP = 10)
表达式	X = X + STEP

当您在以后再不带参数使用该标记符时，系统将自动使用默认值：

键序列	FastInc
命令	Inc(SP12);

另请参见

[将数值转换为字符串](#)

将数值转换为字符串

在某些情况下，您必须将某个数值在使用前转换为一个字符串。在下面的例子中，我们将使用 DspStr() 函数将某个数值的标记在使用前转换为字符串。

标记符名	ShowVariable(TAG)
表达式	DspStr(25, "BigFont", #TAG + "=" + TAG:##.#);

在上面的例子中，只有一个参数 (TAG) 被传递到一个函数中。该函数实际上需要三个参数 (AN, 字体 (font) 和消息 (message))。当用户在某个命令中使用此标记符时，该函数使用 AN 25，并且消息将采用 "BigFont" 字体来显示。只有第三个参数 (实际消息) 是变化的。

传递到函数的第三个参数是：

```
...#TAG+"="+TAG:##.##
```

#TAG 表明显示的是标志名（而并非其数值本身）。

TAG:##.## 表明标志 TAG 的数值将被转换为字符串并显示出来。该数值将采用小数点前两位，小数点后一位的格式进行显示。

您可以象下面这样在一个命令中采用上述标记符：

命令	ShowVariable(SP12);
----	---------------------

当在你的运行系统中使用该命令时，系统将显示“SP12=<数值>”，其中**数值**是运行该命令时 SP12 当时的值（例如：**SP12=42.0**）。

另请参见
[替换字符串](#)

替换字符串

当数个变量的变量名有相同的部分时，您可以将字符串替换作为参数在标记符中进行传递，如下例所示：

标记符名称	SPDev(TAG)
表达式	Prompt("Deviation=" + "IntToStr(CP##TAG## - SP##TAG##));

在上面的范例中，TAG 是变量名称的通用部分，每次出现在表达式中都会被替换。要显示两个变量（CP123 和 SP123）之间的区别，可以在命令中指定 SPDev(123)，例如：

命令	SPDev(123);
----	-------------

不能在字符串中使用替换。在下面的范例中，DESC 参数（文本描述）不会被替换，因为它位于引号之间：


```
Prompt("Deviation for ##DESC##=" + "IntToStr(CP##TAG## -  
SP##TAG##)")
```

另请参见
[定义标记符](#)

定义标记符

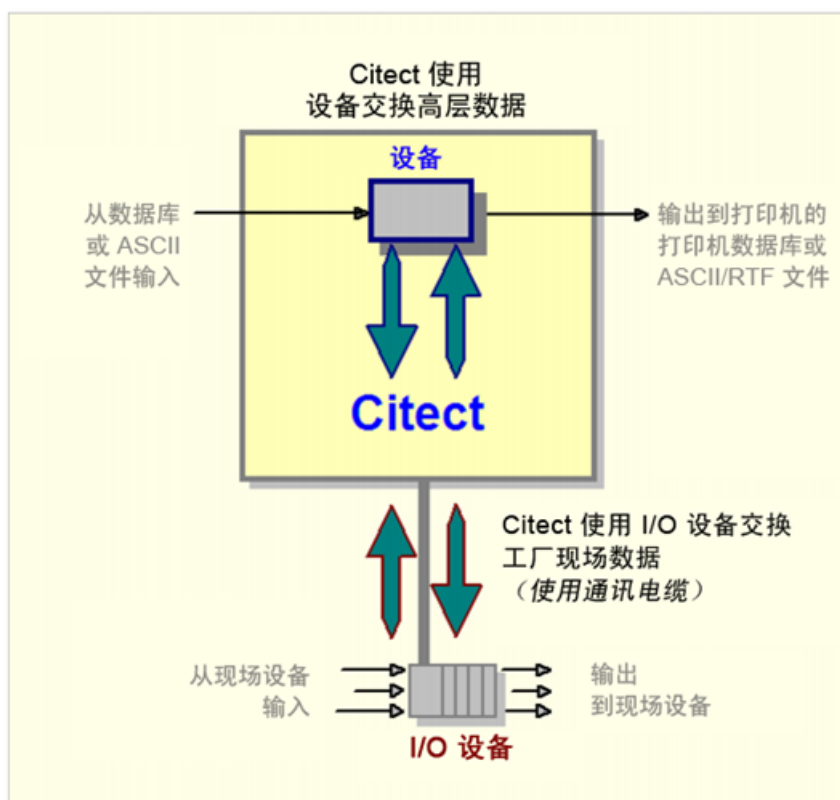
可以定义要在系统中使用的标签。

要定义标记符，请执行以下操作：

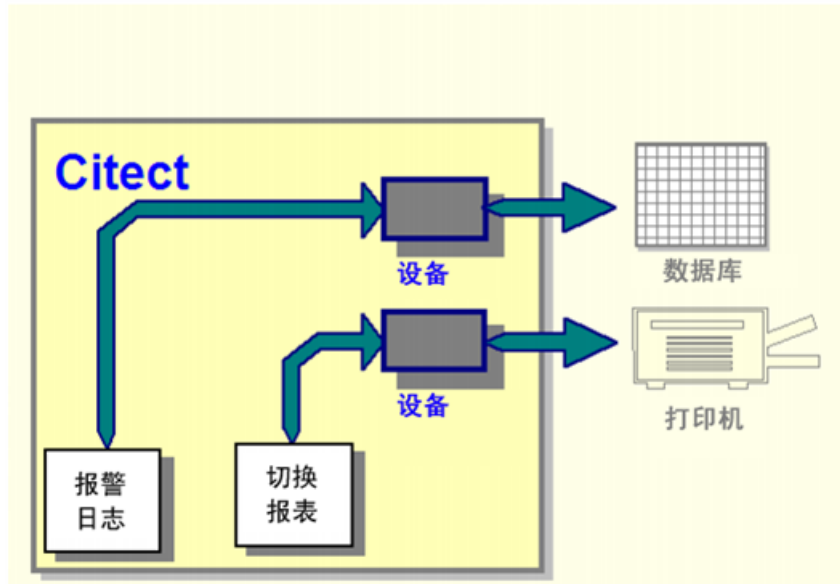
1. 选择**系统 | 标记符**。将显示标记符对话框。
2. 输入不超过 128 个字符的**标记符名称**。如果此名称已被使用（即在 Cicode 或域中），Vijeo Citect 会自动替换下面的表达式。
3. 输入一个用于替换标签的**表达式**（最大长度为 254 个字符）。可以用标签替换实体或 Cicode 表达式的名称；例如，当在几个数据库记录中使用实体（或 Cicode 表达式）时。
4. 添加**注释**（不超过 48 个字符）。
5. 点击**添加**加入新记录，或点击**替换**修改现有记录。

Chapter: 30 使用设备

设备可在 Vijeo Citect 系统中传输 Vijeo Citect 与其它元素（如打印机、数据库、RTF 文件或 ASCII 文件）之间的高级数据（如报表、命令日志或报警日志）。设备与 I/O 设备类似，两者都允许 Vijeo Citect 在控制和监视系统中与其它组件交换数据。



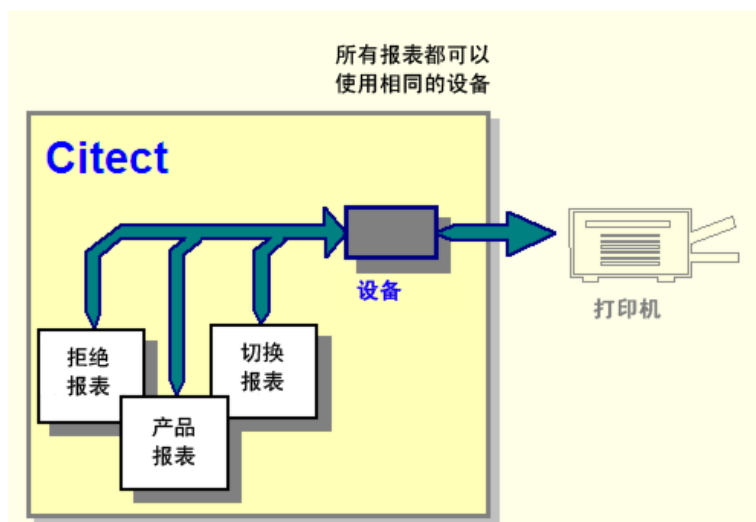
设备可用于多种用途；例如，将报表输出发送到打印机，或者将数据写入数据库。



使用设备可将数据写入：

- RTF 文件
- ASCII 文件
- dBASE 数据库
- SQL 数据库（通过 ODBC 兼容的驱动程序）
- 打印机（连接到 Vijeo Citect 计算机或网络）

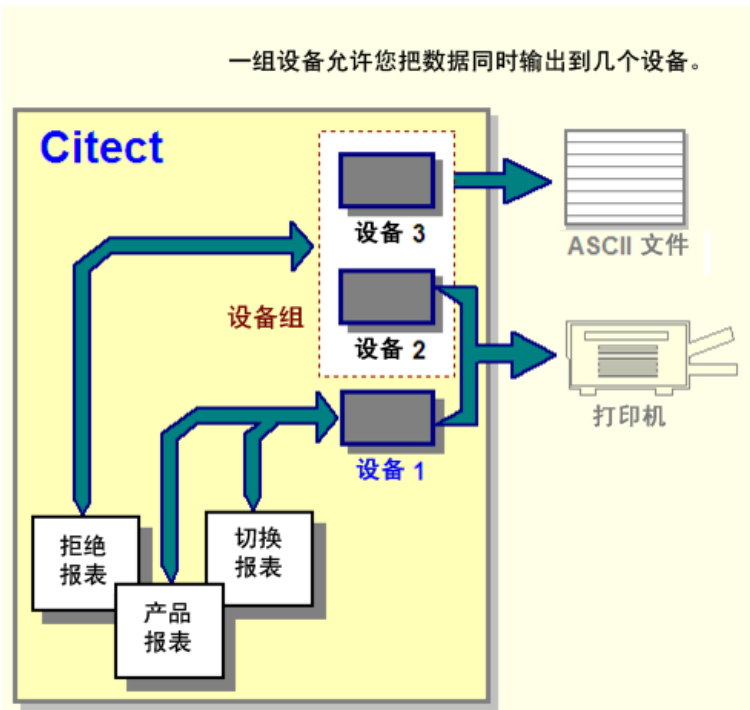
可以配置任意数量的设备，但设备是常用资源。例如，可以配置单个设备，将所有 Vijeo Citect 报表的输出发送到打印机（系统对其发出请求时）。



另请参见
[使用设备组](#)
[配置设备](#)
[格式化设备中的数据](#)
[使用设备历史文件](#)
[关于打印管理](#)

使用设备组

可以使用设备组增加系统的灵活性。通过设备组，可以将相同的数据导出到两个或多个位置。

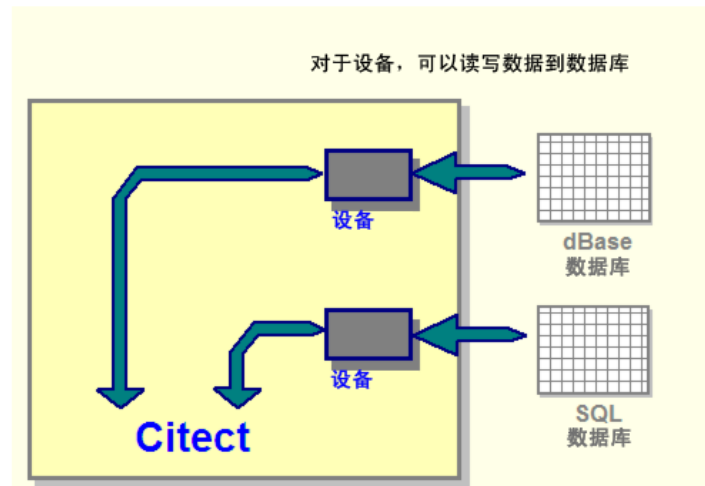


另请参见
[使用设备读取数据](#)

使用设备读取数据

使用设备和 Cicode 函数还可以从以下位置读取数据：

- ASCII 文件
- dBASE 数据库
- SQL 数据库



注意：从设备组读取数据时，数据只能从组中的第一个设备读取。

另请参见
[配置设备](#)

配置设备

必须首先配置设备，然后才能在 Vijeo Citect 系统中使用这些设备。

如何配置设备：

1. 选择**系统 | 设备**。将出现“设备”对话框。
2. 使用下面的文本框的说明完成“设备”对话框。
3. 点击**增加**加入新的记录，或点击**替换**修改现有的记录。

设备具有下列属性：

名称

设备的名称。设备名称可以是一组设备的名称，或是一个设备的标签。输入不超过 16 个字符的值。

另请参见
[使用设备组合](#)
[使用标签](#)

[Vijeo Citect 技术参考中的“预定义设备”](#)

格式

指定设备中的数据格式。此格式由设备类型和发送到设备的数据来决定。输入不超过 120 个字符的值。有关可用字段的信息，请参阅 [使用命令字段](#)。

如果您记录报警或命令消息，则必须指定一个格式，否则不会向设备中写入数据。

注意： 命令的记录设备是在定义命令时指定的。报警的记录设备是在“报警分类”表格中指定的。

生成报表时，格式将被忽略。（为报表定义的格式将用于将报表写入设备。）

另请参见

[格式化设备中的数据](#)

[报警显示字段](#)

[报警汇总字段](#)

[使用命令字段](#)

页眉

设备的附加信息。输入不超过 120 个字符的值。

打印机设备

页眉印制在每一页上。每次达到表格长度时都创建新页。[Device]FormLength 参数用于设置表格长度。

ASCII 文件设备

不适用。

dBASE 数据库设备

包含用于数据库索引的域名，例如：

页眉	{名称}
----	------

注意： 索引键域不能超过 100 个字符。

SQL 数据库设备

特殊数据库类型的连接字符串。

注意： Vijeo Citect 数据库设备只支持 STRING 数据类型。如果您使用其他数据库编辑器修改数据库，则会希望确认所有域均是字符串格式。

文件名称

设备的文件名。输入一个不超过 253 个字符的值。

打印机设备

打印机端口或 UNC 名称，例如：

文件名	LPT1:
文件名称	COM2:
文件名称	\\PrintServer\BubbleJet1

当您指定一个打印机端口时，必须包括冒号字符 (:)，否则 Vijeo Citect 会使用类似于打印机端口（即，LPT1 或 COM2）的名称写入文件（设备）。

注意：在 Windows 95 中使用 UNC 名称时，控制面板的打印机部分中必须包括此打印机。

ASCII 文件设备和 dBASE 数据库设备

已激活文件的名称，例如：

文件名称	ALARMLOG.TXT
文件名称	[DATA]:ALARMLOG.TXT

这个属性是可选的。如果没有指定文件名，**文件名称**将缺省设置为安装有 Vijeo Citect 的硬盘上的 \Vijeo Citect 7.10\bin\

SQL 数据库设备

数据库表格，例如：

文件名称	LOGFILE
文件名称	REPTBL

类型

设备的类型。输入不超过 16 个字符的值。

设备类型	设备描述
ASCII_DEV	ASCII 文件*
PRINTER_DEV	打印机
DBASE_DEV	dBASE 文件
SQL_DEV	SQL 数据库

* 在定义 RTF 报表属性时，如果报表被保存为一个文件，则需选择一个 ASCII 设备。

这个属性是可选的。如果您没有指定类型，设备**类型**将是 ASCII_DEV，除非：

文件名是一个打印机设备（LPT1:到 LPT4: 或 COM1:到 COM4:或 UNC 名称），其中**类型**是 PRINTER_DEV，或

文件扩展名是 .DBF，**类型**是 dBASE_DEV。

另请参见

[关于打印管理](#)

文件数量

历史文件的数量。输入不超过 4 个字符的值。

缺省情况下，Vijeo Citect 为每个设备创建一个数据文件。（此数据文件被称为 <filename.TXT> 或 <filename.DBF>，取决于设备是 ASCII 设备还是数据库设备。）您在此处指定的历史文件的数量是对于数据文件附加的。

注意： 如果不希望创建历史文件，必须在此处输入 0（零），并将 [Device]CreateHistoryFiles 参数设置为 0；否则，默认将创建 10 个历史文件。您还必须确认数据文件是固定大小的。（如果数据是累加的，文件最终将会耗尽全部硬盘空间。）

如果您指定 -1，数据将被附加到一个文件的尾部。

如果您记录报警、键盘命令或设备报表，则应指定要创建的文件数量以及每个文件的时间。

另请参见

[使用设备历史文件](#)

时间

一天中对历史文件的开头进行同步的时间，采用 hh:mm:ss（小时:分钟:秒）的形式。输入不超过 32 个字符的值。如果您接受以上缺省的历史文件数量，并且指定了时间和时间段，

则将创建 10 个历史文件。如果您没有指定时间，文件将在 0:00:00 时刻进行同步（即午夜时刻）。

如果您忽略时间和时间段，附加的历史文件仍会被创建（使用缺省的时间和时间段）。如果不希望创建历史文件，必须将 [Device]CreateHistoryFiles 参数设置为 0（零）。

时间段

历史文件的周期，采用 hh:mm:ss（小时:分钟:秒）的形式。输入不超过 32 个字符的值。或者可以：

- 输入一周中的某一天，例如星期一、星期二、星期三等，在这一天开始记录历史文件，并以此指定每一周的时间段。
- 输入一个月中的某一天，例如 1 号、2 号、3 号、4 号、5 号等，在这一天开始记录历史文件，并以此指定每一个月的时间段。
- 输入一年中的某月某日，例如 1 月 1 日、2 月 25 日等，在这一天开始记录历史文件。日期和月份之间必须用空格分隔。

如果您接受以上缺省的历史文件数量，并且指定了时间和时间段，则将创建 10 个历史文件。

如果没有指定时间段，缺省的时间段将是星期日（每周）。

如果您忽略时间和时间段，附加的历史文件仍会被创建（使用缺省的时间和时间段）。如果不希望创建历史文件，必须将 [Device]CreateHistoryFiles 参数设置为 0（零）。

集群名称

从先前使用“工程编辑器”中的服务器和集群命令定义的集群列表选择一个集群（最多 16 个字符）。See “集群定义”。如果这是一个单一集群系统，则此域可以为空白。

过程

选择运行过程以向设备发送数据的服务器（或客户机）的类型（最多 16 个字符）。此域用于在设备被激活时阻止历史文件的创建。如果没有历史过程，则此域可以为空白。

注释

任何有用的注释。输入一个不超过 48 个字符的值。

格式化设备中的数据

设备格式指定如何设置设备中的数据格式。此格式由设备类型和发送到设备的数据来决定。

- [打印机和 ASCII 设备格式](#)
- [dBASE 和 SQL 数据库设备格式](#)

另请参见

[使用数据库设备](#)

打印机和 ASCII 设备格式

格式指定打印机如何打印每一行的数据或写入 ASCII 文件，例如：

```
RFP3   Raw Feed pump 3   Overload   12:32:21
RFP9   Secondary Feed    Overtemp   13:02:45
```

生成报表时，设备格式将被忽略。为报表定义的格式（即报表格式文件）将用于将报表写入设备。

要包括 Vijeo Citect 数据，您必须指定域名并指定要打印或写入文件的每个域的宽度（可选）。此格式包含以下语法：

```
{<field name>, [width[, justification]]}
```

您需用大括号 ({}) 包括每个域，例如：

```
格式           {Tag,8}{Name,32}
```

在此情况下，以下两个字段将被打印或写入文件：标签（8 个字符）和名称（32 个字符）。对宽度的指定是可选的 - 如果不指定宽度，字段的宽度将由大括弧中的字符数决定，例如：

```
格式           {名称}
```

在此情况下，名称之后有四个位置 - 域用八个字母被打印或写入文件。

创建列表和表格

要设置每个字段中文本的对齐方式，请使用对齐标示符。可以选择三个对齐字符：L（左对齐）、R（右对齐）和 N（无） - 例如：

```
格式           {Tag,8,L} {Name,32,R}
```

对齐标示符是可选的 - 当忽略该属性时，字段为左对齐。在使用对齐标示符时，用户必须同时使用宽度标示符。

要显示列中的文本字段，请使用制表符 (^t) — 例如：

格式	{Tag,8}^t{Name,32}^t{Desc,32} {Time,8,R}
----	--

包括固定文字

用户可指定被打印或写入文件的确切文字来包括固定文字，例如：

格式	报警的名称:
----	--------

文字字符串的任何间隔也被包括在字符串中。

另请参见

[格式化设备中的数据](#)

[使用数据库设备](#)

dBASE 和 SQL 数据库设备格式

此格式指定数据库的结构（域名和域宽度）。此格式包含以下语法：

```
{<field name>, <width>}
```

必须用大括号 ({}) 包含每个域，例如：

格式	{Tag,8}{Name,32}
----	------------------

在本例中，数据库是用以下两个数据库域创建的：标签（使用 8 个字符），名称（使用 32 个字符）。每个数据库域的大小由采用此格式指定的宽度确定。（对齐字符将被忽略）。所有数据库域均为字符（字符串）域类型。

您可以针对数据库设备定义自己的域（以及标准 Vijeo Citect 域），例如：

打开设备后无设备文件可用，而且会用所需的更改生成一个新的数据文件。

另请参见

[格式化设备中的数据](#)

[使用数据库设备](#)

使用数据库设备

必须先打开数据库设备才能使用该设备。您可以同时打开多个设备。DevOpen() 函数返回一个整数句柄，定义每个设备，如下示例：

```
INT hRecipe;
hRecipe = DevOpen("Recipe");
```

使用数据库设备写 dBASE 记录

在某个数据库设备里写数据时，首先在设备的结尾附加新记录，然后在记录字段添加数据。对于 dBASE 数据库，DevAppend() 函数添加记录，DevSetField() 函数在字段中写数据。下面的函数在设备里写一个处方记录：

```
FUNCTION
WriteRecipeData(INT hDevice, STRING sName, INT Water, INT Sugar,
    INT Flour, INT Salt, INT Yeast, INT Milk)
    DevAppend(hDevice);
    DevSetField(hDevice, "NAME", sName);
    DevSetField(hDevice, "WATER", IntToStr(Water));
    DevSetField(hDevice, "SUGAR", IntToStr(Sugar));
    DevSetField(hDevice, "FLOUR", IntToStr(Flour));
    DevSetField(hDevice, "SALT", IntToStr(Salt));
    DevSetField(hDevice, "YEAST", IntToStr(Yeast));
    DevSetField(hDevice, "MILK", IntToStr(Milk));
END
```

使用数据库设备写 SQL 记录

在 Vijeo Citect 中使用一个 SQL 设备，用户不能使用所有 Cicode 设备函数；只能使用以下函数：

DevOpen()	DevClose()	DevGetField()	DevFind()	DevWrite()
DevNext()	DevSeek()	DevAppend()	DevWrite()	DevZap()
DevControl()	DevSetField()			

可使用 `DevWrite()` 函数在 SQL 数据库里写 Vijeo Citect 数据，但是用户必须添加一条新记录而且必须写到新记录的所有字段。如果不写入所有字段就没有数据写入数据库。

例如：

```
FUNCTION
WriteRecipeData(INT hDevice, STRING sName, INT Water, INT Sugar, INT Flour, INT
Salt, INT Yeast, INT Milk)
    DevWrite(hDevice, sName);
    DevWrite(hDevice, Water);
    DevWrite(hDevice, Sugar);
    DevWrite(hDevice, Flour);
    DevWrite(hDevice, Salt);
    DevWrite(hDevice, Yeast);
    DevWrite(hDevice, Milk);
END
```

当设备类型为 SQL 时，下列函数返回错误 267（文件模式无效）：

```
DevFlush()
DevPrev()
DevDelete()
```

当设备类型为 SQL 时，下列函数返回错误 263（无法读取文件）：

```
DevRead()
```

当设备类型为 SQL 时，下列函数返回错误 -1：

```
DevSize()
DevRecNo()
```

使用数据库设备定位并读取数据库记录

要从 dBASE 或 SQL 数据库设备读取数据，请使用 `DevFind()` 函数定位该记录，然后使用 `DevGetField()` 函数读取每个字段：

```

FUNCTION
GetRecipe(STRING sName)
  INT hDev;
  hDev = DevOpen("Recipe");
  IF hDev >= 0 THEN
    IF DevFind(hDev, sName, "NAME") = 0 THEN
      PLC_Water = DevGetField(hDev, "WATER");
      PLC_Sugar = DevGetField(hDev, "SUGAR");
      PLC_Flour = DevGetField(hDev, "FLOUR");
      PLC_Salt = DevGetField(hDev, "SALT");
      PLC_Yeast = DevGetField(hDev, "YEAST");
      PLC_Milk = DevGetField(hDev, "MILK");
    ELSE
      DspError("Cannot Find Recipe " + sName);
    END
    DevClose(hDev);
  ELSE
    DspError("Cannot open recipe database");
  END
END

```

删除使用数据库设备的记录

可以使用 `DevDelete()` 函数删除 dBASE 记录。下列 Cicode 函数可以删除 dBASE 设备中的所有记录:

```

FUNCTION DeleteRecords(INT hDev)
  WHILE NOT DevEOF(hDev) DO
    DevDelete(hDev);
    DevNext(hDev);
  END
END

```

要从 dBASE 数据库中删除所有记录, 请使用 `DevZap()` 函数:

```

FUNCTION DeleteRecords(INT hDev)
  DevZap(hDev);
END

```

要从 SQL 数据库中删除记录, 请使用 Cicode SQL 函数。

关闭数据库设备

完成对设备的操作后, 使用 `DevClose()` 函数将其关闭可释放 Cicode 系统资源:

```
DevClose(hRecipe);
```

定义一组设备：

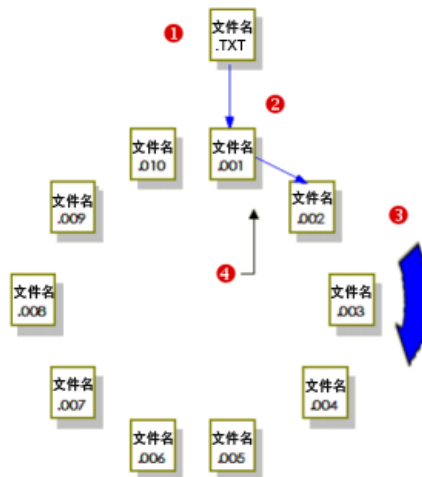
1. 选择**系统 | 组合**。随即会显示“组合”对话框。
2. 完成“组合”窗体。
3. 单击**添加**附加一条新记录，或单击**替换**修改现有记录。

另请参见

[使用设备历史文件](#)

使用设备历史文件

要使记录的数据的长期存储更易于组织和访问，Vijeo Citect 使用轮流历史文件系统存储历史数据。要使用此系统，必须指定要保持的设备历史文件数量。例如，如果要保持 10 个历史文件，则会按以下说明轮流保存：



1. Vijeo Citect 开始记录时，数据会写入名为 <文件名>.txt 或 <文件名>.dbf 的文件，这取决于设备的类型。
2. 第二个星期日的午夜，<文件名>.txt 文件将被重命名为 <文件名>.001 并创建新的 <文件名>.txt 文件。
3. 第二个星期日的午夜，<文件名>.001 文件将被重命名为 <文件名>.002，<文件名>.txt 将被重命名为 <文件名>.001，然后创建新的 <文件名>.txt 文件。
4. 10 星期后，第一个文件将被覆盖（第 11 星期将重复第一个循环）。

注意： 10 个历史文件为所有设备保存的缺省数据文件以外的文件。

缺省情况下，Vijeo Citect 使用 10 个文件（如果指定了历史文件）。可以通过指定要使用的文件数量来更改缺省设置。例如：

文件数量	20
注释	Vijeo Citect 将二十个文件用于数据

可指定的最大文件数量为 999。

还可以指定文件之间的周期，即使用新的历史文件的时间。例如：

周期	注释
1:00:00	每小时使用一个新文件
6:00:00	每六个小时使用一个新文件
72:00:00	每三天使用一个新文件
星期一	在每周的星期一开始使用一个新文件
15 号	在每个月的 15 号开始使用一个新文件
6 月 25 日	在每年 6 月 25 日开始使用一个新文件

注意： 在指定的一个星期或更长时间内可以观察到在系统性能方面所标记的改善。

也可以指定同步历史文件的具体开始时间，例如：

时间	注释
6:00:00	早晨 6:00 对文件进行同步
12:00:00	中午 12:00 对文件进行同步
18:30:00	下午 6:30 对文件进行同步

第一个文件实际不会在此时刻开始：第一个文件在启动运行时系统时开始。时间和周期共同确定新历史文件的创建时间，例如：

时间	周期
6:00:00	星期一

在上例中，Vijeo Citect 在每个星期一的早晨 6:00 创建新文件。如果在星期日下午 7:30 启动运行时系统，则第一个文件只包含 22.5 小时的数据。如果保持系统处于运行状态，则随后的文件将在每个星期一的早晨 6:00 开始，并包含整个一周的数据。

归档数据

要归档数据以进行长期存储，请在覆盖之前先备份历史记录文件。使用主程序组中的 Windows 文件管理器可以检查文件创建日期（历史记录文件）并备份文件。有关文件管理器的说明，请参阅 Windows 文档。

另请参见
[使用命令字段](#)

使用命令字段

可以使用以下字段（或组合）格式化命令记录设备：

字段名称	描述
{UserName,n}	命令发出时登录的用户的名称（用户名）。
{FullName,n}	命令发出时登录的用户的全名（全名）。
{Time,n}	命令发出的时间（短格式，hh:mm）。
{TimeLong,n}	命令发出的时间（长格式，hh:mm:ss）。
{Date,n}	命令发出的日期（短格式，dd:mm:yy）。
{DateLong,n}	命令发出的日期（长格式，日月年）。
{DateExt,n}	命令发出的日期（扩展格式，dd:mm:yyyy）。
{Page,n}	命令发出时显示的页面。
{MsgLog,n}	作为命令记录的 <i>消息日志</i> 属性发送的消息。
以下字段（在命令字段中） 仅对键盘命令 适用：	
{Arg1,n}	第一个键盘命令参数（如果有）。
{Arg2,n}	第二个键盘命令参数（如果有）。
...	
{Arg8,n}	第八个键盘命令参数（如果有）。
{Native_MsgLog,n}	作为命令记录的 <i>消息日志</i> 属性发送的消息本地语言版本。
其中 n 指定显示字段的大小。	

例如，可以按以下方式配置设备：

名称	KeyLog
格式	{Date,9} {MsgLog,27} {Arg1,3} by {FullName,11}

然后会借助以下配置创建键盘命令（对象、页面或系统）：

日志设备	KeyLog
键序列	### ENTER
日志消息	密度设置点改为

导致以下类型的输出：“Timothy Lee 于 1999 年 1 月 1 日将密度设置点改为 123”。

关于打印管理

Windows 打印机管理功能是专门为基于页面的打印机而设计的：激光打印机和共享网络打印机。打印机驱动程序不会在打印机上打印任何内容，直到整个页面完成为止；然后才打印该页面。这是首选的打印方法（在网络上共享打印机时），因为这将防止在多个操作人员同时使用打印机设备时造成的数据冲突。

但是，在记录报警或键盘命令时，此方法不适合。如果将报警日志记录发送到此类型的打印机，则在当前页面已满或超过 [DEVICE]FlushTime 参数（缺省为 10 秒）时，Vijeo Citect 会将数据清除到打印机。例如，如果每分钟发出一次报警，则每个报警都被打印在新的页面中（因为缺省清除时间小于报警频率）。

通过将输出写入一个文件，可以绕过 Windows 打印管理。将设备类型设置为 ASCII_DEV 并将文件名指定为 lpt1.dos、lpt2.dos 或 lpt3.dos（具体取决于打印机连接到的端口）。该打印机必须位于本地端口上，或捕获的网络打印机上。登录此设备时，数据将立即打印在该打印机上，无需额外换页。

为了执行正确的登录操作，请将一个打印机保留为登录打印机。此打印机应为本地打印机，不能位于网络服务器上。然后，您就可以将任何其他非登录打印输出（例如报表）发送到共享网络打印机或本地打印机。

另请参见
[使用设备](#)

Chapter: 31 与其他应用程序交换数据

您可以在 Vijeo Citect 和其他软件之间传输数据以用于存储、分析和后处理，或控制和调试您的 Vijeo Citect 系统。

Vijeo Citect 使用以下方法交换数据：

- 动态数据交换 (DDE)，其中 Vijeo Citect 可作为：
 - 向请求客户机提供标签值的 **DDE 服务器**
 - 从其他应用程序请求数据的 **DDE 客户机**。
- 开放数据库连接 (ODBC)，其中 Vijeo Citect 用作 ODBC 服务器，允许其他应用程序直接读取 Vijeo Citect 变量。
- 通过使用通用**外部数据库**，Vijeo Citect 和其他应用程序可以使用相同的数据库来存储和共享信息。

Vijeo Citect 还支持从外部数据库导入和链接变量标签数据。请参见 [链接、导入和导出标签](#)。

使用 DDE（动态数据交换）

通过 Microsoft Windows DDE，同一台计算机中的不同 Windows 应用程序之间可进行连续自动的数据交换，而无需任何用户干预。例如，公司的产品组可以使用电子表格应用程序以图形方式表示工厂楼层数据（产品输出）。这可以使用 DDE 直接从 Vijeo Citect 读取数据来通过最新实时数据动态更新。

Windows DDE 使用 DDE 协议在共享数据的应用程序之间发送消息。

DDE 客户机应用程序（用于请求数据或服务）与 DDE 服务器应用程序（用于提供数据或服务）之间会发生动态数据交换。DDE 客户机通过建立与 DDE 服务器的会话以及请求数据或服务来启动交换。DDE 服务器通过向 DDE 客户机提供数据或服务来响应这些请求。DDE 客户机会在不再需要 DDE 服务器的数据或服务时终止会话。

注意：由于 DDE 协议不是为高速数据传送设计的，因此 DDE 仅适合在数据通讯速度不重要时使用。

- 有关 DDE 会话的信息，请参见 [DDE 会话和客户机语法](#)。
- 要在同一台计算机的应用程序之间建立 DDE 会话，请参见 [设置 DDE 会话](#)。
- 要在同一网络中的不同计算机中运行的应用程序之间建立 DDE 会话，请参见 [网络 DDE](#)。
- 要通过 Vijeo Citect 标签数据库直接建立 DDE 会话，请参见 [使用 DDE 连接到 Vijeo Citect 标签数据库](#)。

注意：使用 DDE 读取或写入 Vijeo Citect 标签时，可能会无意中添加到 Vijeo Citect 许可点数。一旦动态点数大于许可点数，软件保护机制将终止 Vijeo Citect 运行。因此，通过 DDE 访问标签时，了解已使用的点数非常重要。有关详细信息，请参见《安装与配置指南》中的“许可点数”。

DDE 会话和客户机语法

参与动态数据交换的两个应用程序被认为是在进行 **DDE 会话**。启动此会话的应用程序是 **DDE 客户机**，响应 DDE 客户机的应用程序是 DDE 服务器。

一个应用程序可以具有数个同时运行的 DDE 会话。在某些会话中，此应用程序可以是 DDE 客户机（请求数据或服务），在另一些会话中则是 DDE 服务器（数据/服务提供方）。DDE 会话中的每一个请求或响应指定要发送或接收的数据或服务。

注意：DDE 会话有时被参考为**通道或链接**。

由 DDE 客户机在其尝试与 DDE 服务器建立 DDE 会话时发送的语法包含以下三个部分：

- 要从中检索数据的应用程序的名称。
- 包含要检索数据的文件或主题的名称。
- 被请求的单元范围、值、域或数据项。

它们采用以下格式合并：

```
<DDE server application name>|<DDE Topic name>!<DDE Data item name>
```

其中：

- **<DDE server applicationname>** 表示 DDE 服务器应用程序。
- |（管道字符）无空格地将 DDE 服务器应用程序名称与 DDE 主题名称分开。
- **<DDE Topic name>**表示数据的上下文。对于运行基于文件的文档的 DDE 服务器，DDE 主题名称通常为文件名。对于其它 DDE 服务器，它们是其它 DDE 应用程序的专用字符串。
- !（感叹号）无空格地将 DDE 主题名称与 DDE 数据项名称分开。
- **<DDE Data item name>**是一个字符串，表示可由 DDE 服务器在 DDE 交易期间传递到 DDE 客户机的数据项。在某些情况下，DDE 数据项名称是可选的。要了解特殊情况，请参见 DDE 应用程序文档。

注意：在上面的 DDE 客户机语法结构范例中，尖括号中显示的每个占位符（<placeholder>）都应替换为它所描述的项目的实际名称。其中包含的尖括号和占位符不应包括在语句中，在此处显示只为便于您理解。

由于 DDE 协议是在出现长文件名之前设计的，DDE 仅支持使用短（8 个字符）文件名。要克服这一限制，用单引号将 DDE 语法的三个部分分别括起来。例如：

```
Citect|Variable!'Process Variable 1'
```

这指示 DDE 将引号内的字符看作字符串，从而允许它们包含长文件名、空格 ()、管道符号 (|)、感叹号 (!) 或任何其它非字母数字字符。

另请参见
[设置 DDE 会话](#)

设置 DDE 会话

DDE 协议本身并不支持应用程序的启动，因此在进行任何 DDE 会话之前，DDE 客户端应用程序和 DDE 服务器应用程序都必须已经在运行（除非将调用程序编码为在需要时检测并启动 DDE 服务器应用程序）。

开始 DDE 会话时，DDE 客户端会使用 DDE 客户端语法（这包含 DDE 服务器应用程序名称、主题或文件名称和请求中的数据项名称）请求 DDE 服务器的服务。有关 DDE 客户端语法的详细信息，请参见 [DDE 会话和客户端语法](#)。

要将应用程序设置为 DDE 客户端，即要从 DDE 服务器应用程序请求数据，需要在 DDE 客户端语法中按下述方式使用适当的值：

DDE 服务器应用程序名称

DDE 服务器名称通常是 DDE 服务器应用程序的名称，例如 Vijeo Citect 的 DDE 服务器名称为“Citect”，Microsoft Excel 的 DDE 服务器名称为“Excel”，Microsoft Word 的 DDE 服务器名称为“WinWord”，Microsoft Access 的 DDE 服务器名称为“MSAccess”。多数 DDE 服务器都只响应一个名称。

DDE 主题名称

Vijeo Citect 的 DDE 主题名称为“Data”（如果使用 Cicode DDE 函数）或“Variable”（如果使用 Vijeo Citect 作为 DDE 服务器并且希望直接访问变量标签数据库）。Microsoft Excel 的 DDE 主题名称为工作单的名称（可能还包括位于方括号中的工作簿的名称）。Microsoft Word 的 DDE 主题名称为文档名称。Microsoft Access 的 DDE 主题名称为数据库名称和表名称、查询名称或下面注意中详细说明了的 SQL 字符串：

注意：用于访问 Microsoft Access 数据库的 DDE 主题名称部分的正确 DDE 客户端语法构造如下：`"<DataBaseName>; TABLE <TableName>".`

`<DataBaseName>` 占位符用于后跟分号 (;) 的 Access 数据库的名称。可能还必须将文件路径包括在其中；不过，情况也可能并不如此（即，如果已知 Access 将以打开的目标文件运行）。.MDB 后缀为可选（因为 .MDB 是 Access 数据库的缺省后缀），除非包括了完整的路径。

TABLE <TableName> 是要让 Access 与哪个表数据会话的命令字符串。DDE 还支持使用 **QUERY <QueryName>** 或 **SQL <SQLString>** 代替 **TABLE <TableName>**。

在 '<DataBaseName>' 占位符后使用分号 (;)，在 DDE 字符串语法中将 UPPERCASE 用于 'TABLE'、'QUERY' 和 'SQL' 命令都是必须的。整个部分必须用引号 (") 括起。

DDE 数据项名称

Vijeo Citect 的 DDE 数据项名称取决于所使用的 DDE 主题名称。使用 'Variable' 作为 DDE 主题名称直接访问变量标签数据库时，DDE 数据项名称为 Vijeo Citect 变量标签名称。使用 'Data' 作为 DDE 主题名称访问使用 Cicode DDEPost() 函数公布的值时，DDE 数据项名称为该公布名称。

Microsoft Excel 的 DDE 数据项名称为行号列号格式（例如 R1C1）的单元格范围。Microsoft Word 的 DDE 数据项名称为书签名称。Microsoft Access 的 DDE 数据项名称取决于所使用的主题名称。有关详细信息，请参阅 Microsoft Access 帮助。

注意： 这些 Vijeo Citect DDE 帮助主题和示例原是为 Windows 3.xx 编写的，后来又针对 Windows 95 上的 Office 95 进行了更新。自此之后，Microsoft 在 Office 2000 和后续版本中引入了安全措施，缺省情况下，是不允许 DDE 客户端访问 Office 应用程序的。有关安全的详细信息，请参见 [将 DDE 用于 Microsoft Office 应用程序](#)。

Vijeo Citect 可以根据需要作为 DDE 服务器和 DDE 客户端。要将 Vijeo Citect 设置为使用 DDE，请参见 [通过 DDE 交换 Vijeo Citect 数据](#)。

Vijeo Citect 随附了一个 Excel 工作簿文件，它包含可帮助您将正确的 DDE 客户端语法链接从 Vijeo Citect 运行工程标签数据库直接插入到 Excel 工作表的宏。

DDE 函数类型

Cicode 中有两类 DDE 函数，即原始的 **DDE** 函数和更新的 **DDEh** 函数。

DDE 函数

原始的 Cicode DDE 函数不返回 DDE 通道号，旨在免除用户对 DDE 通道的管理。DDERead()、DDEPost()、DDEWrite() 和 DDEExec() 函数各自执行单一的数据交换。这些函数中的每一个都启动与外部应用程序的 DDE 会话，发送或接收数据（或命令），然后终止会话 - 多合一操作。

DDEh 函数

Cicode DDEh 函数的引入是为了对 DDE 通讯担负更多的控制任务，尤其是对于网络 DDE 和需要明确终止并重新启动 DDE 通道（例如，从表中删除行之后）数据交换。

DDE 句柄 (*DDEh...*) 函数返回指向会话的句柄 - 一个 DDE 通道号。

可将 DDEh 句柄函数用于网络 DDE 和 Access DDE。

另请参见

[通过 DDE 交换 Vijeo Citect 数据](#)

通过 DDE 交换数据

Vijeo Citect 运行时可以交换数据作为 DDE 服务器或 DDE 客户机。

当 Vijeo Citect 允许其他应用程序访问其数据时，它将作为 DDE 服务器。当作为 DDE 服务器时，Vijeo Citect 运行时可以：

- 自动允许 DDE 访问完整的变量标签数据库，不需要额外的步骤
- 通过使用 DDE 公布选择 Vijeo Citect 数据，允许访问选定的变量值

当 Vijeo Citect 请求其他应用程序向它提供数据访问权限时，它将作为 DDE 客户机。当作为 DDE 客户机时，Vijeo Citect 运行时可以：

- 直接从其他应用程序读取数据
- 直接向其他应用程序写入数据

注意：您也可以使用 DDEExec() 函数从 Vijeo Citect 在其他应用程序中执行命令。同样，您可以从其他应用程序运行 Cicode 函数，方法是通过该应用程序的 DDE 执行命令传递函数。

另请参见

[使用 DDE 连接到 Vijeo Citect 标签数据库](#)

使用 DDE 连接到标签数据库

Vijeo Citect 运行时可作为 DDE 服务器，无需额外安装，DDE 便可访问全部标签变量数据库。

为了建立 DDE 和 Vijeo Citect 标签变量之间的链接，应使用 DDE 客户端语法。有关语法的详细信息，请参见 [DDE 会话和客户端语法](#)。

DDE 客户端调用时，DDE 应用程序的名称必须是 "Citect"，DDE 标题名称必须是 "Variable"，DDE 数据项的名称必须是 Vijeo Citect 标签名称。

例如，含有下列公式的 Excel 单元格可以访问 PV1 标签值：

```
=Citect|Variable!PV1
```

如果 Vijeo Citect 标签变量名称含有空格和非字母数字字符，DDE 客户端调用语法的 DDE 数据项部分必须放在单引号之间。例如：

```
=Citect|Variable!'Process Variable 1'
```


Vijeo Citect 运行时和 DDE 客户端程序（如：Excel）必须运行在同一台计算机上。关于 DDE 会话的信息，请参见 [DDE 会话和客户端语法](#)。

在同一台计算机的不同应用程序间建立 DDE 会话的方法请参见 [设置 DDE 会话](#)。

在同一网络上运行在不同计算机上的应用程序之间建立 DDE 会话的方法请参见 [网络 DDE](#)。

使用 DDE 公布特定数据

您的标签命名约定可能与 DDE 不兼容，或者不适合在 DDE 调用中使用。Vijeo Citect 提供了在 DDE 中以不同名称发布特定标签的能力。这涉及使用 DDEpost 函数。

要使选定的 Vijeo Citect 变量值或计算结果可用于当前运行在相同计算机上的外部 DDE 客户端应用程序，请使用 Cicode DDEPost() 函数让 Vijeo Citect 运行时跟一个 DDE 服务器一样。

这种转换是单向的，这使得外部 DDE 客户端应用程序（如 Excel、Word 等）可以使用 DDE 从 Vijeo Citect 读取值。该客户端应用程序无法更改 Vijeo Citect 中的该值。

可以使用此函数以不同于数据来源的名称提供数据。例如，下面的示例以“Process1”提供变量标签 PV1 的值。

下面的 Cicode 示例即是使用 DDE 公布变量 PV1 的值：

```
DDEPost("Process1", PV1)
```

一旦公布之后，即可从比如包含以下公式的 Excel 单元格中访问此值：

```
=Citect|Data!Process1
```

或者从包含以下函数的 Microsoft Word 域中访问：

```
{DDEAuto Citect Data Process1 }
```

注意：

- 如果要通过 Microsoft Word DDE 域链接到公布的值，该值的名称（例如，Process1）必须不超过 8 个字符或包含空格。不过，如果将长数据项名称用单引号引起（例如“Process Variable 1”），则 Microsoft Excel 能接受长名称。
- 访问公布的值时，必须将数据作为 DDE 主题。请参见 [设置 DDE 会话](#)。

DDE 连接的此方法要求 Vijeo Citect 运行时和 DDE 客户端应用程序（例如 Excel 或 Word）同时运行在相同的计算机上。建立起 DDE 连接后，更新值一旦可用 DDE 客户端即会正常显示 Vijeo Citect DDE 公布的更新值，通常在公布后几毫秒之内即显示。

不足之处在于，只要 Vijeo Citect 运行时关闭，即使随后重新启动 Vijeo Citect 运行时，DDE 链接的此公布方法也会中断。DDE 客户端应用程序可能不检测中断，因为对 DDE 客户端（例如 Excel）中数据的更新仅在 DDE 服务器（Vijeo Citect 运行时）每次公布之后进行。如果没有进行后续公布，而且在此种情形下也不会进行任何公布（因为 DDE 链接已中断），则 DDE 客户端应用程序就收不到任何更新，随后显示的可能是已经过时的数据。此中断状态会一直保持到 DDE 客户端应用程序中的 DDE 链接被刷新或 DDE 客户端应用程序被重启。

另请参见

[将值写入 DDE 应用程序](#)

将值写入 DDE 应用程序

要直接将 Vijeo Citect 变量中的参数值写入外部 DDE 服务器应用程序（当前与 Vijeo Citect 运行在同一台计算机上），可使用 Cicode DDEWrite() 函数。

例如从 Vijeo Citect 向 Microsoft Office 写入数据（使用 Vijeo Citect 作为 DDE 客户端，Office 应用程序作为 DDE 服务器），可以使用下列 Cicode DDEWrite() 函数，例如：

```
! Write PV1 to Excel
! Assumes the existence of Sheet1
DDEWrite("Excel", "Sheet1", "R1C1", PV1);

! Write PV1 to Word
! Assumes the existence of TestDDE.doc already loaded in Word
! containing the bookmark named TagPV1
DDEWrite("Word", "TestDDE", "TagPV1", PV1);
! MS Access does not support direct DDE writes.
```

此 DDE 函数是单向的，所以为了更新远程 DDE 服务器应用程序中的标签值，每次 Vijeo Citect 变量中的参数值变化时都需要调用此函数。

直接写入 DDE 服务器以对 DDE 服务器有一定了解为前提。在上面的例子中，电子表格或者文档必须存在并且为了 DDE 通信成功 Excel 或 Word（适当的）必须正在运行。

注意：

- 除了使用一次性 DDEWrite()，还可以使用多用 DDE 句柄函数 DDEhPoke()。
- 确认在另一个应用程序中启用了远程请求。例如，在 Excel 中，您必须确认取消选中了 **忽略其他应用程序** 复选框（缺省设置）。在 Excel 2003 中，使用 **工具 | 选项 | 常规** 来调整设置。
- “高”安全性设置（如果选定）在 Microsoft Office 2000 以及更高版本中，不会影响远程 DDE 客户端请求与这些 Office 应用程序配合使用。请参见 [将 DDE 用于 Microsoft Office 应用程序](#)。

另请参见

[读取 DDE 应用程序参数值](#)

读取 DDE 应用程序参数值

要读取直接来自外部 DDE 服务器应用程序（当前与 Vijeo Citect 运行在同一台计算机上）的 Vijeo Citect 变量中的参数值，可使用 `Cicode DDERead()` 函数。例如：

```
PV1 = DDERead("Excel", "[Book1]Sheet1", R1C1);
```

读取 Excel Book 1 的表单 1 中的第 1 行，第 1 列的数据，并将其存储在 Vijeo Citect 变量 "V1" 中。此 DDE 函数为单向的，只要 Vijeo Citect 中需要更新参数值时就会调用此函数。

从 DDE 服务器进行读取以对 DDE 服务器有一定了解为前提。在上面的示例中，Excel 必须运行，且必须存在名称适当的电子表格。Vijeo Citect 变量 PV1 也必须进行预先声明。

注意： 与其使用一次性 `DDERead()`，您可以使用多用 DDE 句柄函数 `DDEhRequest()`。

确认在另一个应用程序中启用了远程请求。例如，在 Excel 中，您必须确认取消选中了**忽略其他应用程序**复选框（缺省设置）。在 Excel 2003 中，使用**工具 | 选项 | 常规**来调整设置。

“高”安全性设置（如果选定）在 Microsoft Office 2000 以及更高版本中，不会影响远程 DDE 客户端请求与这些 Office 应用程序配合使用。请参见 [将 DDE 用于 Microsoft Office 应用程序](#)。

将 DDE 用于 Microsoft Office 应用程序

Microsoft 已在 Office 2000 和更高版本中引入了安全措施，缺省情况下会阻止来自 DDE 客户端的 Office 应用程序。请参见 [Microsoft Office 安全性](#)。

Microsoft Office 应用程序支持在 DDE 中使用各种程度的长文件名。请参见 [DDE 中的长文件名](#)。

要在 Microsoft Excel 中启用 DDE 远程请求，必须确认**忽略其它应用程序**复选框被清除（缺省设置）。在 Excel 2003 中，使用**工具 | 选项 | 常规**可调整此设置。

DDE 中的长文件名

根据 MSDN 知识库文章 Q109397，DDE 不支持长文件名，因此对于长于 8 个字符的目录和文件名，必须使用 DOS 别名（即，C:\mydocu-1\file.mdb）。

当用作 DDE 客户端时，不同的 Microsoft Office 应用程序对长文件名的支持情况有所不同。例如，Microsoft Word 似乎不支持使用超过 8 个字符的 DDE 数据项名称，而 Microsoft Excel 在用单引号引起的情况下则接受长数据项名称。有关示例，请参见 [使用 DDE 公布精选的 Vijeo Citect 数据](#)。

Microsoft Office 安全性

为了保证数据安全，Microsoft Office 2000 以及之后版本缺省将其安全设置设定为高。要查看或更改 Excel 或 Word 中的安全级，请选择**工具 | 宏**，点击**安全性**，然后点击**安全级别**分页。

- 如果将安全级设置为**高**（缺省设置），则无法与外部 DDE 服务器进行通信，除非它们获得数字签名和信任。在使用 DDE 函数的 Excel 单元格中所看到的全是 #N/A，并且没有关于 DDE 函数为何无法发挥作用的进一步说明。“高”安全性设置（如果选定）似乎不影响远程 DDE 客户端将这些 Office 应用程序作为 DDE 服务器使用。
- 如果将安全级设置为**中**，则会询问您是否要运行未获数字签名和信任并且被 DDE 函数引用的任何 DDE 服务器。
- 如果将安全级设置为**低**，则无论是否获得数字签名和信任，均会运行所有外部 DDE 服务器。

注意：如果需要从 Microsoft Office 操作另一个应用程序的对象，请考虑使用“OLE 自动化”。

网络 DDE

网络 DDE 是在网络之间使用的 DDE 协议的版本。有关 DDE 的信息，请参见 [使用 DDE \(动态数据交换\)](#)。有关设置 Vijeo Citect 使用 DDE 的详细信息，请参见 [通过 DDE 交换 Vijeo Citect 数据](#)。

与在同一台计算机上运行的应用程序之间建立 DDE 会话相同，网络 DDE 会话使用相同的 DDE 协议在应用程序（在连接到共同网络的单独计算机中运行）之间共享数据。

网络 DDE 是 Windows 服务，用于初始化和维护网络连接、安全和通过网络使用 DDE 所需的共享文件空间，它必须同时在两台计算机上运行。有关启动的详细信息，请参见 [启动网络 DDE 服务](#)。

必须为网络 DDE 服务器应用程序计算机中的网络 DDE 服务器应用程序手动创建网络 DDE 信任的共享。对于要连接的应用程序和主题，上述操作会对 DDE 服务器应用程序计算机中的网络 DDE 服务器做出指示。网络 DDE 客户机应用程序随后可以与共享名称进行通讯。有关详细信息，请参见 [设置网络 DDE 共享](#)。

网络 DDE 客户机通过连接到网络 DDE 服务器计算机中的网络 DDE 共享启动网络 DDE 会话。有关详细信息，请参见 [使用网络 DDE](#)。

启动网络 DDE 服务

要使网络 DDE 发挥作用，必须先安装 NetDDE.EXE 并使其同时在两台机器上运行，然后再尝试实施网络 DDE 会话。NetDDE.exe 是一个 Windows 服务系统文件，用于与 Network DDE 所使用的共享动态数据交换通讯。它没有图形用户界面（作为后台 Windows 服务运行）。

在尝试连接前，需要初始化 Network DDE 服务的自动激活，或手动在两台机器上运行 NetDDE.EXE。

要手动启动网络 DDE 服务：

- 在 Windows 开始菜单中，点击**开始 | 运行**，键入“netdde”（不加引号）并按 **Enter** 键。在两台机器上都执行此操作。

要使网络 DDE 服务在机器启动时自动启动：

- 对于 Windows 2000 及更高版本，使用 Windows 服务管理器（选择**开始 | 控制面板 | 管理工具 | 服务**）将网络 DDE 服务从**手动**设置为**自动**。为此，右键单击该服务并从弹出式菜单中选择**属性**。从**常规**分页的**启动类型**域中选择**自动**。点击**确定**。关闭所有窗口并重新启动机器。

要验证 NetDDE 服务是否正在运行：

- 运行时，Windows 任务管理器会在**进程**分页中列出 NetDDE.exe。要查看 Windows 任务管理器，按 **CTRL+ALT+DEL**。
- 服务管理工具会同时列出网络 DDE 和网络 DDE DSDM 的状态。要查看 Windows 服务管理器，选择**开始 | 设置 | 控制面板 | 管理工具 | 服务**。

注意：如果只将 Microsoft 客户机服务用于 NetWare 网络，则必须启用兼容 NW IPX/SPX/NetBIOS 的协议才能加载 NetDDE.exe。

为测试网络 DDE 是否能在同一网络中的两台机器间运行

Microsoft Windows 随附了一个名为“聊天”的网络 DDE 应用程序。该程序安装在 system32 文件夹中。

1. 在 Windows 开始菜单中，点击**开始 | 运行**，键入“winchat”（不加引号）并按 **Enter** 键。在两台机器上都执行此操作。
2. 在一台机器上，选择聊天菜单中的**会话 | 拨号**或点击拨号按钮。将显示**选择计算机**对话框。
3. 从列表中选择另一台计算机，然后点击**确定**。聊天将尝试在这两台计算机间建立网络 DDE 会话。

注意：如果聊天未在另一台计算机上运行，它将中止连接并声明另一台计算机没有应答。但如果聊天已在另一台计算机上运行，它将保持拨号状态直至收到应答。

4. 在另一台机器上（当其正在被拨号期间），选择聊天菜单中的**会话 | 应答**或点击应答按钮。键入一条消息，该消息将显示在另一台机器上的聊天窗口中。在任一台机器挂起之前，此会话将继续。

一旦有聊天会话被建立，它就可以证明两台机器已被正确设置并且可以处理网络 DDE 会话。如 [设置网络 DDE 共享](#) 中所述，您可以使用 DDEShares.exe 应用程序查看 Chat\$ 的共享属性。

并非必须运行聊天才能将网络 DDE 与 Vijeo Citect 一起使用。此网络 DDE 测试仅使用聊天作为示例，以确认两台机器间的 DDE 功能是否正常。确定该网络 DDE 已生效后，关闭聊天窗口，为自己的网络 DDE 服务器应用程序创建**网络 DDE 受信任共享**，然后使用 Network DDE 连接到该共享。参见 [连接到网络 DDE 共享应用程序](#)。

设置网络 DDE 共享

为了能够创建经由网络的 DDE 链接，起网络 DDE 服务器作用的计算机必须设置为提供**网络 DDE 共享**以建立一个网络 DDE 通道。

注意：如果要在运行于不同计算机上的两个独立应用程序之间使用 DDE，您只需要创建一个 DDE 共享，并且只需要在包含将作为 DDE 服务器的应用程序的计算机上创建该 DDE 共享。

用户可使用 Windows DDESHARE.EXE 工具管理 DDE 共享。所有 Microsoft 操作系统随附了 32 位版本，位于 Windows\System32 子目录下。

如何手动启动 DDE 共享工具：

- 在 Windows 开始菜单上，点击**开始 | 运行**，键入“ddeshare”（不包括引号）并按 **Enter** 键。

当 DDEShare.EXE 运行时，它会显示 DDE 共享工具窗口，该窗口包含用于启动“DDE 共享”对话框和“DDE 信任共享”对话框的两个图标：

在 DDE 共享工具中，双击左侧图标（不带复选标记）以显示“DDE 共享”对话框：

“DDE 共享”对话框用来创建、管理和删除您计算机上的全局 DDE 共享，并且可以用来查看网络上任何计算机的 DDE 共享。可以使用此对话框确定同一网络任何计算机上的可用共享的名称。从 DDE 共享菜单中，选择**共享 | 选择计算机**，并从列表中选择您感兴趣的计算机名称。

DDE 共享

有三种 DDE 共享类型：旧风格、新风格和静态。Vijeo Citect 仅支持静态类型。静态共享的名称符合以下规则

```
<ShareName>$
```

因此要将 Vijeo Citect 服务器计算机设置为网络 DDE 共享，使用名称“Citect\$”作为该计算机上的共享名称。要针对合适的 DDE 链接展示 Vijeo Citect 运行期变量标签数据库，使用词“Variable”作为 DDE 共享主题名称。

注意： 需要使用拖尾的美元符号 (\$) 作为 DDE 共享名称语法的一部分。

要创建 DDE 共享，请执行以下操作：

1. 在“DDE 共享”对话框中，点击**添加共享**。将出现 **DDE 共享属性**对话框。确认所有域均为空，或处于未选中状态，唯一的例外是必须选中**允许访问所有项目**单选按钮。
2. 点击**许可**。将出现“DDE 共享名称许可”对话框。
3. **读取和链接**是缺省的许可设置。如果要向 DDE 共享应用程序中写入数据，将此许可更改为**完全控制**。
4. 点击**确定**。
5. 点击**确定**保存共享，然后返回 **DDE 共享**对话框。

另请参见

[使用 DDE 受信任共享](#)

使用 DDE 受信任共享

网络 DDE 客户机用户从远程计算机连接到网络 DDE 共享时，网络 DDE 仅在以下两种情况下接受请求：

- 创建共享的用户已授予共享受信任的状态。
- 创建共享的用户当前登录到服务器计算机。

要链接到 Vijeo Citect 标签数据库并允许使用网络 DDE 从外部应用程序写入的操作，必须授予 DDE 客户机计算机适当的受信任状态。

如何创建受信任共享：

1. 在“DDE 共享”对话框中，加亮新的 'Citect\$' 共享条目，然后单击**信任共享以显示受信任共享属性**对话框。
2. 单击**初始化应用程序**，建立到 DDE 共享的新连接。
3. 单击**确定**。

如何查看受信任共享：

1. 在 DDESHARE 工具中，双击正确的图标（带有复选标记）以显示“DDE 受信任共享”对话框。
2. “DDE 受信任共享”对话框列出当前用户上下文中信任的 DDE 共享。可以查看和修改受信任共享属性，也可以从受信任共享的列表中删除 DDE 共享。
3. 设置完成后，关闭 DDE 共享工具对话框。

使用网络 DDE

Microsoft 网络 DDE 服务必须在两台都使用网络 DDE 的进行通讯的计算机中运行。有关启动的详细信息，请参见 [启动网络 DDE 服务](#)。

网络 DDE 客户机建立与网络 DDE 服务器应用程序的会话之前，网络 DDE 服务器应用程序计算机必须已设置了**网络 DDE 共享**。有关详细信息，请参见 [设置网络 DDE 共享](#)。

注意：不能使用网络 DDE 连接到同一台计算机中的共享应用程序。只能使用网络 DDE 连接到另一台计算机（此计算机必须也在同一个网络中）中的共享应用程序。

要连接到网络 DDE 共享应用程序，请使用 DDE 语法的已更改版本，该版本将 "<ApplicationName>" 替换为 "<ComputerName>\NDDE\$", 将 "<TopicName>" 替换为网络 DDE 服务器共享 "<ShareName>", 并继续使用 "<DataItemName>" 作为常用配置。

乍看上去，似乎没有方法可以在网络 DDE 语法调用中指定 DDE 应用程序或主题名称，而实际上是有这样的方法的。但 DDE 应用程序和主题名称在 DDE 服务器共享设置中定义。因此，网络 DDE 服务器计算机接收包含共享名称的调用（来自网络 DDE 客户机）时，它会识别要连接的应用程序和主题。请参见 [连接到网络 DDE 共享应用程序](#)。

连接到网络 DDE 共享程序

调用时通过将 DDE 应用程序名称和 DDE 主题名称更改为 DDE 服务器计算机名称和 DDE 服务器共享名称这种方式，网络 DDE 客户机用普通 DDE 客户机语法指定远程 DDE 服务器共享。有关 DDE 客户机语法的详细信息，参见 [DDE 会话和客户机语法](#)。

遵循网络 DDE 客户机语法，DDE 应用程序名称使用下面单引号括起来的字符串进行替换：

```
'\\<ComputerName>\NDDE$'
```

此处的 "<ComputerName>" 是运行 DDE 服务器程序的计算机名称，"NDDE\$" 通知远程 Windows 系统正在调用的 DDE 客户机请求建立一个网络 DDE 通道。用户不得省略 "NDDE\$" 字符串，否则不起作用。

DDE 的主题名称由以下带单引号的字符串取代：

```
'<ShareName>'
```

此处的 "<ShareName>" 是之前在 DDE 服务器上建立的 DDE 可信共享的名称。DDE 服务器上的 DDE 共享包含创建网络 DDE 与哪一个应用程序和主题链接的细节。绝大多数情况下，DDE 服务器共享名字以 "\$" 字符结尾。

注意：远程计算机上可共享的 DDE 应用程序名称和 DDE 主题名称的组合必须使用不同的 DDE 共享名称。主题不能设定为通配符 (*)。

例如，使用如下标准创建网络 DDE 链接：

- 变量标签名称："PV1"

- 服务器名称: "PlantSvr"
- 远程 DDE 共享名称: "Citect\$"

构建网络 DDE 客户机调用应包含如下代码:

```
'\\PlantSvr\NDDE$'!'Citect$'!PV1
```

在 Excel 中, 下列公式可以直接放在工作表的单元格中:

```
='\PlantSvr\NDDE$'!'Citect$'!PV1
```

如果提示输入用户名和密码, 可使用某个 DDE 服务器上的可用权限。

注意: 不得省略 DDE 语法管道符号 (|) 或者感叹号 (!), 字符中也不能包含在单引号 (') 中。

Vijeo Citect 附带了一个 Excel 工作簿文件, 它包含可帮助您将正确的 DDE 客户端语法链接从 Vijeo Citect 运行工程标签数据库直接插入到 Excel 工作表的宏。

使用 Citect 标签 Excel 宏

Vijeo Citect 提供 Citect 标签 Excel 宏, 通过它可直接在 Excel 工作表单元中显示 Vijeo Citect 变量的参数值 (因此无需使用 Cicode DDE 函数)。宏包含在工作簿 `ddeformu.xls` (位于 `\Vijeo Citect 7.10\bin` 目录) 中, 这些宏更新为支持 CitectSCADA 版本 5.41 和更高版本的网络 DDE。

注意: Microsoft Office 97 附带的 Microsoft Excel 版本 8 提供宏病毒保护, 防止潜在的恶意宏运行。要在 Excel 8 中启用宏并使用 `ddeformu.xls` 提供的功能, 请从主菜单中选择 **工具 | 选项**, 然后在“常规”标签中清除 **宏病毒保护** 选项。

Microsoft Office 2000 附带的 Microsoft Excel 版本 9 提供安全级别, 缺省情况下会禁止宏。要在 Excel 9 或更高版本中启用宏并使用 `ddeformu.xls` 提供的功能, 请从主菜单中选择 **工具 | 宏 | 安全性**, 然后选择 **中或低**。

如果启用了 Excel 安全性设置, 则尝试打开 `ddeformu.xls` 时, Excel 会告知用户文件包含宏。要启用 Citect 标签功能, 请选择 **启用宏**。

启动时, Citect 标签宏会希望 Vijeo Citect 运行时在同一台计算机中操作。如果不是, Excel 会显示请求启动 `citect.exe` (此操作会由于 `citect.exe` 不存在而失败, 除非运行的是 Citect 版本 3 或更低版本) 权限的对话框。如果进一步选择 **是**, 则会为不存在的程序搜

索系统“路径”，随后报告无法找到应用程序。如果选择否，则通过工作表保存以前的参数值，这些参数值是最初显示的参数值，并在更新时替换为 '#REF!'。在任何情况下，范例工作表中都不会显示有效参数值，直到启动了 Vijeo Citect 运行时并随后刷新或重新启动了 Excel。

Citect 标签宏希望将 citect.ini 存放在安装期间选择的 vijeo citectv7.10 用户和数据文件夹的 config 文件夹中。如果在该位置未找到文件，将不会搜索其它位置，而显示“读取 Citect.INI 时出错”对话框，请求适当的位置。输入包括文件名在内的完整路径，然后清除启动时恢复缺省值，防止下次启动宏时发生同样的情况。如果要使用备用 INI 文件，输入它即可。

运行后，在包含四个附加菜单项的 Excel 中右键单击菜单，这样可在执行右键单击事件时对鼠标指针位置下的单元执行两个新的工作簿相关命令和两个新的 Vijeo Citect 相关命令。ddeformu.xls 提供的新菜单项为：

- Citect 设置 - 工作簿命令
- Citect 获取标签 - 工作簿命令
- Citect 选择标签 - Vijeo Citect 命令
- Citect 选择趋势 - Vijeo Citect 命令

注意：此功能仅与 Excel 版本 5.00 或更高版本兼容。

使用外部数据库

可以通过 Vijeo Citect 外部数据库中的工厂级存储和更新运行时数据。也可以使用 Vijeo Citect 将信息（如处方）从数据库发送到工厂中的 I/O 设备。

通过使用外部应用程序读取和写入相同的数据库记录，可以通过关系数据库与外部应用程序有效交互。

Vijeo Citect 支持两种类型的数据库：

- [dBASE 数据库](#)
- [SQL 数据库](#)

dBASE 数据库

dBASE 文件格式已成为数据存储的行业标准，并且 Vijeo Citect 支持标准 dBASE 格式。可以使用任何数据库编辑器（支持 dBASE III 文件）创建数据库，并读取和写入数据库记录。

要连接到数据库，必须定义 Vijeo Citect 设备。设备指定数据库的格式和位置，例如：

名称	Recipe
----	--------

格式	{Name,16}{Water,8}{Sugar,8}{Flour,8} {Salt,8}{Yeast,8}{Milk,8}
标题	名称
文件名	[DATA]:RECIPE.DBF
类型	DBASE_DEV
注释	Recipe 设备 (dBASE 文件)

SQL 数据库

SQL (结构化查询语言) 也已成为行业标准。SQL 是一种命令语言, 您可以通过该命令定义、操作、和控制 SQL 数据库和其它相关数据库中的数据。大多数数据库服务器支持 SQL。SQL 提供了对其它平台上 (如计算机、微型计算机和大型计算机) 的数据库服务器的直接访问权。

您可以使用 Vijeo Citect 设备函数设置每个 SQL 数据库的格式和位置。此外, 还可以使用 SQL 函数直接控制 SQL 事务。

必须定义用于指定数据库格式和位置的 Vijeo Citect 设备, 例如:

名称	Recipe
----	--------

在格式域中, 指定 SQL 数据库的域名, 例如:

格式	{Name,16}{Water,8}{Sugar,8}{Flour,8} {Salt,8}{Yeast,8}{Milk,8}
----	--

标题是 ODBC 连接的数据库连接字符串, 例如:

标题	DSN = ORACLEDATABASE
----	----------------------

在文件名域中输入数据库表的名称:

文件名	RECIPE
-----	--------

类型	SQL_DEV
注释	处方设备 (SQL)

另请参见

[使用结构化查询语言](#)

使用结构化查询语言

可以使用结构化查询语言 (SQL) 函数直接访问 SQL 数据库，而不是访问作为设备的数据库。直接使用数据库访问可以提供更高的灵活性。SQL 函数提供通过任何与 ODBC 兼容的数据库驱动程序访问 SQL 数据库（例如 MS Access、FoxPro、Paradox 等）。

另请参见

[连接到 SQL 数据库](#)

[执行 SQL 命令](#)

[使用事务](#)

[在 SQL 中表示日期和时间](#)

限制:

在 Vijeo Citect 中使用的 SQL 库有许多限制，这些限制已记录在知识库中（文章 Q4014）。请注意，此库的宗旨不是处理大型 SQL 数据事务。对于大容量数量，建议使用 Vijeo Citect 报表 (CitectHistorian)。

如果想要使用 Vijeo Citect 的内嵌 SQL 库，应考虑下列事项：

1. 使用 SQL 数据库的“简易”模型将事务信息仅限于记录到事务日志文件。
2. 使用固定的事务日志大小来限制日志文件的生长。
3. 定期备份数据库以将事务日志大小保持在可控范围之内。当 SQL 服务器完成备份数据库或其事务日志时，它自动截去事务日志的休停部分。如果事务日志文件已满，数据库事务将停止。
4. 保持工作表尽可能小。当 Vijeo Citect SQL 向数据表添加或追加新的数据行时，它首先使用 SELECT * 语句来获取列信息。如果表中记录成百上千，此操作肯定会影响到性能，如果这些 SQL Cicode 函数是阻塞函数，还可能导致锁死。
5. 使用 SQL 服务器触发器将记录从工作表移至永久表。在这种方法中，直接与 Citect 接口的工作表的大小不会无限制增长。
6. 切勿在关键的 Vijeo Citect 计算机上运行 SQL Cicode 函数。如果所有 SQL Cicode 函数都是阻塞函数并且以主要线程运行，则当 SQL 函数在 IO 或趋势上执行时，有可能发生 COM 号或趋势号缺少样例。与此类似，如果这些函数运行在报警或报表服务器上，报警可能缺少扫描或报表缺少触发器。

连接到 SQL 数据库

连接到 SQL 数据库系统之后，SQL 命令方可使用。使用 SQLConnect 函数进行访问。用户在使用其他 SQL 函数之前必须调用此函数。格式如下：

```
SQLConnect(sConnect);
```

sConnect 表示连接字符串，例如：

```
INT hSQL;
hSQL = SQLConnect("DSN=DBASE_FILES;DB=C:\ODBC\EMP;LCK=NONE;CS=ANSI");
! Connect to a dBASE Compatible Database File.
INT hSQL;
hSQL = SQLConnect("DSN=EXCEL_FILE;DB=C:\ODBC\EMP;FS=10");
! Connect to an Excel File.

INT hSQL;
hSQL = SQLConnect("DSN=ORACLE_TABLES;SRVR=X:ACCTS;UID=SCOTT;PWD=TIGER");
! Connect to an Oracle Database.
```

有关连接到 SQL 数据库的详细信息，请参见 SQL 服务器随附的文档。

另请参见

[执行 SQL 命令](#)

执行 SQL 命令

SQL 允许以非程序方式操作数据，您可以根据应该做些什么来指定一个操作，而不是根据应该如何去做。SQL 命令允许您：

- 在数据库中创建表格。
- 在表格中存储信息。
- 从数据库中精确选择所需的信息。
- 对数据和表格结构进行更改。
- 合并和计算数据。

SQLExec() 函数将执行您的 SQL 服务器支持的任何 SQL 命令。例如，要创建数据库表格，您可以执行 SQL "CREATE TABLE" 命令：

```
SQLExec(hSQL, "CREATE TABLE recipe ('Name' CHAR(16), 'Water' CHAR(8), 'Sugar' CHAR(8), 'Flour' CHAR(8), 'Salt' CHAR(8), 'Yeast' CHAR(8), 'Milk' CHAR(8));");
```

要将记录添加到数据库表格中，请使用 "INSERT INTO" 命令。此命令的语法如下：

```
INSERT INTO <filename> [(<col_name>, . . .)] VALUES (<expr>, . . .)
```

此命令将每个域中的值添加到表格中，例如：

```
SQLExec(hSQL, "INSERT INTO recipe VALUES ('Bread', '10', '5', '7', '1', '1', '2')");
```

列名称是可选的；然而，如果您忽略列（域）名称，值将以相同顺序被插入到域中。

要从 SQL 数据库读取数据，请使用 SQL "SELECT" 命令。您可以使用 "SELECT" 命令从表格中读取整个记录集或一行记录。随后您可以使用 SQLGetField() 函数读取每个域中的数据，例如：

```
SQLExec(hSQL, "SELECT * FROM recipe WHERE NAME = 'Bread'");
If SQLNext(hSQL) = 0 Then
    PLC_Water = SQLGetField(hSQL, "WATER");
    PLC_Sugar = SQLGetField(hSQL, "SUGAR");
    PLC_Flour = SQLGetField(hSQL, "FLOUR");
    PLC_Salt = SQLGetField(hSQL, "SALT");
    PLC_Yeast = SQLGetField(hSQL, "YEAST");
    PLC_Milk = SQLGetField(hSQL, "MILK");
END
```

要删除数据库记录，请使用 SQL "DELETE" 命令。此命令的语法如下：

```
DELETE FROM <filename> [WHERE <conditions>]
```

此命令从表格中删除值，例如：

```
SQLExec(hSQL, "DELETE FROM recipe WHERE NAME = 'Bread'");
```

[另请参见](#)
[使用事务](#)

使用事务

可以对较为复杂的数据库操作使用数据库事务。通过数据库事务，可以执行一系列 SQL 命令，然后将更改提交到数据库，或“回滚”（取消）更改，例如：

```
SQLBeginTran(hSQL); ! Begin the transaction
SQLExec(hSQL, "UPDATE recipe SET water = '12' WHERE NAME = 'Bread'");
SQLExec(hSQL, "UPDATE recipe SET milk = '1' WHERE NAME = 'Bread'");
IF . . . THEN
    SQLCommit(hSQL); ! Commit the transaction
ELSE
    SQLRollBack(hSQL);! Cancel the transaction
```

```
END
```

如果不能使用事务，请检查数据库驱动程序的 ODBC 兼容性级别。

另请参见
[在 SQL 中表示日期和时间](#)

在 SQL 中表示日期和时间

SQL 日期的表示方法取决于特定的数据库系统。对于 dBASE，通常在大括号中指定日期，例如 {02/18/95}。对于 Oracle，使用格式：to_date('02/18/95','MM/DD/YY')。其他 ODBC 驱动程序可能要求其他格式 - 常见 ODBC 格式是：'YYYY-MM-DD'。

注意：外部数据库中的日期参考应基于格列高利历，或者必须首先将数据库表格导出到文本文件中，然后才能在 Vijeo Citect 中使用。导出为文本文件的 Microsoft Access 数据库表格中的日期被存储为格列高利值。

数据库独立的日期-时间语法

针对数据库独立性，您可以为日期和时间使用下列语法：

```
[<format>'YYYY-MM-DD HH:MM:SS.FFFFFFF']
```

其中：

- **<format>**（左方括号后面的第一个字符）必须为以下字符之一：
- d - 日期
- t - 时间
- dt - 日期和时间

无论是否指定日期、时间或日期和时间，都必须提供完整的 26 字符串，例如：

```
[d'1995-02-18 00:00:00.000000']
```

有关 SQL 命令的详细信息，请参见 SQL 服务器随附的文档。

使用 ODBC 驱动程序

Vijeo Citect 支持开放数据库连接 (ODBC) 标准。很多数据库包制造商还为其软件提供 ODBC 数据库驱动程序。还有一些独立团体为各种数据库制造的 ODBC 数据库驱动程序，

如带有 DataDirect ODBC 包的 Intersolv Q+E。通常，数据库的任何 ODBC 驱动程序都能够正常工作。

另请参见

[安装 ODBC 驱动程序](#)

[关于 ODBC 驱动程序](#)

[设置 ODBC](#)

[获得正确的 ODBC 语法](#)

[ODBC 的编程风格](#)

[将 Vijeo Citect 用作 ODBC 服务器](#)

安装ODBC驱动程序

必须从 Windows 控制面板中安装并设置 ODBC 驱动程序。安装 ODBC 驱动程序：

1. 打开 Windows 控制面板。
2. 点击 **ODBC** 图标启动 ODBC 设置实用程序（如果控制面板中还没有 ODBC 图标，您可能需要安装该图标。参见随 ODBC 驱动程序提供的文档）。
3. 点击**添加**设置一个数据源。

注意：如果您的 ODBC 驱动程序未包括在“已安装 ODBC 驱动程序”列表中，请返回到 ODBC 设置实用程序并安装您的驱动程序（选择**驱动程序...**按钮）。

4. 从**添加数据源**对话框中，选择您的 ODBC 驱动程序。随即会显示“设置”对话框。
5. 输入您以前使用的驱动程序的数据源名称。例如，如果您曾将 SQL 用于 dBASEIII 数据库，则您的连接字符串就应该是 "DRV=QEDBF"。要避免在整个工程中更改连接字符串，请使用 **QEDBF** 的数据源名称。
6. 运行您的 Vijeo Citect 工程。

如果您的 ODBC 驱动程序具有有限功能，则某些 Cicode SQL 函数将不能运行。这种情况并不常遇到，在多数情况下，只会影响将事务用于 SQLBeginTran()、SQLCommit() 和 SQLRollBack() 函数的能力。如果使用的是 Intersolv Q+E ODBC 驱动程序，不可能会遇到任何功能丢失。

每次从使用一个 ODBC 兼容驱动程序转向使用其他驱动程序（例如，从 dBASE 文件转向 Access 数据库）时，可能需要在控制面板中更改数据源。点击 ODBC 图标并从可用数据源列表中选择。（有关详细信息，请参阅随驱动程序提供的文档。）

注意：

- 对于与 Cicode SQL 函数的充分兼容性，ODBC 驱动程序应提供最少数量的函数。例如，如果您的驱动程序不支持 ODBC 函数 SQLTransact，您就不能使用 Cicode 函数 SQLBeginTran()、SQLCommit() 和 SQLRollback()。
- 向下兼容您可能已在早期版本中创建的任何函数。Q+E 驱动程序现在兼容 ODBC，因此您需要将旧的 Q+E 数据库驱动程序升级到 Q+E ODBC 数据库驱动程序。

另请参见

[关于ODBC驱动程序](#)

关于 ODBC 驱动程序

Vijeo Citect 直接连接至 Microsoft Access ODBC 驱动程序，这允许应用程序访问 MDB（Microsoft Access 数据库）文件中存储的信息，而无需实际运行 Microsoft Access。（Microsoft Access 使用“Jet Engine”DLL 来访问 MDB 文件中存储的信息。）

ODBC 通常暗示多数情况下会使用 SQL 语句来处理数据。SQL 语句相当复杂和详细。要在 Cicode 中实现这些语句，通常必须将这些语句分成子字符串，以便不会超过 Cicode 变量的字符串长度最大值。

利用 Access，可以调用已在 Access 中定义的查询，以使 SQL 语句变得非常简单和直观。Access 表和查询可用于实现**关系和链接**，以便只对这些行（记录）进行**排序和选择**，并仅返回此时特别重要的那些列（字段）。

在 Access 中开展查询还有一个优点，就是可以在 Access 中查看生成的记录集，以测试记录集中包含预期的数据。这些查询可以合并 SQL 函数（如 BETWEEN 和 AND）。

Jet Engine 还可以调用 VBA Expression Service。这意味着，如果不需要以后迁移到非 Access 系统，则还可以使用许多非 ANSI 函数（在 SQL 语句和 Access 查询定义中）。请参考 Access 或 Excel 帮助系统中的《VBA 函数参考》（仅限于后面带有 (VBA)，适合于 SQL 环境，而且很可能可在 SQL 语句中发挥作用的那些函数）。

另请参见

[设置 ODBC](#)

设置 ODBC

要使用 ODBC，必须安装 Access ODBC 驱动程序。该驱动程序可以从 Microsoft 获取，与 Microsoft Office 一起提供。如果选择了适当的数据 Access/ODBC 选项，安装程序（例如 Microsoft Office 的安装程序）会将必要的驱动程序和 Jet Engine DLL 复制到相应的 Windows 目录。

将驱动程序安装到 PC 上之后，就可以从控制面板中选择 ODBC 图标，并为所需的 MDB 设置数据服务名称。该名称在连接字符串的 DSN= 部分使用。

Jet Engine DLL 非常大 (1 MB)，如果 Windows 虚拟内存管理器 (VMM) 交换它时内存不足，运行时（和应用程序）的执行速度则会受到影响。下次执行 SQL 时，当该 DLL 加载回内存时，将会出现短暂延迟。要强制让 VMM 将该 DLL 保留在内存中，可设计一个只具有一条记录和一个字段的简单虚拟表，并设置一个仅根据该虚拟表频繁（比如每隔 10-15 秒）调用 SELECT 查询的 Cicode 任务。这对 CPU 负载没有显著影响，并且可以将该 DLL 保留在内存中。

另请参见

[获得正确的 ODBC 语法](#)

获得正确的 ODBC 语法

SQL 的 ODBC 语法在某些方面与访问语法不同。获得正确语法并查看最终 Recordset 的很好方法是使用 **Microsoft Query** 中的查询设计器，然后将其中的 SQL 文本复制到 Cicode 中。因为 MS Query 使用 ODBC，在其中运行的任何语法都将在通过 ODBC 从 Cicode 中执行调用操作时运行。还可以使用 MS Query 确认 DSN 是否正确。

MS Query 倾向于创建比必要更复杂的 SQL 文本。尤其是它包含带有非必要文件名的路径，因为这是已经在 DSN 条目中定义的路径。对于硬编码文件路径，它被认为是不好的实践。MS Query 还倾向于在所有列（域）名前面添加表格名称前缀，以避免可能出现的不确定性。再次说明，这并非总是必要的，理想的情况是保持代码中的 SQL 文本尽量简洁。

查询设计器生成的 SQL 语句文本将被粘贴到“执行 SQL”窗口（位于 MS Query 的文件菜单下）中，将删除多余的文本并测试 SQL 语句，直至找到最简单的有效语法。需要按一定的规则保存 SQL 文本（如果需要）。可以在 Cicode 中大胆使用 SQL 语句最终版。

另请参见：

[ODBC 的编程风格](#)

ODBC 的编程风格

以下主题中的大多数示例代码不包含错误检查和报告功能：

- 通过 ODBC 从访问表中读取数据
- 通过 ODBC 向访问表中写入数据
- 通过 ODBC 删除访问表中的行
- 通过 ODBC 调用动作查询
- 使用 ODBC 进行参数查询

注意： 这些示例不包括为将保持尽可能简单而执行的错误检查。但是，无论是普通意义，还是接受的编程实务，都要求在 ODBC 代码中包含错误检查。

应考虑在访问查询定义中使用尽可能复杂的查询，在该定义中，复杂的查询易于设计，结果也易于查看。可以在调用查询时使用 WHERE 从句，以便在运行时精确选择所需行。如果表包含许多列（域），可以使用访问查询定义将任何特定调用限制在仅查看感兴趣的域。

这对于将 SQL 测试构建成字符串很有帮助。首先，ODBC 函数调用变得更简单。其次，可以将字符串传递到 TraceMsg() 从而使调试变得更简单。

请注意，Jet 引擎与 Vijeo Citect 在同一 PC 上运行，返回大型记录集的复杂查询可能会对 CPU 和内存资源产生不利影响。但是，通过对表、查询和关系尽心仔细设计，可以避免对 PC 资源的过度影响。

如果需要在**远程计算机**上执行查询，可以在报表服务器或事件服务器上设置代码。这在代码要由 PLC 中的事件触发时具有特殊的相关性。如果代码要由显示站上的用户触发，并且查询被认为过分集中在 CPU 上，则可以使用显示站设置要求代码或使用 Cicode Report()

函数调用报表的 PLC 位。另一种可能性是使用 Cicode MsgRPC() 函数从远程计算机上调用 Cicode 函数（如有必要，连同参数一起调用）。所有这些备选项都需要 Vijeo Citect 在远程计算机上运行。

另请参见

[比较 DDE 与 ODBC](#)

比较 DDE 与 ODBC

二者各有利弊。通常，DDE 适用于简单的请求，但如果 DDE 的限制过于严格，应该认真考虑 ODBC。

DDE 的优点

- 无需设置数据服务名 (DSN)；但需要为网络 DDE 设置 DDEShareName。
- 可以调用“访问 Macros 和功能”。

DDE 的缺点

- 无法直接读取使用超出最大 Cicode 字符串长度的行设置的记录。
- 行（记录）将返回列之间带 TAB 字符的字符串变量。要获得列（域）值，用户必须解析 Cicode 中的字符串。
- SQL 在 DDE 中无法执行动作（如插入、更新或删除）。
- DDE 客户端和服务端应用程序必须同时运行。

ODBC 的优点

- 可以不运行 MS Access。ODBC 在同一台 PC 上使用 JET Engine DLL。此优点在许多方面都有所体现，但如果管理不当可能会消耗过多 PC 资源。
- SQL 长语句可以划分成子字符串。
- SQLGetField 简化了从域（列）获取数据的过程。无需解析 Cicode 中的数据。
- 可以处理记录集中的大量域（列）。
- SQL 可以执行动作（如插入、更新或删除）。

ODBC 的缺点

- 需要设置数据服务名 (DSN)。

无法通过远程 PC 直接调用 JET Engine DLL，但可以使用报表或 MsgRPC() 通过远程计算机（该计算机必须正在运行 Vijeo Citect）运行 SQL 语句。

另请参见

[ODBC 兼容](#)

ODBC 兼容性

本节介绍您的数据库驱动器应该支持的所需和可选 ODBC 功能：

- [基本功能](#)
- [可选功能](#)

基本功能

只有数据库驱动程序支持下列 ODBC 功能时，Vijeo Citect SQL 设备和 Cicode 函数才能正常：

级别 0	级别 1
SQLAllocConnect	SQLColumns
SQLAllocEnv	SQLDriverConnect
SQLAllocStmt	SQLGetData
SQLBindCol	SQLGetFunctions
SQLColAttributes	SQLGetInfo
SQLDescribeCol	SQLGetTypeInfo
SQLDisconnect	SQLParamData
SQLError	SQLPutData
SQLExecDirect	SQLSetConnectOption
SQLExecute	SQLSetStmtOption
SQLFetch	
SQLFreeStmt	
SQLGetCursorName	
SQLNumResultCols	
SQLPrepare	
SQLRowCount	
SQLSetParam	

可选函数

Vijeo Citect SQL 设备和 Cicode 函数同样使用下表中列出的 ODBC 功能。当这些功能不是操作必不可少的功能，如果它们不在驱动程序中，则当 ODBC 驱动程序不支持所需的 ODBC 功能时将返回错误代码 307（SQL 数据库错误）。要获得与此错误关联的信息，请调用 Cicode 函数 `SQLErrMsg`。

级别 0

SQLTransact	如果驱动程序不支持此函数，那么 Cicode 函数 SQLBeginTran()、SQLCommit() 和 SQLRollBack() 将不受支持。
-------------	---

级别 1

SQLSpecial 列	如果此函数可用，Vijeo Citect 会使用它。这样将不会有功能缺失。
SQLTables	(对 Vijeo Citect 没有影响)

级别 2

SQLData 源	驱动程序不需要支持此函数。它是由 ODBC 提供的。
SQLExtended 提取	没有此函数，Vijeo Citect 将无法利用本地数据库的功能随机提取记录。
SQLSetScroll 选项	没有此函数，Vijeo Citect 将无法利用本地数据库的功能随机提取记录。
SQLMore 结果	(对 Vijeo Citect 没有影响)
SQLNativeSql	(对 Vijeo Citect 没有影响)
SQLProcedure 列	(对 Vijeo Citect 没有影响)

另请参见

[使用 Vijeo Citect 作为 ODBC 服务器](#)

将 Vijeo Citect 用作 ODBC 服务器

通过 ODBC 服务器支持，Vijeo Citect 可以用作 SQL 数据库服务器。这样，支持 ODBC 的第三方应用程序就可以直接从 Vijeo Citect 访问数据。这表示用户可以直接访问 Vijeo Citect 中的数据，而无需开发 Cicode 或报表来导出数据。

当前，Vijeo Citect ODBC 服务器允许访问变量标签。变量标签表名为“TAGS”，格式如下：

NAME	变量标签名称	只读
VALUE	当前运行时参数值	读/写

Vijeo Citect 只能在运行时用作数据库服务器。运行时通过 ODBC 使用标签仍可添加到 Vijeo Citect 许可点数。一旦动态点数大于许可点数，软件保护机制将关闭 Vijeo Citect 运行。因此，通过 ODBC 服务器访问标签时，了解已使用的点数非常重要。有关详细信息，请参见《安装与配置指南》中的“许可点数”。

设置 Vijeo Citect ODBC 服务器:

计算机中必须先安装 TCP/IP。

1. 选择**开始 | 设置 | 控制面板**。
2. 双击 ODBC 图标。
3. 单击**用户 DSN** 标签中的**添加**。

注意： 如果选择其它标签，则会看到 Vijeo Citect ODBC 驱动程序已自动安装。

4. 从列表中选择 **Citect 驱动程序** 并单击**完成**。
5. 在**数据源**字段中输入“**Citect**”。如果不希望使用此名称，则确保使用的名称是一个单词。
6. 在**主机**字段中输入计算机名。计算机名在控制面板的“网络”部分中指定。
7. 单击**确定**。

使用 MS Query (V2.00) 访问 Vijeo Citect ODBC 服务器:

所有适合 ODBC 的应用程序使用不同方法构建用于访问 Vijeo Citect 标签的查询。此处给出的对于使用 MS Query 的示例说明显示简单的实施方案。MS Excel 和 MS Access 使用相同的方法。

1. 确认计算机中安装了 MS Query。
2. 设置 Vijeo Citect ODBC 服务器。
3. 运行 Vijeo Citect。
4. 运行 MS Query。
5. 从**文件**菜单（在 MS Query 中）中选择**新建查询**。
6. 从“可用数据源”列表中选择 Vijeo Citect 数据源名称 (DSN)。单击**使用**按钮。
7. 选择**标签表**。单击**添加**按钮，然后单击**关闭**按钮。
8. 随即可运行查询从 Vijeo Citect 提取标签数据。查看标签数据最简单的方法是双击**名称**和**标签**。

使用 MS Query (V8.00) 访问 Vijeo Citect ODBC 服务器:

与版本 2.00 不同，版本 8.00 中不使用用户 DSN，而使用文件 DSN（缺省情况下存储在 Program Files\Common Files\ODBC\Data Source 文件夹中）。文件 DSN 不会存储在 Windows 寄存器中，它们是扩展名为 .DSN 的文本文件。连接到现有数据源时，只显示计算机中存储的可用文件 DSN。MS Query V8.00 不显示用户或系统 DSN。最简单的解决方案是创建指向用户 DSN 的文件 DSN。

如何创建指向用户 DSN 的文件 DSN:

1. 使用文本编辑器（例如，记事本），并创建包含以下两行的文件：

[ODBC]

DSN=<MyUsrDSN>

其中 <MyUserDSN> 是现有用户 DSN 的名称，该用户 DSN 是通过控制面板中的 ODBC 图标创建的。

2. 单击文件菜单中的**另存为**，并键入包含 .DSN 文件扩展名的名称。例如，"**Citect_File.dsn**" 是有效的名称。包含括号可确保 .DSN 文件扩展名正确添加。将其保存到上面列出的缺省文件 DSN 目录，随后该文件在 DSN 列表框中显示。
3. 从控制面板打开 ODBC 管理器，并确认能够看到新创建的 File.DSN。
4. 从控制面板打开 ODBC 管理器，并确认已创建名为 <MyUsrDSN> 的用户 DSN。例如：
 1. 选择 Citect 驱动程序，然后单击**完成**。
 2. 在**数据源**字段中键入 "**Citect**"（即 <MyUsrDSN>）。
 3. 在**主机**字段中输入 *计算机名*。

运行 MS Query 时，可以随即从列表中选择文件 DSN。

另请参见

[使用 ODBC 读取来自访问表的数据](#)

使用 ODBC 读取来自访问表的数据

您可以使用 SELECT 查询来读取来自“Access”表的数据，或调用“Access”查询。

如果表中所含的列多于所需的列，如果需要对数据进行排序，或如果需要关联多个表，则查询比表更加适用。所需的 Cicode 如下：

```
Function SQLTest
  INT hSQL, iResult;
  hSQL = SQLConnect("DSN=ODBCTest;UID=YourUID_C;PWD=YourPWD");
  IF hSQL <> -1 Then
    iResult = SQLExec(hSQL, "SELECT * FROM qryRecipes WHERE Recipe Between
'3000' And '6000'");
    IF iResult = 0 Then
      WHILE SQLNext(hSQL) = 0 DO
        TraceMsg(">" + SQLGetField(hSQL, "Recipe") + "<>"
+SQLGetField(hSQL, "Flour") + "<>" +SQLGetField(hSQL, "Water") + "<>"
+SQLGetField(hSQL, "Cocoa") + "<");
        END
        SQLDisconnect(hSQL);
      ELSE
        Message("SQL Error", SQLErrMsg, 48);
      END
    ELSE
      Message("SQL Error", SQLErrMsg, 48);
    END
  END
END
```

另请参见

[使用 ODBC 附加数据](#)

使用 ODBC 附加数据

要使用 ODBC 将数据附加到“访问”表中，可以使用 SQL INSERT 语句。

```
Function SQLInsert
  INT hSQL, iResult;
  hSQL = SQLConnect("DSN=ODBCTest;UID=YourUID;PWD=YourPWD");
  IF hSQL <> -1 Then
    iResult = SQLExec(hSQL, "INSERT INTO tblRecipes (Recipe, Flour, Water,
Cocoa) VALUES ('X1234', 2, 3, 4)");
    SQLDisconnect(hSQL);
  END
END
```

为避免必须处理 SQL 语句，可以使用标准的 Cicode 设备函数将记录附加到“访问”表。首先应配置 SQL 设备。如果该表具有多个不需要写入内容的字段，只需定义设备定义中所需的那些字段（这将使设备定义始终很简单，并会降低 DevWrite 指令的数目）。然后，可以使用 DevOpen、DevWrite 和 DevClose 将记录添加到表中。

Vijeo Citect 接受连续的 DevWrites，直到它们等于构造 SQL INSERT 语句时设备定义中的字段数目。DevWrites 包含的字段数据必须与设备定义的顺序相同。因为需要尽可能多的记录，所以应在 DevOpen 后面直接进行连续的 DevWrites，以使 DevClose 避免数据超出上下文范围。

另请参见

[使用 ODBC 编辑数据](#)

使用 ODBC 编辑数据

要编辑“访问”表中的数据，必须有唯一的（通常为主要的）键以识别要更改的行（记录）。这可以提供一个 WHERE 子句，仅用于 SQL UPDATE 中的该行。

要编辑数据，请以正常的方式读取数据，并保持对唯一键的追踪。稍后可以使用带有 WHERE 子句的 UPDATE 查询将任何更改过的值写入到同一行中。

```
Function SQLUpdate
  INT hSQL, iResult;
  hSQL = SQLConnect("DSN=ODBCTest;UID=YourUID;PWD=YourPWD");
  IF hSQL <> -1 Then
    iResult = SQLExec(hSQL, "UPDATE tblRecipes SET Flour = 20, Water =
30,Cocoa = 40 WHERE Recipe = 'X1234'");
    SQLDisconnect(hSQL);
  END
END
```


注意：在 ODBC/SQL 环境中，您无法编辑“当前记录”（因为不存在当前记录）。因此，您无法使用 `DevAppend` 或 `DevSetField` 添加或修改记录。

另请参见

[删除“访问”表中的行](#)

删除“访问”表中的行

`DELETE` 关键字与 `WHERE` 分句一起使用可以删除所需行（一个或多个）。如果 `WHERE` 分句不以主键为基础，可能会删除多个记录。

```
Function SQLDelete
    INT hSQL, iResult;
    hSQL = SQLConnect("DSN=ODBCTest;UID=YourUID_C;PWD=YourPWD");
    IF hSQL <> -1 Then
        iResult = SQLExec(hSQL, "DELETE FROM tblRecipes WHERE Recipe =
'X1234'");
        SQLDisconnect(hSQL);
    END
END
```

另请参见

[通过 ODBC 调用动作查询](#)

通过 ODBC 调用动作查询

无法使用下面这样的 `SELECT` 查询调用 Access 的 `ACTION` 查询：

```
"SELECT * FROM qdeDeleteRecipe"
```

要通过 ODBC 调用 Access 的 `ACTION`，请在 `SQLExec` 中使用 `Call` 语句：

```
"{Call qdeDeleteRecipe}"
```

该语句必须用花括号({curly})括起。

可以使用 `Call` 语句来调用 `SELECT` 查询，得到的 `Recordset` 可通过正常方式访问。

另请参见

[参数查询](#)

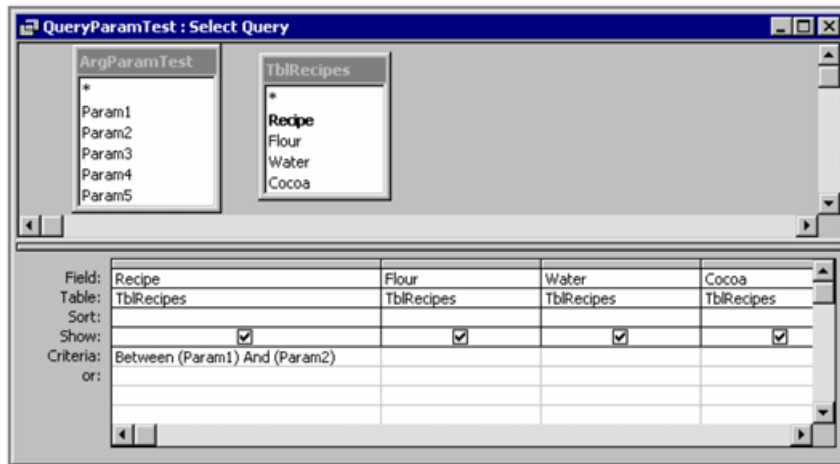
参数查询

许多 ODBC 服务器将接受 Call 语句中的参数，这样就可以在服务器上的查询定义中定义参数，并在运行时由 ODBC 客户机为其提供值。遗憾的是，Access Jet Engine 使用的是标准 Call 语句不支持的参数标记。此处概述的方法可作为一种解决办法。

对于需要参数的每个查询，设计一个与查询名称相同但前缀不同的参数表。例如，如果查询为 qryParamTest，则该表可称为 argParamTest。

该表应有五个字段，比如 Param1、Param2、Param3、Param4、Param5。

该表就添加到 qryParamTest 的访问查询定义中。然后，可以查询定义中的任何地方将这些字段名称用作参数。



可以编写一个简单 Cicode 函数来传递查询名称（不带前缀）和参数。该函数会在查询名称前插入“arg”，并从表中执行一个删除操作（以确认表是空的），然后执行一个插入操作，以使表包含相应字段中有所需参数的一条记录。然后，函数会在查询名称前加上“qry”前缀并调用该查询。

```
Function SQLCall(INT hSQL, STRING sQueryName, STRING sArg1 = " ", STRING sArg2 = " ",
    STRING sArg3 = " ", STRING sArg4 = " ", STRING sArg5 = " ")
    STRING sTable, sQuery;
    sTable = "arg" + sQueryName;
    sQuery = "qry" + sQueryName;
    SQLExec(hSQL, "DELETE FROM " + sTable);
    SQLExec(hSQL, "INSERT INTO " + sTable + " (Param1, Param2, Param3, Param4,
    Param5) VALUES ('" + sArg1 + "', '" + sArg2 + "', '" + sArg3 + "', '" + sArg4 +
    "', '" + sArg5 + "')");
    SQLExec(hSQL, "{Call " + sQuery + "}");
END
```

然后，通过 Cicode 调用函数就很简单了：

```
SQLCall(hSQL, "ParamTest", "2000", "4000");
```

如果在字段 Param1、Param2 等的 Access 表定义中“允许零长度”为“否”，则 SQLCall 的缺省参数必须是 SPACES。

该函数可用来调用许多不同的参数查询。

此解决办法的优势在于，即使在 Vijeo Citect 关闭之后，仍然可以从 Access 调用该查询，并且由于参数仍然存储在参数表中，因此可以在 Access 中查看最终的记录集。

另一种方法是设计查询来执行任何所需的联接、排序和字段选择，这会使用 WHERE 子句调用它们来选择所需的行（记录）。

另请参见

[Access 和 Cicode 日期/时间转换](#)

Access 和 Cicode 日期/时间转换

Access 和 Cicode 具有不同的日期/时间变量数据类型。在一个外部数据库里的所有数据参考应该以 Gregorian Calendar 为基础。

您可以采用三种方式在日期和时间之间进行转换：

- **转换为实型数** - 在 Cicode 和 Access 中，可以将实型数与数据/时间变量对等，如下所示：

```
AccessTime = 25568.66667 + (CicodeTime/86400);
CitectTime = 86400*(AccessTime - 25568.66667);
```

- **转换为字符串** - 可以使用每个环境中可用的标准转换函数在数据和/或时间与文本字符串之间来回转换。
- **使用 #Date/Time# SQL 语法** - Jet Engine 将转换用 # 标记括住的日期和时间字符串。此日期在 WHERE 子句中可能非常有用：

```
SELECT * FROM qryMyQuery WHERE 'Date' BETWEEN #3/20/96# AND #3/27/96#
```

在这种情况下使用美式日期格式，Jet Engine DLL 将忽略 Windows 控制面板中设置的本地日期和时间设置。

使用 Microsoft Excel 编辑 .dbf 表

Vijeo Citect 允许您编辑和保存 Vijeo Citect 中使用的 .dbf 文件（表），方法是在 Microsoft® Office Excel® 中打开它们。

使用 Microsoft Office Excel® 2007，您无法以 .dbf 的格式保存文件，即使您使用“文件”>“打开”命令进行打开和编辑也是如此。为了克服这一限制，Vijeo Citect 包括了一个针

对 Microsoft Excel 的插件，其名称为 ProjectDBFAddIn。在 Excel 中加载了此插件后，您将可以使用正确的格式浏览、打开、编辑并保存 Vijeo Citect .dbf 文件。

ProjectDBFAddIn 是一个包括在 Vijeo Citect 安装中的可选组件，可通过安装程序来访问。如果在安装过程中不选择安装插件，您可以在以后安装它，方法是切换至 Vijeo Citect 安装的 ProjectDBFAddIn 文件夹并运行该文件夹中的 setup.exe。

注意：只有当安装程序在目标计算机上检测到 Microsoft Excel 2003 或更高版本时，才能够安装 projectDBFAddIn。

兼容性

虽然此插件是专为 Microsoft Excel 2007 和更高版本设计的，但它能够与 Microsoft Excel 2003 兼容。

与 Microsoft Excel 集成

打开 Microsoft Excel 时，根据您使用的 Excel 版本，一个名为工程 DBF 插件的附加工具栏或附加选项卡可供您使用。

利用此工具栏上字段和按钮，您可以选择：

1. 到 master.dbf 的路径。
2. master.dbf 中的 Vijeo Citect 工程。
3. 属于选定工程的 .dbf 文件（表）。
4. 保存对 .dbf 文件（表）的更改。
5. 保存更改但不重新编制表的索引，或者是保存并重新编制索引。
6. 以其他名称保存 dbf 文件。
7. 打开任意位置中的 dbf 文件

但是，您也可以使用“文件”>“打开”命令打开 .dbf 文件。

受保护的表

此插件的安装将结合这些表的一个列表，这些表在缺省情况下是不会使用此插件界面打开的。此列表在名为 ExcludedProjectTables.xml 的文件中，可以在插件安装目录中找到。

这些表是 Vijeo Citect 维护的系统表，建议不要在外用 Microsoft Excel 编辑这些文件。要防止使用 Microsoft Excel 编辑其他表，您可以使用文本编辑器将它们名称添加到 ExcludedProjectTables 文件。您也可以使用文本编辑器或 XML 兼容浏览器查看这些表的缺省列表。

只读表

当从插件按钮中选择时，只读工程或包含只读表的工程将显示在这些表的列表中。如果您尝试打开一个只读表（.dbf 文件），将显示一个警告消息，提示您该表是只读的。您仍然能打开和编辑一个只读表，但不能保存它。

一个表可因下列原因而成为只读表：

- 工程是只读的。
- 该表在 Vijeo Citect 工程浏览器中打开。
- 尝试编辑此文件的用户没有读和/或写权限。

如果您尝试保存一个只读表，将显示一个提醒信息，提示您该表是只读的。

另请参见

[功能](#)

功能

Vijeo Citect 安装时，其 Excel 插件将自动生效。打开 Microsoft Excel 时，您将看到一个名为工程 DBF 插件的新工具栏，其中包含了下列字段和按钮：

1. Master.dbf 位置
2. SaveDBF 表
3. SCADA 工程
4. SCADA 表
5. 保存 DBF
6. 另存为
7. 打开 DBF
8. 仅保存/保存并重新编制索引

使用插件访问表：

1. 如果 master.dbf 位置字段内没有显示 Master.dbf 文件，其将显示输入 master.dbf 的路径。点击这个字段会显示一个对话框，可以在此输入位置。这将填充 SCADA 工程字段。
2. 点击 SCADA 工程字段的向下箭头，选择想要打开的存放 dbf 文件（表）的工程。这将填充 SCADA 表字段。

注意：任何列在 ExcludedProjectTables.xml 中的表都不会在此列表中显示。

3. 单击 SCADA 表字段的向下箭头选择表。表在 Microsoft Excel 中以新建一个同名工作表的方式被打开。如果表是只读的，Excel 会显示一个警告信息，提示你表是只读的。

另外，如果您要打开的 dbf 文件不是 Master.dbf 文件，或不是位于根 dbf 文件位置，请单击“打开 DBF”按钮并浏览至您要打开的 dbf 文件所在位置。

使用 Microsoft Excel 编辑表内容

在 Microsoft Excel 中 .dbf 格式的表显示为下列相应形式：

- 字段显示为列。
- 记录显示为行。

- 记录（数据）的字段显示为单元格。

你可以编辑表中的任何数据，每行数据显示为一行内的多个单元格。你可以整行删除或者添加一条记录。但是，你不能通过修改任何显示为列的字段的名字来改变表的结构、增加列或者删除列。如果改变了表结构，保存表时会显示提醒信息。

编辑单元格时，小心不要超过表里设置的最大字段长度。

在 Microsoft Excel 中保存表

1. 从工具栏的下拉控件中选择“保存并重新编制表索引”或“仅保存”。

如果您不对一个已编制索引的表重新编制索引，可以更快地保存该表，但您需要在完成编辑后手动重新编制表索引。要手动重新编制索引，请从 Vijeo Citect 工程编辑器的“文件”菜单中选择“打包”。

2. 单击“保存 DBF”表按钮。

保存表时，插件会在保存之前执行下列检查：

1. 校验文件的类型是否是“.dbf”。
2. 检查表结构是否被改变。有关更改表结构的限制，请参见“编辑 Microsoft Excel 表的内容”。
3. 根据 Microsoft Excel 工作表中的数据更新现存表。
4. 如果已更新的表是一个已编制索引的表，请重新编制表索引（如果选中“保存并重新编制索引”）。

如果 Microsoft Excel 电子表格中被编辑表格的检查失败，系统将会显示一个警告信息；除非更正了导致检查失败的问题，否则当前磁盘的表格不会受到影响。

您也可以使用其他名称来保存 dbf 文件，方法是单击工具栏中的“另存为”按钮并输入文件的名称。这种保存方法还包括保存 dbf 文件时是否重新编制表索引的选项。

注意：如果使用 Microsoft Excel 的“文件”>“保存”命令保存 .dbf 格式的表格，系统将显示一条的提醒信息，告知文件不能采用当前格式保存，应使用 Microsoft Excel 工具栏中“保存 DBF”按钮。

Chapter: 32 使用精灵和超级精灵

通常，图形画面中的每个图形对象都会单独配置。通过精灵，可以将几个相关对象组合到一个组中，然后将该组存储到精灵库（类似于符号库）中。然后精灵可用作单个对象（粘贴、移动、调整大小等）和一起配置的元素。

所有类型的图形对象及其配置数据可以存储在精灵中。例如，可以为启动/停止控制器（通过启动按钮、停止按钮和指示灯）定义精灵，并对使用该类型控制器的所有设备（泵、传送带等）使用同一个精灵。使用此精灵时，只需指定对泵或传送带唯一的信息（即变量标签）。

Vijeo Citect 有两种类型的精灵：

- **精灵** - 关联对象的集合，在配置系统时添加到图形页面。图形页面中可以添加任意数量的精灵（例如，在同一页面中可以添加多个泵）。
- **超级精灵** - 动态页面（通常是弹出式），页面在运行时系统中显示时将信息传递到其中。可以将超级精灵用于弹出类型控制器，以控制过程或单个现场设备。

还可以使用超级精灵和精灵的组合以使用它们的共同特征。大多数超级精灵是通过附加到精灵实现的。

Vijeo Citect 已包含可在 Vijeo Citect 系统中使用的精灵和超级精灵库，用户也可以定义各自的库。可以为复杂实体（如回路控制器、自定义控件和指示组合）构建一个精灵或超级精灵。

注意：

- 如果在工程中使用精灵或超级精灵后对其进行了修改，则整个工程中所有出现精灵或超级精灵的地方就会被自动更新（超级精灵环境变量除外）。
- 如果工程在后台运行时修改精灵，则必须执行更新页面才能看到运行时工程的变化。如果进行修改时正在显示包含该精灵的运行时页面，则只有退出并重新显示它时才会被更新。

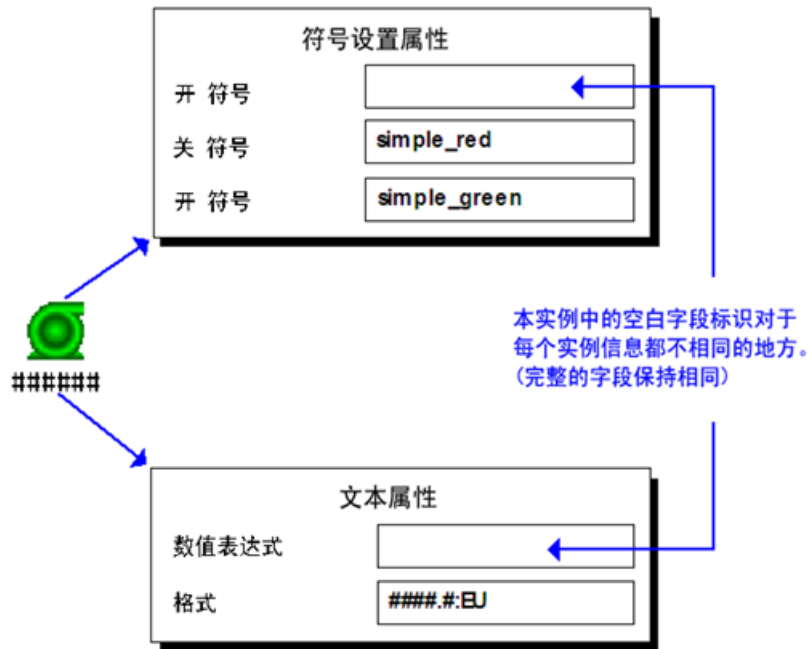
另请参见

[了解精灵](#)

[使用超级精灵](#)

了解精灵

精灵的工作方式为将普通信息替换到对象组中的每一个相关对象。例如，显示泵及其速度的典型配置使用两个对象：(1) 显示速度的文本对象，以及 (2) 表示泵状态的符号对象（显示多种符号）：



如果不用精灵执行上述配置，则必须对画面中的每一个文本和符号实例分别进行配置。这说明一些常用的对象组合在每一个实例中通常会有相同的配置。精灵的概念使得插入特定的信息之前，预备性地进行了共通部分的配置。

精灵的作用在于只对对象进行一次定义。每次将精灵放在画面上时，只需指定替换信息。

- [另请参见](#)
- [创建精灵](#)
- [打开精灵](#)
- [保存精灵](#)
- [为精灵定义替换](#)
- [使用精灵](#)

创建精灵

创建新精灵类似于创建画面，它使用图形对象但不使用背景。通常，可以使用“图形编辑器”创建新精灵，添加对象，定义精灵替代并将精灵保存到精灵库中。

创建新精灵：

1. 从文件菜单中选择新建。
2. 单击精灵按钮。
3. 现在即可以创建精灵对象（定义替换文本）。

请参见 [为精灵定义替换](#)

另请参见
[打开精灵](#)
[保存精灵](#)

打开精灵

您可以打开现有精灵以使用它进行工作。

要打开现有精灵，请执行以下操作:

1. 单击**打开**或选择**文件 | 打开**。
2. 选择**精灵**标签。
3. 选择存放精灵的**工程**和**库**。
4. 选择**精灵**。
5. 单击**确定**。

要从工程中删除精灵，请选择相应的精灵名称，然后点击**删除**。

如果在工程中使用了精灵或超级精灵后对其进行了修改，那么这个精灵或超级精灵在整个工程中所有出现的地方就会被自动更新（超级精灵环境变量除外）。

如果工程在后台运行时修改了一个精灵，那么必须执行更新画面才能看到运行的工程的变化。如果做修改时一个包含了这个精灵的运行画面正在显示，那么只有退出并重新显示它时才会被更新。

另请参见
[保存精灵](#)
[打开精灵](#)

保存精灵

如何保存当前精灵:

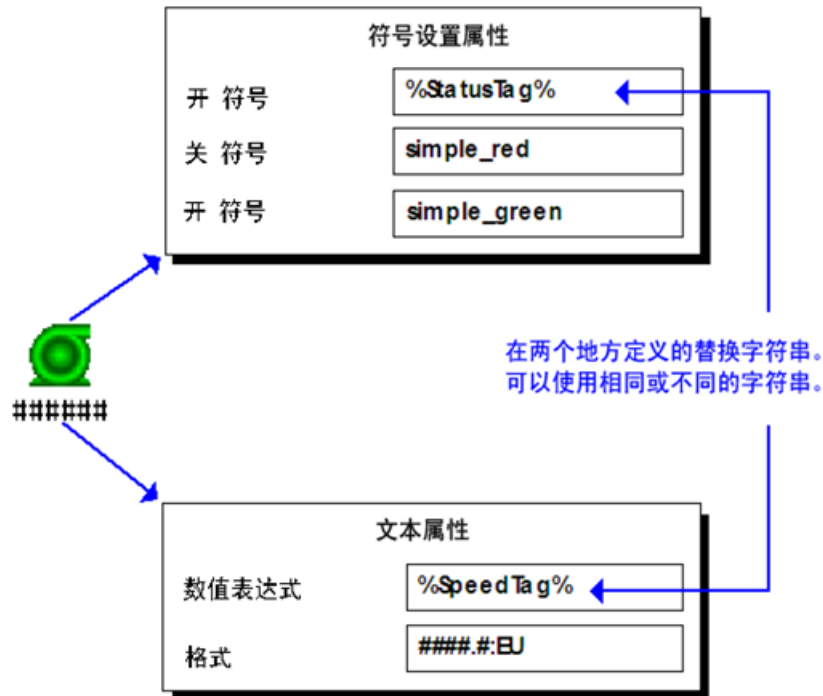
1. 单击**保存**工具，或选择**文件 | 保存**。
2. 选择从中存储精灵的**工程**和**库**。
3. 在**精灵**中输入精灵的名称。
4. 单击**确定**。（要为精灵创建新库，请单击**新建**。）

另请参见
[为精灵定义替换](#)

为精灵定义替换

要定义精灵，需将替换字符串用于特定于每一实例的对象的属性。可以将替代字符串用于任何对象（位于对象组中）中的任何文本属性。要将一段文本指定为替换文本，用百分号(%) 字符将该字符串括起来。

例如，要创建标准精灵，您可以使用两个替换字符串 - 一个用于状态标签变量，一个用于速度标签变量：



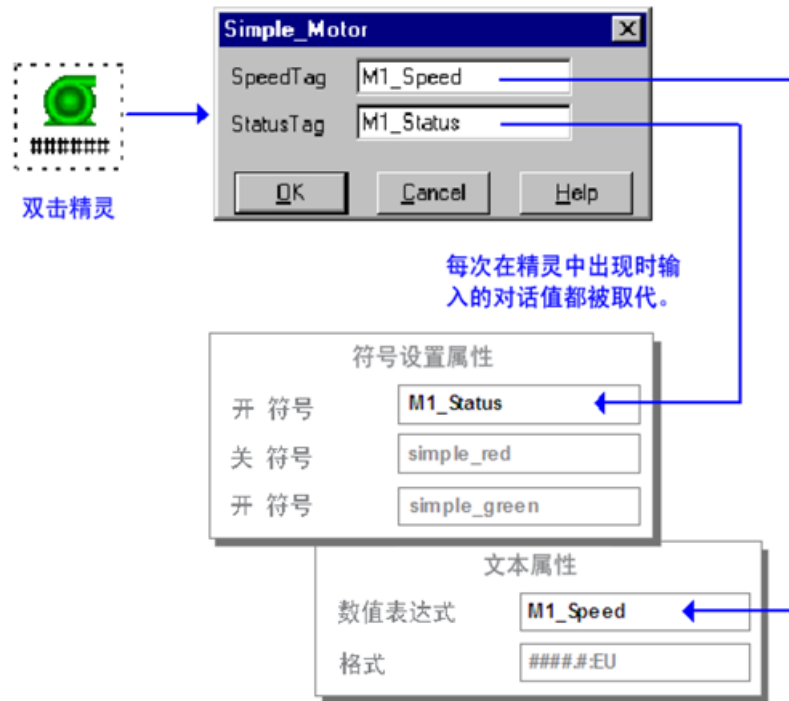
注意：并非只能使用标签变量作为替换字符串。任何表达式都能被替换，如常数或标签。只有接受文本的域可以使用精灵标签替换。还可以使用 IFDEF 函数为不位于当前工程中的变量定义替换。

另请参见
[使用精灵](#)
[在模板中使用精灵替换](#)

使用精灵

创建、定义和保存精灵后，可以在任何图形页面中使用它。要使用精灵，必须用粘贴精灵工具将其粘贴到页面。粘贴精灵之后，可通过双击图像对其进行配置。

例如，每次使用上述精灵时，只需在一个对话框中输入两个参数值：一个用于速度变量标签 (%SpeedTag%)，一个用于状态变量标签 (%StatusTag%)，而不是输入组中每个对象的属性。



双击粘贴的精灵会显示精灵属性。要显示精灵中各个对象的属性，请按下 **Control (CTRL)** 键并双击特定对象。但是，如果指向精灵的链接被保留，则这些属性中的大多数属性将为只读。

上例是精灵的简单使用，它只包含两个对象和两个替换字符串。可以通过对象任何**文本属性**的替换字符串定义使用很多对象的精灵。

注意：如果使用结构化标签，则可以在标签内使用替换字符串构建更复杂的精灵。请参见[通过精灵和超级精灵使用结构化的标签名称](#)。

如何将精灵粘贴到图形页面：

1. 单击工具框中的“粘贴精灵”工具，或者选择**编辑 | 粘贴精灵**。
2. 从库列表中选择包含精灵的库。
3. 从“粘贴精灵”对话框的“精灵”列表选择一个精灵缩略图。
4. 双击缩略图或单击**确定**。

另请参见
[“粘贴精灵”对话框](#)

粘贴精灵对话框

使用“粘贴精灵”对话框可向图形画面（或模板）中添加精灵。

精灵

工程中的精灵列表，显示粘贴的超级精灵。

要添加某个精灵，可使用滚动条定位到该精灵的缩略图，然后选定该精灵并单击**确定**按钮（或双击缩略图）。

注意：要编辑精灵，可将其选定后单击**编辑**。要生成新精灵，请单击**新建**。

库

存放精灵的库。

超级精灵

如果将选定的精灵粘贴到一个超级精灵，那么就会在此显示该超级精灵的缩略图；否则此字段为空。

精灵属性

精灵对话框显示您为精灵输入的替换字符串。您在表中看到的替换标签是在精灵中定义的。输入到标签旁边的值将被替换到精灵中（如果附加了超级精灵，则可能是超级精灵）。

注意：要在精灵显示单独对象的属性（而不是精灵的属性），按下**控制 (CTRL)** 键并双击对象。然而，如果指向精灵的链接被保留，这些属性中的大多数将是只读属性。

另请参见
[了解精灵](#)

在模板中使用精灵替换

可以通过向模板添加对象，并为每个对象的相关属性使用替换字符串 (%) 来创建与精灵具有相同特性的自定义页面模板。（如果任何属性有缺省参数值，则可以将缺省参数值添加到本机对象。）

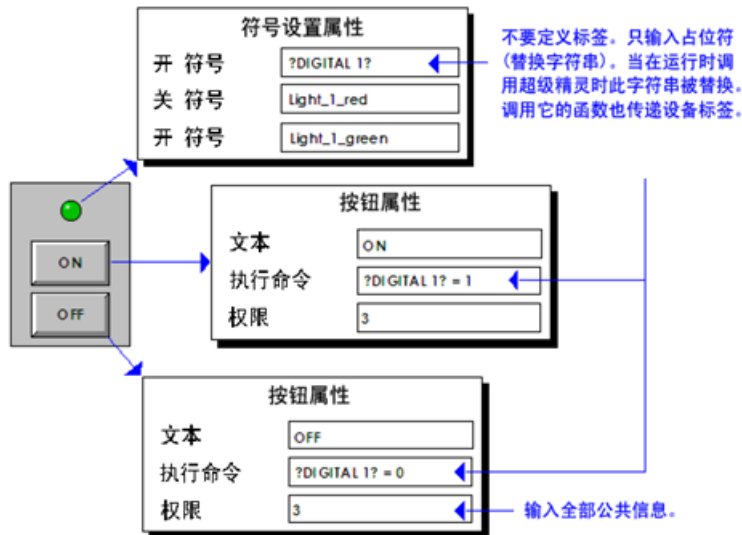
其后基于模板创建新页面时，会出现一个对话框提示输入在模板中使用的所有替换字符串的参数值。趋势和 SPC 模板就是这样创建的。

使用超级精灵

单页面（弹出式控制器、回路调节页面等）经常用于控制和监视设备。超级精灵在有很多相同类型的设备时非常有效，因为通过它可以多次重新使用设备，而无需对每个设备重新配置。配置常用信息一次，特定于设备的信息就会在运行时传递给超级精灵。

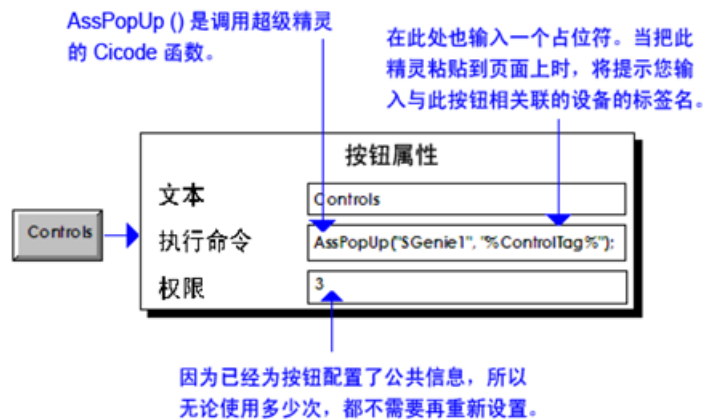
例如，可能需要使用超级精灵配置一个弹出式页面，用于控制功能相同的电泵。配置此控制器的最佳方法是：

在图形编辑器中，选择文件 | 新建超级精灵，绘制控制器并填充关联的属性表单，如下所示：



使用叹号 (!) 前缀将其保存到超级精灵库中。这样可使页面隐藏在配置环境中（只有附加到精灵控制器时才可见）。

选择文件 | 新建精灵，然后绘制用户将在运行时单击的按钮，以显示弹出式控制器。此按钮称为精灵控制器。它将调用超级精灵 Cicode 函数，执行替换并显示弹出式菜单。



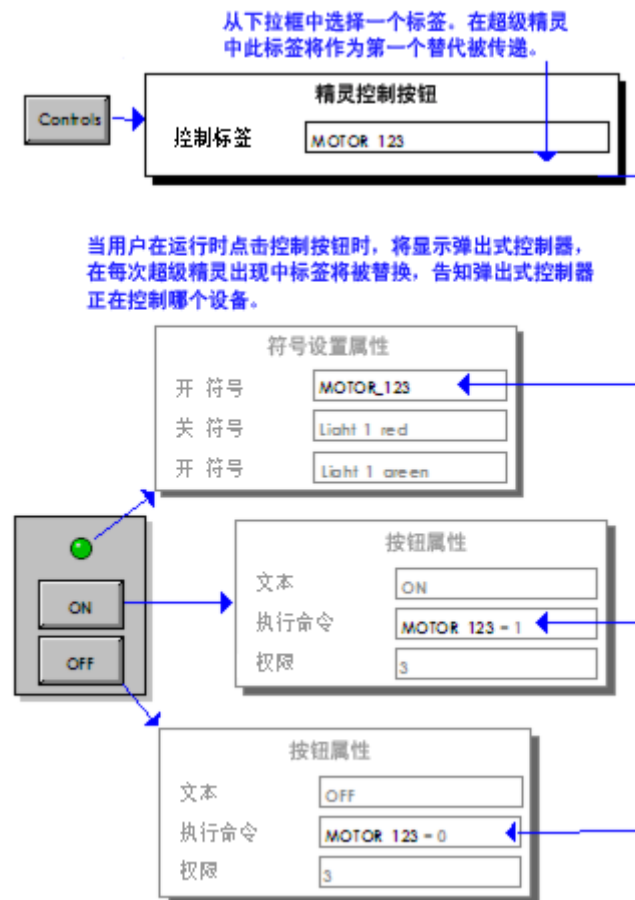
由于超级精灵函数是从精灵控制器调用的，因此只需对其配置一次。

将精灵保存到精灵库。与精灵库一样，超级精灵库也是全局的，并可以在 Vijeo Citect 工程之间使用。

精灵处于打开状态时，选择**编辑 | 附加超级精灵**，然后选择刚创建的超级精灵。从现在开始，粘贴此精灵将调用新的超级精灵。

将每个超级精灵附加到精灵，可帮助您将超级精灵按顺序存储在精灵库中。这样便于维护和粘贴到工程中。超级精灵可以附加到多个精灵控制器。

将精灵粘贴到希望用户使用弹出式控制器的位置。选择**编辑 | 粘贴精灵**，浏览刚创建的精灵，然后选择该精灵。系统会在工程中为超级精灵自动创建新的页面。



如果不使用精灵和超级精灵实现上述状况，则必须手动为应用程序中的每个泵配置单独的页面，以及调用每个页面的单独按钮。使用超级精灵，只能手动配置一个页面。剩余的可自动创建。

注意： 为了避免编译错误，超级精灵中使用的所有变量标签必须在变量标签数据库中定义。也可以使用报警标签（允许用户使用报警标签属性）。

注意：超级精灵比其他编程方法（如数组）需要更多的系统开销。即使是数百个变量，数组也可以执行得很好，而超级精灵即使只有五十个变量也会表现出性能下降，这取决于所用的变量类型。因此，如果应用程序要求良好的性能，您应限制超级精灵的变量数目，并执行完整的测试以确定最终性能。

在运行时通过超级精灵使用标签可增加动态许可点数。达到点数限制后调用的超级精灵会返回 #COM。有关详细信息，请参见《安装与配置指南》中的“许可点数”。

实现超级精灵不一定使用精灵控制器。请参见 [不通过精灵使用超级精灵](#)。

另请参见

[为超级精灵定义替换](#)

[不通过精灵使用超级精灵](#)

[通过超级精灵使用常量和数组](#)

[嵌套超级精灵](#)

[超级精灵区域](#)

[超级精灵环境变量](#)

[通过精灵使用结构化标签](#)

为超级精灵定义替换

超级精灵替换比精灵替换更加严格和复杂。最重要的是，只能在接受标签、命令和表达式的对象的属性中使用超级精灵替换。（还可以在对象触击、键盘命令、工具提示、画面键盘命令的日志消息中使用超级精灵替换，或将其用作趋势对象和颜色填充的一部分。）不能在报表、报警、趋势或背景 Cicode 函数中使用超级精灵语法。

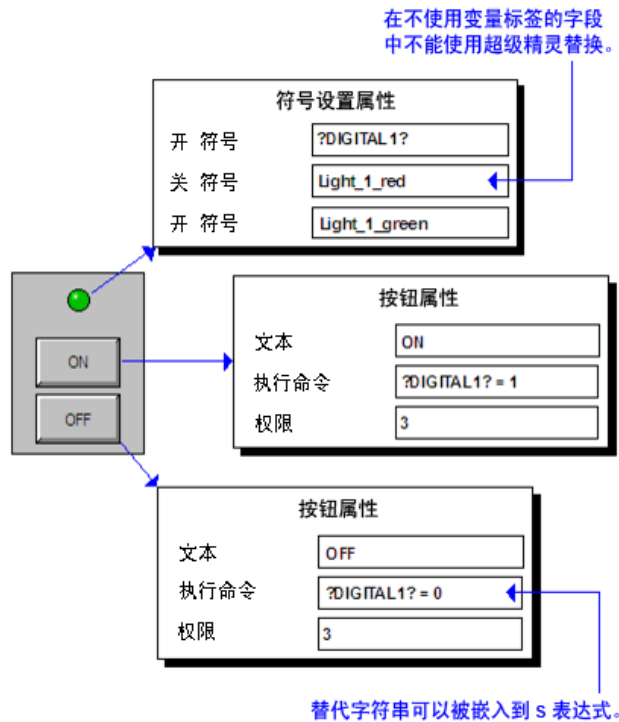
要将标签标记为替换字符串，将此标签包括在问号 (?) 之间，其格式如下：

```
?<Data Type> <Substitution String Number>?
```

其中：

- **Data Type** 除了 STRING 变量类型的情况外是可选的，必须明确指定。如果 Data_Type 被保留空白，Vijeo Citect 会在运行时正确地指派所有变量类型，除了 STRING 类型外。
- **Substitution String Number** 确定哪个标签变量（1 至 256）会在超级精灵显示时被替换（使用超级精灵函数）。如果在超级精灵中使用多个替换文本，您的数字应是连续的。这使得超级精灵函数更易于使用。

例如，要为弹出式控制器定义替换，需将替换字符串用于标签变量，如下所示：



注意： 应将此超级精灵保存为名为 **SGenie1** 的画面（而不是超级精灵），以便在没有精灵控制器的情况下使用此超级精灵。

如果不指定数据类型，缺省类型为 TYPELESS。Typeless 替换允许传递 BYTE、BCD、DIGITAL、INT、UINT、LONG、LONGBCD 或 REAL 类型的标签，但不允许传递 STRING 类型的标签。在创建无类型替换时，Vijeo Citect 将自动尝试在运行时将替换的“数据”转换为正确类型。

例如，上图中使用 **?Digital 1?** 作为替换字符串。运行时，如果传递被声明为 INT 的变量，将会出现硬件提醒信息。如果使用 **?1?** 取代之，那么在运行时就可以传递一个除 STRING 外任何类型的变量。

注意： 因为 typeless 替换提供了更大灵活性，您可能希望使用该替换，但应该注意的是，使用它后错误将更难查找。

另请参见
[不通过精灵使用超级精灵](#)

不通过精灵使用超级精灵

实现超级精灵不一定要附加到精灵。可以将未附加的超级精灵保存为普通的 Vijeo Citect 页面。这种方法的优点在于不必定义控制精灵，而缺点在于不能使用“粘贴精灵”工具来放置它。

如果通过此方法配置超级精灵并通过 ! 前缀命名页面来将其隐藏，则必须从图形编辑器的选项菜单中选择 **列出系统页面** 来编辑页面。

Vijeo Citect 提供的所有超级精灵都附加到精灵（作为超级精灵的控件）。

如何创建新的超级精灵：

1. 单击 **新建** 工具，或选择 **文件 | 新建**。
2. 单击 **超级精灵**。
3. 此时可以创建超级精灵 **页面**（定义替换字符串）。

注意：对于要在“粘贴精灵”对话框中显示的超级精灵，可创建精灵用作精灵控制器并将超级精灵附加到其中。超级精灵名称的前八个字符对于每个超级精灵必须唯一。

如何打开现有的超级精灵：

1. 单击 **打开** 工具，或选择 **文件 | 打开**。
2. 选择 **超级精灵** 标签。
3. 选择存储精灵的 **工程** 和 **库**。
4. 选择 **超级精灵**，然后单击 **确定**。

注意：要从工程中删除超级精灵，请选择超级精灵名称，然后单击 **删除** 按钮。

如何保存当前的超级精灵：

1. 单击“**保存**”工具，或选择“**文件**”|“**保存**”。
2. 选择从中存储超级精灵的 **工程** 和 **库**。
3. 在 **超级精灵** 字段中输入超级精灵的名称，然后单击 **确定**。

另请参见

[通过超级精灵使用常量和数组](#)

通过超级精灵使用常量和数组

可以通过超级精灵使用 [常量](#) 和 [数组](#)。

另请参见

[创建精灵控制器](#)

常量

向超级精灵传递常量时，这个常量组合只能是一个可以输入普通的 Cicode 变量标签的地方—如键盘命令，图符地址字段等。可以支持所有常量类型：STRING、INTEGER、DIGITAL、REAL 和 LONG。

在传递常量时需要通过在 Ass 函数中的参数两边加单引号对其格式化。例如，在向超级精灵传递常量数 **1.2345** 时，应该这样调用 Ass 函数：

```
Ass(hWin, nArg, "'1.2345'");
```

在传递一个变量标签时，不必使用单引号。例如，在向超级精灵传递变量标签 **Tag1** 时，可以如下调用 Ass 函数：

```
Ass(hWin, nArg, "TAG1");
```

另请参见
[创建精灵控制器](#)

数组

超级精灵可以接受数组元素或者整个数组作为替换。传递数组元素方法简明，是通过参照元素来实现的，其方法如下：

```
AssPopUp("MyPopUp", "DigArray[42]");
```

向一个超级精灵传递整个数组时，只使用数组的名称。例如：

```
AssPopUp("MyPopUp", "DigArray");
```

要将整个数组传递给超级精灵，应将超级精灵配置为接受数组，而不是接受单个值。超级精灵替换字符串使用下列语法：

```
?<Data Type>[<array size>] <Substitution String Number>? [<element>]
```

超级精灵只支持 DIGITAL、INT、REAL 和 LONG 数据类型的数组。

注意： <array size> 是可选的，未定义时其缺省为 2048 个数字，128 个整型或者 64 个实型元素。它用来核对数组的范围 – 因此如果向一个超级精灵传递了一个比预期小的数组，范围之外的值就会被缺省为 0（或者一个空字符），而不会产生 Cicode 错误。

例如，要在第一个替换变量标签中显示第三个元素（数字量数组），应使用以下语法：

表达式	?DIGITAL[] 1? [3]
-----	-------------------

而下面的语句用于测试向超级精灵传递一个预期大小的数组：

表达式	?DIGITAL[4] 1? [3]
-----	--------------------

另请参见
[创建精灵控制器](#)

创建精灵控制器

创建精灵控制器：

1. 保存超级精灵（应将超级精灵的名称限定为 8 个字符）。
2. 创建一个精灵，使其用超级精灵函数来显示超级精灵。
3. 选择**编辑 | 附着超级精灵**。
4. 点击**添加**。随即会显示“选择超级精灵”对话框。
5. 选择在第 1 步中保存的超级精灵，并将超级精灵添加到此精灵的列表中，然后单击**确定**。
6. 保存精灵。超级精灵会出现在“粘贴精灵”对话框中。

从精灵库中将超级精灵（控制器）粘贴到画面：

1. 点击**粘贴精灵**工具，或选择**编辑 | 粘贴精灵**。
2. 从“粘贴精灵”对话框中的**库**列表选择一个库。
3. 从**精灵**列表选择一个精灵略图。绑定了超级精灵的缩略图即会显示在**超级精灵**框中。
4. 双击缩略图或单击**确定**。

注意： 此步骤添加一个绑定了超级精灵的精灵（到画面）。精灵是这个超级精灵的控制器。当操作员在运行时间系统中选择精灵时，超级精灵就会显示。

另请参见
[绑定超级精灵对话框](#)

“绑定超级精灵”对话框

可以使用“连接超级精灵”对话框将超级精灵绑定至当前精灵。

绑定超级精灵

绑定到当前精灵的超级精灵列表。

绑定新的超级精灵：

1. 点击**添加**。
2. 使用“选择超级精灵”对话框以选择要绑定的超级精灵。
3. 点击**确定**以保存更改，或点击**取消**。

分离超级精灵：

1. 点击**删除**。系统不会要求您确认是否要删除绑定。
2. 点击**确定**以保存更改，或点击**取消**。

另请参见
[“选择超级精灵”对话框](#)

“选择超级精灵”对话框

此“选择超级精灵”用于选择一个超级精灵绑定到现在的精灵上。

超级精灵

工程中一个超级精灵的表。

要选择一个超级精灵，请使用滚动条查找超级精灵的微缩图，然后选择这个超级精灵并单击 **OK** (或者双击微缩图)。

注意：要编辑超级精灵，选择它，并单击**编辑**。要创建新的超级精灵，点击**新建**。

库

存放超级精灵的库。

嵌套超级精灵

Vijeo Citect 允许用户嵌套超级精灵。嵌套是指一个超级精灵嵌入另一个。要使嵌套正常工作，嵌入的精灵控制器（用于嵌入的超级精灵）必须使用 `AssChain` 函数，而不是 `Ass` 函数。

另请参见
[超级精灵区域](#)

超级精灵区域

显示超级精灵时，该超级精灵的区域是从它的父区域继承的。例如，如果父页面在区域 1，那么您要显示的超级精灵同样也在区域 1。这使得您可以从不同区域的不同页面调用同一个超级精灵页面。

可以通过将超级精灵定义成拥有一个特定区域来取消继承的区域。超级精灵的每一个实例都有同一个区域，而不管它的父区域来自哪个区域。只有在区域为空白时超级精灵才会继承区域。

另请参见

[超级精灵环境变量](#)

超级精灵环境变量

定义超级精灵时，实际上就是在创建与页面模板类似的超级精灵模板。当精灵控制器调用超级精灵时，此模板被用来创建一个新的超级精灵页面。此时，所有与模板一起保存的环境变量都将被复制到超级精灵页面上。不过，如果这之后模板的环境变量有了变化，超级精灵页面的环境变量仍将保持不变。

要使用对模板进行的更改更新超级精灵画面环境变量，必须找到并删除超级精灵页面（请记住，该页面可能带有一个感叹号 (!) 前缀），然后使用精灵控制器再次调用这个超级精灵。这样就可以创建一个拥有更新过的环境变量的新超级精灵页面。



警告

不希望的设备操作

如果在修改了模板环境变量之后没有按照过程从超级精灵模板中更新超级精灵画面，则存在一个潜在问题，即画面可能显示过时的环境变量（可能导致系统以不可预知的方式响应）。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

通过精灵和超级精灵使用结构化标签名称

使用结构化标签名称可在使用精灵和超级精灵时提供更强大的功能，因此可使用结构化标签约定。大多数精灵引用同一个物理设备，因此对设备中的每个元素使用类似的标签名称可以减少工程配置。

另请参见

[通过精灵使用结构化标签](#)

[通过超级精灵使用结构化标签](#)

通过精灵使用结构化标签

定义精灵时，可以向精灵属性添加前缀或后缀，以便在使用精灵时生成完整标签。例如，如果将精灵属性定义为 %tag%_PV，然后将 DEV1 用于标签，则精灵将生成完整的标签 DEV1_PV。

可以在精灵属性的开头（前缀）或结尾（后缀）添加额外的信息，或者在同一个精灵属性中同时使用前缀和后缀。例如，如果已定义了包含三个条形图的回路控制器（使用长方形的填充属性创建）来显示标签 DEV1_PV、DEV1_SP 和 DEV1_OP，则可以按以下方法配置精灵：

每个长方形都有单独的精灵标签：

级别表达式	%PV_Tag%
级别表达式	%SP_Tag%
级别表达式	%OP_Tag%

通过“精灵”对话框配置精灵时，必须输入三个单独的标签：DEV1_PV、DEV1_SP 和 DEV1_OP。但如果使用结构化标签，则可以按以下方式配置长方形：

级别表达式	%TAG%_PV
级别表达式	%TAG%_SP
级别表达式	%TAG%_OP

在这种情况下，只需输入一个标签 (DEV1) 即可生成六个对象。精灵将 DEV1 与 _PV、_SP 或 _OP 自动连接，这取决于标签的替换位置。除了可以减少配置时间，还可以使精灵更容易维护。

注意：上例说明了精灵的功能。精灵中的对象越复杂，数量越多，使用结构化标签的优点就越突出。也可以通过使用精灵属性的多个变量来创建复杂精灵。例如，%Area%_TIC_%Occ%_PV 或精灵变量前缀、后缀和数字的任意组合。

另请参见

[通过超级精灵使用结构化标签](#)

通过超级精灵使用结构化标签

超级精灵不支持超级精灵变量标签与其他信息的直接连接（像象精灵那样）。例如，?INT 1?_PV 是无效的，会产生编译错误。但可以使用 Cicode 表达式来连接标签。必须对每个真正的标签使用唯一的超级精灵变量，并通过 Ass Cicode 函数连接标签。例如，如果已定义了包含三个条形图的回路控制器（使用长方形的填充属性创建）来显示标签 DEV1_PV、DEV1_SP 和 DEV1_OP，则可以按以下方法配置超级精灵：

每个长方形都有单独的精灵标签：

级别表达式	?INT 1?
级别表达式	?INT 2?
级别表达式	?INT 3?

如果不使用结构化标签，则可以按以下方式对上述超级精灵调用 Ass 函数：

```
AssPage("PageName", "DEV1_PV", "DEV1_SP", "DEV1_OP");
```

要对超级精灵连接信息，还可以自行编写 Cicode 函数，如下所示：

```
FUNCTION
AssMine(STRING sPage, STRING sTag)
    AssPage(sPage, sTag + "_PV", sTag + "_SP", sTag + "_OP");
END
```

通过此函数，可以调用 AssMine() 函数（例如，从命令按钮），并传递一个标签 (DEV1)，如下所示：

```
AssMine("PageName", "DEV1");
```

自行编写 Cicode 函数调用超级精灵提供了更多灵活性，但还可以使用超级精灵（例如，从命令按钮）调用 Ass 函数，如下所示：

```
执行命令    AssPage("%Tag%", "%Tag%_PV", "%Tag%_SP", "%Tag%_OP");
```

使用上述超级精灵时，只需输入页面名和标签一次。

必须将标签名称传递给超级精灵函数（通过将其在括号中括起）。不能传递标签参数值。例如，如果传递 `%tag%_SP`（无括号），则会将变量参数值而不是标签名称传递给超级精灵，关联将不会成功，同时可能导致运行错误。

另请参见

[通过精灵使用结构化标签](#)

隐藏图形对象

您可以配置一个图形对象，这样，如果编译时未在标签数据库中定义指定用于对象的变量标签，该对象将不会显示在图形画面上。

在对象属性的**隐藏时间**域中输入的表达式用于确定对象是否显示。此表达式评估为 `TRUE` 或 `FALSE`，对象在它是 `TRUE` 时隐藏。

您可以定义变量标签和对象的隐藏条件，方法是在配置对象时向**隐藏时间**域中输入 `IFDEF` 语句。`IFDEF` 语句由编译器进行评估，最终表达式或变量标签的值确定是否隐藏对象。

这可以显著降低所需精灵的数量，因为配置工程师无需为迎合由存在细微差异的标签驱动的操作而生成数个较小的精灵。

有关使用 `IFDEF macro` 隐藏图形对象的详细信息，请参见 [IFDEF macro](#)。要总体了解有关 `IFDEF macro` 的详细信息，请参见 `IFDEF`。

IFDEF宏

`IFDEF` 语句由三个参数组成：

```
IFDEF (<"Tag name">, <Hidden When value if tag defined>, <Hidden When value if tag undefined>)
```

第一个参数包含标签变量名。如果编译工程时在标签数据库中定义了该标签变量，则第二个参数会在**何时隐藏**字段中代替 `IFDEF` 语句。如果未定义标签变量，则**何时隐藏**字段将包含第三个参数。

示例 1

```
IFDEF("Bit_1", 0, 1)
```

在上面的示例中，如果在标签数据库中定义了 `Bit_1`，则**何时隐藏**字段中的值将为 0。如果未定义 `Bit_1`，该值将为 1。由于对象在值为 `TRUE` 时隐藏，因此在未定义 `BIT_1` 时（即，当**何时隐藏**字段包含 1 时），对象将隐藏。

示例 2

```
IFDEF("Bit_2",,"1")
```

正如在示例 2 中看到的，如果省略第二个参数，将使用在第一个参数中指定的标签变量。因此，如果定义了 Bit_2，**何时隐藏**字段将包含 Bit_2。然后，将使用标签变量 Bit_2 的值确定是否隐藏对象。非零值将等于 TRUE，这将使对象被隐藏。

如果未定义 Bit_2，“何时隐藏”表达式的值将等于 1 (TRUE)，对象将被隐藏。

在“何时隐藏”字段中输入 IFDEF 语句：

1. 双击要为其编辑该字段的图形对象。
2. 选择**外观**分页标签。
3. 单击**何时隐藏**字段，输入 IFDEF 语句。
4. 点击**确定**。

Chapter: 33 使用多国语言工程

Vijeo Citect 的语言切换机制使得用户能够使用某种语言来配置一个工程，并用另一种语言来作为运行时的文本输出，例如报警描述、按钮文字、键盘/报警日志、图形文字、Citect 字符串等等。用户也能在运行时动态地改变语言。

注意：

- 没有自动化翻译，您必须安排将本国语言字符串翻译成所需的本地语言并将它们插入本地化语言数据库。
- 翻译之后，您应检查外国语言接口并且确认翻译的字符串是否适合其相应的图形元素。

例如，假设用户的本国语言为英语，用户可以在对工程进行配置的过程中输入一个英文的报警描述但却将该报警描述采用法语或者德语（或者任何其它的语言）进行显示。用户所希望使用的语言可以在运行工程前指定，也可以在系统运行时动态地进行改变（调用 `SetLanguage()` 函数）而不会对工程的正常运行产生任何影响。

Vijeo Citect 区别 *本国语言*（开发者的语言）和 *本地语言*（终端用户的语言）。语言的变换是通过使用语言数据库来实现的。在语言数据库中有一个字段中记录的是本国语言文字，而另一个字段中则是其翻译好的本地语言文字。在工程运行的时候，本国语言文字将被其相应的本地文字所取代。

报警和键盘日志都能用本国与本地语言进行记录，可以使得本国与本地用户都读得懂历史日志。数据可被记录到同一个设备，或不同设备中。

另请参见

[改变语言](#)

改变语言

本节介绍如何改变 Vijeo Citect 使用的语言。

另请参见

[对文本做标记来改变语言](#)

[语言数据库](#)

[多语言](#)

[多工程](#)

[在运行期间改变语言](#)

[用不同的语言来记录数据](#)

[ASCII 和 ANSI 字符集](#)

[OEM 字符集](#)

标记要改变语言的文字

在工程开发过程中，必须用语言改变指示器像下面这样标记要在运行时改变为另一种语言的文字：

```
@( Native Text [,Width [,Justify]])
```

其中，**Native Text** 是在配置时需要显示的待辨识文字。该文字将在运行时被其对应的本地语言所代替。请注意，当指定母语文字的范围时，需要用括号将其括起；**width** 和 **Justify** 为可选项（用方括号表示）。

例如，假设母语为英语，可输入以下报警描述：

报警描述

@(Motor Inoperative)

此指示器有两个作用：将该文字标记为母语，并告诉 **Vijeo Citect** 在运行时将该文字从母语转换为本地语言。

缺省情况下，在此处输入的文字可以是任何大小写字母的组合。换句话说，*Motor Inoperative* 将被视为与 *motor Inoperative* 或 *MOTOR Inoperative* 相同的字符串，它们将具有相同的本地语言译文。用户可通过将 `[Language]CaseSensitive` 参数设置为 1 使得系统区分大小写。

width 可以是 0 到 254 之间的任何值。若本地文字比指定值长，本地文字会被截断并且同母语左对齐。如果没有指定宽度，该字段将是本地文字的长度，该文字也将左对齐。

Justify 指定文字对齐方式，并且只能与 **width** 一起使用。**Justify** 可以是以下值中的一个：

- **l** 或 **L** - 左对齐
- **r** 或 **R** - 右对齐
- **c** 或 **C** - 居中
- **n** 或 **N** - 无

例如，要将前面提到的本地文字限制在 20 个字符并且右对齐，我们可以采用下面的方法：

报警描述

@(Motor Inoperative, 20, R)

通常是格式一部分的字符 - **@**, **()** - 也可以在母语文字中使用。为此，请将脱字符号 (^) 字符置于它们之前。例如，要在其中插入一个逗号而不破坏其格式，可以采用下面的方法：

报警描述	@(Motor Inoperative^, thermal overload, 20, R)
------	--

注意： 脱字符号(^)字符也会出现在运行时或语言数据库中。

另请参见
[语言数据库](#)

语言数据库

在对工程进行编译时，Vijeo Citect 会创建一个语言数据库（dBASE III 格式），由两个字段组成：NATIVE 和 LOCAL。用语言变更指示器标记过的文字将自动写入 NATIVE 字段。接着用户就可以打开数据库并在 LOCAL 字段中键入翻译好的文字。

例如：

NATIVE	LOCAL
Line Disconnected Alarm at Line Speed {LineSpeed1}	<Line Disconnected Alarm at Line Speed {LineSpeed1}> 的译文
Main Menu page	<Main Menu page 的译文>
Conveyor Belt Trip	<Conveyor Belt Trip 的译文>

当工程在运行时，*Line Disconnected Alarm at Line Speed {LineSpeed1}* 的译文将替代原来的英文文字显示出来，*Main Menu page* 的译文将替代原来的英文文字显示出来，等等。

对于在工程运行时自动发生的语言改变，用户必须在工程运行之前通过使用 [Language]LocalLanguage 参数指定要使用的语言数据库。否则，用户就必须自己在运行时使用 SetLanguage() 函数更改所使用的语言。

如果没有母语文字字符串的等效译文，就会缺省显示该母语文字。用户可通过将 [Language]DisplayError 参数设置为 1（缺省值为 0）来指定显示“#MESS”而不是母语文字。

对于单字节语言（例如法语），可以用 Microsoft Excel 对数据库进行编辑；对于双字节语言（例如中文），应使用 Visual FoxPro 进行编辑。

缺省情况下，在编译过程中产生的语言数据库称为 **English.dbf**（此文件可使用 [Language]LocalLanguage 参数进行更改）。该数据库保存在工程目录下。一旦创建数据库之后，每次编译时都将对其进行更新。自上次编译后标记的文字都会附加到数据库；而数据库的其余部分保持不变。

另请参见
[多语言](#)

多语言

注意：要使用波罗地海语、中欧语、古斯拉夫语、希腊语、土耳其语、亚洲语言或书写习惯为从右向左的语言（阿拉伯语、希伯来语、波斯语以及乌尔都语）中的字符，操作系统必须具有相应语言的 Windows 版本，或安装支持该语言的系统。

每个本地语言都必须有各自的语言数据库，这样可以在运行时代替指定母语显示。同时，它还必须使用 [Language]LocalLanguage 参数设备为本地语言。如果在编译之前设置此参数，则 Vijeo Citect 将自动创建/更新相关语言数据库。

例如，要在运行时显示法语文本，请将 [Language]LocalLanguage 参数设置为“法语”，在工程中使用 @() 标记所有必需的母语文本，然后进行编译。编译完成后，在工程目录中查找 **French.dbf** 并将其打开，在**本地**字段中输入所需的法语译文，然后保存数据库。工程处于运行状态时，所有标记的母语文本将被适当的法语文本替代。

由于数据库的数量是任意的，因此可以使用任意数量的不同语言。

进行编译时，无论使用 [Language]LocalLanguage 参数设置为本地语言的是哪个数据库，都将在**母语**字段中输入标记有语言更改标识符的所有文本。因此，应了解进行编译之前设置的数据库。

同时，如果多个语言数据库的母语相同，请记住，新标记的文本只附加到**当前**本地语言数据库（由 [Language]LocalLanguage 参数指定）。要将此文本添加到母语相同的其它数据库，请更改 [Language]LocalLanguage 参数，更新页面，然后对每个数据库重新编译。请记住，对于每个数据库，只有添加最后的编译后才会进行相关更改。

另请参见
[多工程](#)

多工程

语言数据库可以包含实际上工程中并不包括的条目。这表示开发的一个语言数据库可适用于很多工程。

另请参见
[运行时更改语言](#)

在运行期间改变语言

运行期间需要显示的项目（如报警描述、按钮文字、键盘/报警日志、图形文字、Cicode 字符串等）所使用的语言可以在运行时利用 SetLanguage() 函数动态地进行更改。在进行更改时所有与工程有关的正常操作都会继续而不受影响。

注意：当使用 `SetLanguage()` 函数更改语言后，窗体并不会自动更新语言。必须先将窗体关闭然后再重新打开才能使语言更改生效。

指定的语言数据库中缺少的本地译文会被其相应的母语文字所取代。用户可通过将 `[Language]DisplayError` 参数设置为 1（缺省值为 0）来指定显示 "#MESS" 而不是母语文字。

另请参见
[用不同的语言来记录数据](#)

用不同语言来记录数据

报警以及键盘日志既可以用母语也可以用本地语言来处理。这意味着母语用户以及本地语言用户都可以阅读历史记录。日志可以采用同一设备，也可以采用不同设备。

生成本地语言的日志时都采用标准的字段名称。例如，若在某个报警分类的标准格式字段中输入 `{NAME} {DESC} {COMMENT}`，则该类别所有报警的报警名称、描述和注释都将用本地语言写入日志中。

所有支持这种语言自动变更机制的字段也同样可以用母语记录到日志中。要实现此目的，只需要预先将字段名设置为 **NATIVE**。举例来说，若要记录某类报警的名称、描述和注释，只需在该报警类型的格式字段中输入 `{NATIVE_NAME} {NATIVE_DESCRIPTION} {NATIVE_COMMENT}` 即可。

要想对同一设备的日志既使用母语又使用本地语言，只需以格式字段在标准字段和母语字段中输入内容即可。要将它们记录到不同设备，请使用由两个设备组成的组，并以该格式为一个设备输入本地字段，并以该格式为另一个字段输入母语字段。

另请参见
[ASCII和ANSI字符集](#)

ASCII 和 ANSI 字符集

屏幕上的每一个字符都是由一个代码（数值）来定义的。操作系统和应用程序需要了解这些代码来将相应的含义赋予各个字符。字符集为每个字符提供了一个代码。要保证操作系统/应用程序能够正确地诠释这些字符，必须使用正确的字符集。

注意：字符集同字体是有区别的。字体定义的是某个字符的视觉/外观属性，而不是它的含义。

ASCII（美国信息交换标准码，American Standard Code for Information Interchange）是一种被广泛采用的用来说明基本的英文字母-数字字符集的 7 位代码。例如，字符大写字母 "A" 所对应的 ASCII 代码值为 65，而字符小写字母 "a" 的 ASCII 码值为 97。

ASCII 字符集包含 96 个字符，它通常作为各种协议以及文件的一种标准。

Vijeo Citect 大多采用 ANSI（美国国家标准化组织，American National Standards Institute）字符集。ANSI 字符集是基于语言本身的，对于每一个不同语言版本的 Windows（法语、朝鲜语等）都需要各自特定的 ANSI 字符集。代码 32 到 127 包含了标准的 ASCII 字符。

Windows 使用的是 Unicode，但仍然支持 ANSI 字符集。针对 Unicode 创建了多个 Vijeo Citect 实用程序，例如过程分析器。Unicode 通过采用一种 16 位（世界通用的）字符编码标准对所有已知的字符集进行了编码。

另请参见

[OEM 字符集](#)

OEM 字符集

OEM 字符集是 MS-DOS 或 Console 应用程序（取决于操作系统）使用的字符集。大多数 OEM 字符集同 ANSI 字符集是不匹配的。例如，在 MS-DOS 字符集中常用的线状字符已经被 ANSI 字符集中的语言字符所取代了。

如下所述，在建立多语言工程中应对 ANSI 以及 OEM 字符集在语言字符串的存储和解释方面的作用给予充分考虑。

语言配置信息保存在 dBASE 文件中（一种主要应用于 MS-DOS 应用程序的数据库标准）。在这种文件中字符串信息传统上是保存为 OEM 字符的。当使用 Windows 应用程序（例如 Excel）来编辑 dBASE 文件时，屏幕上所显示的字符将是 ANSI 字符集中的字符。当用户将编辑好的信息保存到该 dBASE 文件中时，Excel 会将其转换为相应的 OEM 字符。要使该转换能够正确无误的工作，该 OEM 字符集必须同在 Excel 中使用的 ANSI 字符集相兼容。例如，如果用户在工程中建立了一个俄语的字符串（使用 Excel），那么 OEM 字符集必须支持俄语（古斯拉夫语）字符集。Windows 所采用的 OEM 字符集主要是由用户的系统设置所决定的而且是不能随意更改的。这就给多语言工程带来了挑战。

例如，我们来看一个想要支持俄语、法语和英语的工程。我们采用 Excel 来建立 dBASE 文件。当 Excel 保存信息时，信息从各自的 ANSI 字符集翻译成 OEM 等效字符集。要显示该信息，Vijeo Citect 需要将该信息从 OEM 转换回 ANSI。然而，俄语字符需要一个古斯拉夫语的 OEM 字符集，法语以及英语需要一个拉丁语的 OEM 字符集。由于 Windows 一次只能使用一个 OEM 字符集（无法动态更改字符集），在任何给定的会话期间，此三种工程语言中仅有一种能够获得正确支持。

在一个 Vijeo Citect 工程中用不同的字符集来支持多语言的唯一方法就是确保将 dBase 文件中的语言信息保存为 ANSI（而非 OEM）格式。此外，必须将 [CtEdit]ANSItoOEM 参数设置为 0（零），以防发生自动转换。所以对于开发者来说在建立工程过程中的挑战就是信息保存的第一步，因为多数的应用程序将会把信息保存为 OEM 格式。

注意：您的安装 CD 盘上的范例目录下有一个多语言工程。在这个工程中允许您采用 ANSI 格式将信息输入到语言 dBASE 文件中。

Chapter: 34 使用 OPC 服务器 DA2.0

借助 Vijeo Citect OPC 服务器，可通过任何 OPC 客户机应用程序（v1.0 或 v2.0）访问 Vijeo Citect 运行时环境中可用的所有数据。

本部分内容包括：

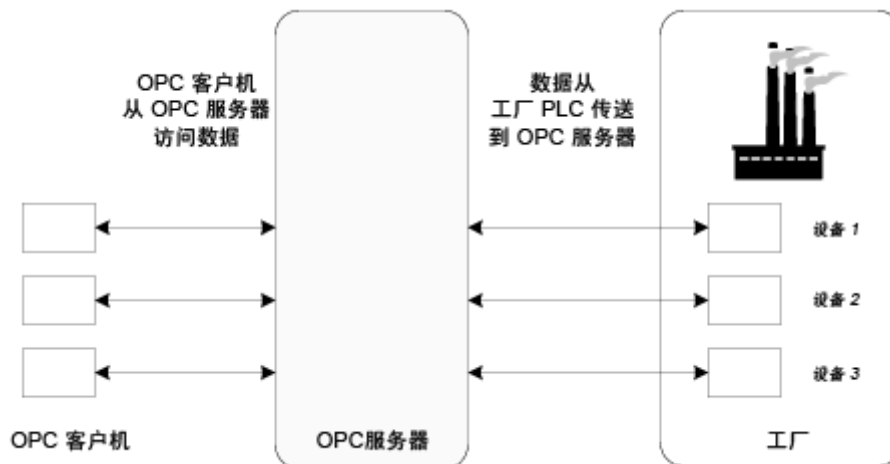
- [OPC 概述](#)
- [Vijeo Citect OPC 服务器](#)
- [Vijeo Citect OPC 服务器安装](#)
- [配置对 Vijeo Citect OPC 服务器的远程访问](#)
- [疑难解答](#)

OPC 概述

OPC（工程控制 OLE）是一组开放标准的通讯规范，适用对象是工业自动化行业。OPC 提供了一种供应用程序从 PLC 和数据库等源中访问数据的通用方法。

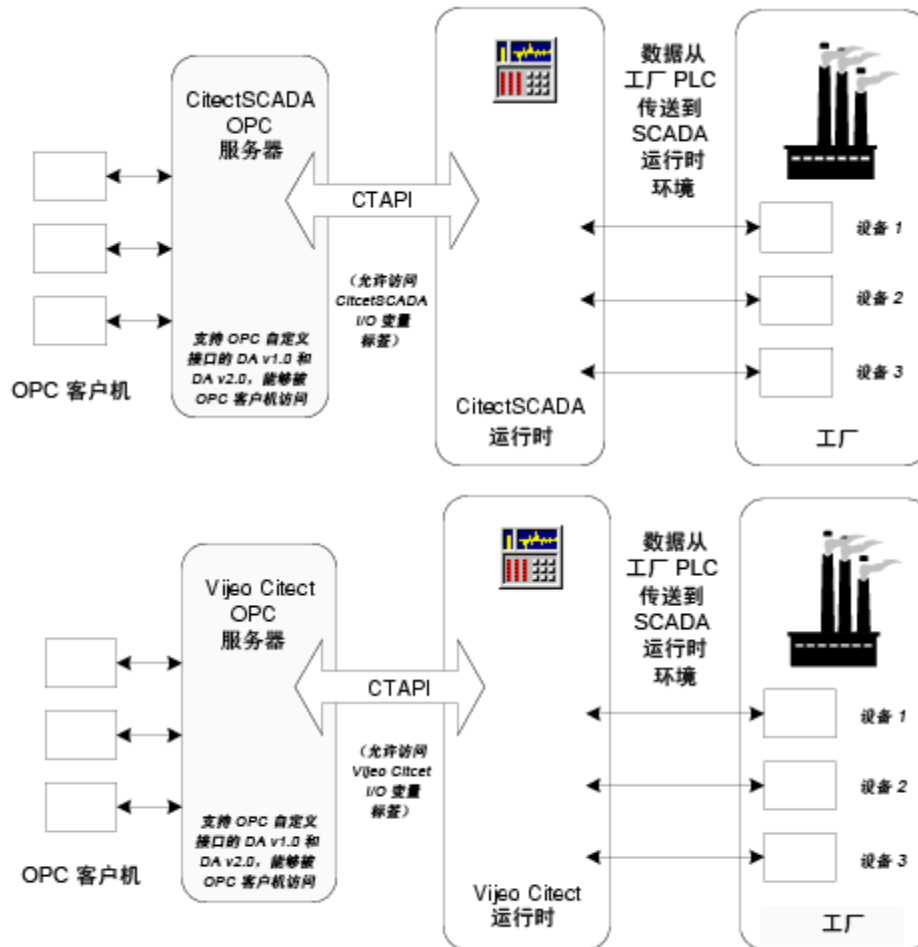
这些规范由 OPC 基金会维护并会被定期检查，以确认其符合行业需要。OPC 基金会的目标是通过使用开放式标准提高互操作性（开放式连接）。Vijeo Citect Runtime 实施 OPC 规范中的 vDA2.05。有关 OPC 规范的详细信息，请浏览到 www.opcfoundation.org。

当使用 OPC 服务器从工厂的物理设备中收集数据时，所有 OPC 客户机都能直接从 OPC 服务器中读取数据，如下所示：



Vijeo Citect OPC 服务器

使用 Vijeo Citect 来监控工厂时，就会从 PLC 收集数据并显示在 Vijeo Citect 运行期间环境中。OPC 客户端可以从 Vijeo Citect 经过 OPC 接口到 OPC 服务器访问设备和标签信息，这会依次与 CtAPI 接口和 Vijeo Citect 运行期间交互，如下图所示：



Vijeo Citect OPC 服务器可以从相同机器或不同机器上的 OPC 客户端调用。当连接至与客户端运行在同一计算机上的 OPC 服务器时，使用 Citect.OPC 连接选项。当连接到在远程计算机上运行的 OPC 服务器时，使用 Citect.OPCRemote 选项。

当启动 Vijeo Citect 运行环境时，OPC 服务器会自动启动，并且不需要进行配置。

Vijeo Citect OPC 服务器安装

在 Vijeo Citect 安装过程中，将：

- 安装 Vijeo Citect OPC 服务器。

- 安装 OPC 核心组件。
- 配置工作站的注册表设置以支持 OPC 服务器。下面两个表的内容便是适合于以下这两项的注册表设置：
 - 正在运行的 OPC 服务器 - Citect.OPC
 - 本地/远程 OPC 服务器 - Citect.OPCRemote

正在运行的 OPC 服务器

正在运行的 OPC 服务器	
应用程序 ID (AppID)	Citect.OPC
类 ID (CLSID)	{BA198B61-32E3-11d1-A1B5-00805F35623C}
二进制文件	CtOpc32.dll
接口	DA 1.0 自定义接口:
OPCServer:	IOPCServer IOPCBrowseServerAddressSpace (可选)
OPCGroup:	IOPCItemMgt IOPCGroupStateMgt IOPCSynchIO IOPCASynchIO IDataObject
	DA 2.0 自定义接口:
OPCServer:	IConnectionPointContainer IOPCItemProperties
OPCGroup:	IOPCASyncIO2 IConnectionPointContainer
系统要求	2000、XP 和 Windows Server 2003
服务器组织	Flat (OPC_NS_FLAT)
实现的类别	OPC DA Servers v1.0 {63D5F430-CFE4-11D1-B2C8-0060083BA1FB}
	OPC DA Servers v2.0 {63D5F432-CFE4-11D1-B2C8-0060083BA1FB}

本地/远程 OPC 服务器

本地/远程 OPC 服务器	
应用程序 ID (<i>AppID</i>)	Citect.OPCRemote
类 ID (<i>CLSID</i>)	{ BA198B62-32E3-11d1-A1B5-00805F35623C }
二进制文件	CtOpc32.exe
接口	<i>DA 1.0 自定义接口:</i>
OPCServer:	IOPCServer IOPCBrowseServerAddressSpace (可选)
OPCGroup:	IOPCItemMgt IOPCGroupStateMgt IOPCSynchIO IOPCASynchIO IDataObject
	<i>DA 2.0 自定义接口:</i>
OPCServer:	IOPCCommon IConnectionPointContainer IOPCItemProperties
OPCGroup:	IOPCASynchIO IConnectionPointContainer
系统要求	2000、XP 和 Windows Server 2003
服务器组织	Flat (OPC_NS_FLAT)

注意：在 OPC 架构中，每种服务器均分配一个唯一标识符，称为类 ID。OPC 客户端使用上述 128 位编号（通常显示为 6B29FC40-CA47-1067-B31D-00DD010662DA 这样的形式）来访问特定供应商的 OPC 服务器。为了使它对用户更容易，这些编号经常被称为 *ProgID*（*程序标识符*）的更易管理的字符串标识符所引用。这些标识符通常采取 **供应商.应用程序** 附加可选版本号的形式；例如，Vijeo Citect OPC 服务器的 ProgID 为 **Citect.OPC**。

安装成功后，Vijeo Citect OPC 服务器即可随时使用。当启动 Vijeo Citect 运行期间环境时，OPC 客户端访问的 OPC 接口会自动启动。仅当要通过远程 OPC 客户端访问 Vijeo Citect OPC 服务器时，才需要进行进一步的配置。有关详细信息，请参见 [配置为远程访问 Vijeo Citect OPC 服务器](#)。

配置对 OPC 服务器的远程访问

可以使用任何 OPC 客户机访问 Vijeo Citect OPC 服务器。虽然每个 OPC 客户机都有自己的界面，操作方式也不尽相同，但是要访问通过 OPC 服务器公开的数据，必须执行一些相同的步骤：

- [配置 OPC 服务器](#)
- [配置 OPC 客户机](#)
- [创建数据组](#)
- [添加数据项目到组](#)

有关如何操作 OPC 客户机的详细信息，请参考产品的文档。

注意：如果客户机和服务器位于同一台计算机上，则不需要配置。

配置 OPC 服务器

OPC 使用 Microsoft Distributed COM (DCOM) 架构进行工作。这表示如果 OPC 客户机连接了远程 OPC 服务器，您必须在客户机和服务器的机器上都配置 DCOM 安全设置。

注意：必须为用户指定足够的 Citect 和 DCOM 安全性权限，否则 OPC 客户端无法连接到 OPC 服务器。

配置 DCOM 安全设置的过程因您使用的平台的不同而不同：

- [Windows Vista 和 Windows XP SP2](#)
- [Windows 2000](#)
- [Windows Server 2003](#)

Windows Vista 和 Windows XP SP2

此节介绍了如何在 Windows Vista 和 Windows XP Service Pack 2 上配置 OPC 服务器。

注意：配置您的 OPC 服务器之后，您必须重新启动服务器以使设置生效。

配置 OPC 服务器：

1. 在 Windows 控制面板中，双击**安全中心**，然后单击 **Windows 防火墙**。
2. 选择**例外**分页标签，然后单击**添加程序**。
3. 找到您希望添加为例外的 OPC 服务器和客户机。您必须添加 **Microsoft 管理控制台**（由 DCOM 配置实用程序使用）和 OPC 实用程序 OPCEnum.exe 到 Windows/System32 文件夹中。

您可能必须**浏览**计算机中安装的其他可执行程序以便将它们添加为例外。请注意，例外列表里仅添加 .exe 文件。对于处理中的 OPC 服务器和客户机（DLL 和 OCX），您必须向列表中添加调用它们的 .exe 应用程序。

4. 单击**确定**保存您的例外。
您必须现在添加 TCP 端口 135 以启动 DCOM 通信并允许传入回送请求。
5. 在“Windows 防火墙例外”分页标签上，单击**添加端口**。
6. 在**名称**文本框中键入 **DCOM**。
7. 在**端口号**文本框中键入 **135** 并选择 **TCP** 选项。
8. 单击**确定**保存您的更改。
您已经定义了例外，现在可以重新启动防火墙了。
您现在就可以为启动、激活和访问权限配置 DCOM 了。
9. 选择**开始|运行**。键入 **dcomcnfg** 并单击**确定**。
将出现**组件服务**对话框。
10. 在树状窗格中，找到并选择**组件服务**下的**计算机**。
11. 在右侧的窗格中右键单击**我的电脑**，然后从上下文菜单中选择**属性**。
12. 选择**缺省属性**分页标签。确保按下面的说明调整设置：

属性	设置
启用计算机上的分布式 COM	选择
启用计算机上的 COM 互联网服务	未选择
缺省分布式 COM 通讯属性	
缺省验证级别	连接
缺省假冒级别	识别
提供参考追踪的附加安全	未选择

- 单击**应用**保存您的更改。
13. 您必须随后编辑访问权限。选择 **COM 安全设置** 分页标签，然后单击**访问权限**部分中的**编辑限制**。
 14. 使用**访问权限**对话框选择**匿名登录**，然后选中**远程访问**选项的**允许**。
这是 OPCEnum.exe 正确运行所需要的，对于所有 OPC 服务器和客户端，请将其 DCOM 身份验证级别设置为“无”，以允许匿名连接。如果不使用 OPCEnum.exe，则不需要为匿名用户启用远程访问。
 15. 单击**应用**保存您的更改。
您必须随后编辑启动和激活权限。
 16. 在“缺省 COM 安全性”分页标签的**启动和激活权限**部分中选择**编辑限制**。
在**组或用户名**列表中，选择**每个人**，然后选中**允许**列中的所有复选框。
 17. 单击**确定**保存您的更改。
最后，您就可以编辑参与 OPC 通信的每个用户或组的**访问和启动**的缺省权限（例如“OPC 用户”）。

18. 在“COM 安全性”分页标签的**访问权限**部分，单击**编辑缺省值**。选择所需的用户/组并允许**本地访问**和**远程访问**。单击**确定**保存您的更改。
19. 在“COM 安全性”分页标签的**启动和激活权限**部分，单击**编辑缺省值**。选择所需的用户/组并允许**本地访问**和**远程访问**。单击**确定**保存您的更改。

到现在就完成了 OPC 服务器的配置。

注意： 您必须重新启动此计算机以使设置生效。

现在便可以 [配置 OPC 客户机](#)了。

Windows 2000

此节介绍了如何在 Windows 2000 上配置 OPC 服务器。

配置 OPC 服务器：

1. 选择**开始|运行**。键入 **dcomcnfg** 并单击**确定**。
将出现**组件服务**对话框。
2. 在树状窗格中，找到并选择**组件服务**下的**计算机**。
3. 在右侧的窗格中右键单击**我的电脑**，然后从上下文菜单中选择**属性**。
4. 选择**缺省属性**分页标签。确保按下面的说明调整设置：

属性	设置
在此计算机上启用分布式 COM	选定
在此计算机上启用 COM Internet 服务	不选定
缺省分布式 COM 通信属性	
缺省身份验证级别	连接
缺省模拟等级	标识
为参考跟踪提供额外的安全措施	不选定

单击**应用**保存您的更改。

您必须编辑**访问权限**和**启动和激活权限**设置。

5. 选择**缺省安全设置**分页标签，然后单击**缺省访问权限**部分中的**编辑缺省值**。
6. 在**注册表值权限**对话框中，选择要编辑权限的用户，然后从**访问类型**菜单中选择**允许访问**。

此设置是 OPCEnum.exe 运行所需要的，对于某些 OPC 服务器和客户机，请将其 DCOM 身份验证级别设置为“无”，以允许匿名连接。如果不使用 OPCEnum.exe，则不需要为匿名用户启用远程访问。

单击**应用**保存您的更改。

现在您必须编辑缺省启动权限。

- 单击**缺省启动权限**部分中的**编辑缺省值**。确保将所有必需用户的访问类型（包括系统和交互用户）设置为**允许启动**。
单击**确定**保存您的更改。
到现在就完成了 OPC 服务器的配置。

注意：您必须重新启动此计算机以使设置生效。

现在便可以 [配置 OPC 客户机](#)了。

Windows Server 2003

此节介绍了如何在 Windows Server 2003 上配置 OPC 服务器。

配置 OPC 服务器：

- 选择**开始|运行**。键入 **dcomcnfg** 并单击**确定**。
将出现**组件服务**对话框。
- 在树状窗格中，找到并选择**组件服务**下的**计算机**。
- 在右侧的窗格中右键单击**我的电脑**，然后从上下文菜单中选择**属性**。
- 选择**COM 安全设置**分页标签，然后单击**访问权限**部分中的**编辑限制**。
- 使用**访问权限**对话框选择**匿名登录**，然后选中**远程访问**选项的**允许**。
这是 OPCEnum.exe 正确运行所需要的，对于所有 OPC 服务器和客户机，请将其 DCOM 身份验证级别设置为“无”，以允许匿名连接。如果不使用 OPCEnum.exe，则不需要为匿名用户启用远程访问。
- 单击**应用**保存您的更改。
您必须随后编辑启动和激活权限。
- 在“缺省 COM 安全性”分页标签的**启动和激活权限**部分中选择**编辑限制**。
在**组或用户名**列表中，选择**每个人**，然后选中**允许**列中的所有复选框。
- 单击**确定**保存您的更改。
最后，您就可以编辑参与 OPC 通信的每个用户或组的**访问和启动**的缺省权限（例如“OPC 用户”）。
- 在“COM 安全性”分页标签的**访问权限**部分，单击**编辑缺省值**。选择所需的用户/组并允许**本地访问**和**远程访问**。单击**确定**保存您的更改。
- 在“COM 安全性”分页标签的**启动和激活权限**部分，单击**编辑缺省值**。选择所需的用户/组并允许**本地访问**和**远程访问**。单击**确定**保存您的更改。
到现在就完成了 OPC 服务器的配置。

注意：现在您必须重新启动此计算机以使设置生效。

现在便可以 [配置 OPC 客户机](#)了。

配置 OPC 客户机

必须先配置 OPC 客户机，然后它才能与 OPC 服务器通讯。

配置 DCOM 安全设置的过程因您使用的平台的不同而不同：

- [Windows Vista 和 Windows XP SP2](#)
- [Windows 2000](#)
- [Windows Server 2003](#)

Windows Vista 和 Windows XP SP2

此节介绍了如何在 Windows Vista 和 Windows XP Service Pack 2 上配置 OPC 客户端。

注意：配置 OPC 客户端后，用户必须重新启动客户端才能使设置生效。

配置 OPC 客户端：

1. 在 Windows 控制面板中，双击**安全中心**，然后单击 **Windows 防火墙**。
2. 选择**例外**标签然后单击**添加程序**。
3. 定位要作为例外添加的 OPC 服务器和客户端。用户必须在 Windows/System32 文件夹里添加 **Microsoft 管理操作台**（由下面的 DCOM 配置应用程序使用）和 OPC 应用程序 OPCEnum.exe。
用户也许还要**浏览**其他安装在计算机上的可执行程序，以便将它们作为例外添加。请注意，例外列表里仅添加 .exe 文件。而对于进程中的 OPC 服务器和客户端（DLLs 和 OCXs），用户必须将调用它们的 .exe 应用程序添加到列表。
4. 单击**确定**保存例外。
现在用户必须添加启动 DCOM 通信的 TCP 端口 135 并且允许传入回送请求。
5. 单击 Windows 防火墙例外标签中的**添加端口**。
6. 在**名称**文本框中键入 **DCOM**。
7. 在**端口号**文本框中键入 **135** 并选择 **TCP** 选项。
8. 单击**确定**保存更改。
现在已经重新定义了例外，因此可以重新启动防火墙。
现在要配置 DCOM 的启动、激活和访问权限。
9. 选择**起始|运行**。键入 **dcomcnfg** 并单击**确定**。
会出现**组件服务**对话框。
10. 在树窗格里**组件服务**下定位并选择**计算机**。
11. 右键单击右面窗格中的**我的电脑**并从上下文菜单中选择**属性**。
12. 选择**缺省属性**标签。确保设置调整如下：

属性	设置
启用计算机上的分布式 COM	选择
启用计算机上的 COM 互联网服务	未选择

属性	设置
缺省分布式 COM 通讯属性	
缺省验证级别	连接
缺省假冒级别	识别
提供参考追踪的附加安全	未选择

单击**应用**保存更改。

13. 然后用户必须编辑访问权限。在**访问权限**部分中选择 **COM 安全** 标签，并且单击**编辑界限**。
14. 使用**访问权限**对话框选择**匿名登陆**，然后选中**远程访问的允许**。
此操作对于 OPCEnum.exe 正确地运行和并且对于任何为了允许匿名连接而设置 DCOM 确认级别为“无”的 OPC 服务器和客户端是必需的。如果不使用 OPCEnum.exe 就不用为匿名用户启用远程访问。
15. 单击**应用**保存更改。
然后用户必须编辑启动和激活权限。
16. 在缺省 COM 安全标签的**启动和激活权限**部分中选择**编辑界限**。
在**组或用户名**列表中选择**每一个**然后在**允许**栏中选中所有复选框。
17. 单击**确定**保存更改。
最后准备为每一个参加 OPC 通讯的用户或组（例如，“OPC 用户”）编辑**访问和启动**的缺省权限。
18. 在“COM 安全性”分页标签的**访问权限**部分，单击**编辑缺省值**。选择需要的用户/组并且允许**本地访问**和**远程访问**。单击**确定**保存更改。
19. 在 COM 安全标签的**启动和激活权限**部分单击**编辑缺省**。选择需要的用户/组并且允许**本地访问**和**远程访问**。单击**确定**保存更改。
一些老式 OPC 客户端不能在客户端机器上定位 Vijeo Citect OPC 服务器。在这种情况下，复制位于 bin 目录下的 ctopc.reg 文件到客户端机器。双击此文件将特定的 OPC 服务器数据添加到注册表。
现在 OPC 客户端已经配置完成。

注意：为了使设置生效必须重启机器。

Windows 2000

此节介绍了如何在 Windows 2000 上配置 OPC 客户机。

配置 OPC 客户机：

1. 选择**开始|运行**。键入 **dcomcnfg** 并单击**确定**。
将出现**组件服务**对话框。
2. 在树状窗格中，找到并选择**组件服务**下的**计算机**。
3. 在右侧的窗格中右键单击**我的电脑**，然后从上下文菜单中选择**属性**。

4. 选择**缺省属性**分页标签。确保按下面的说明调整设置：

属性	设置
在此计算机上启用分布式 COM	选定
在此计算机上启用 COM Internet 服务	不选定
缺省分布式 COM 通信属性	
缺省身份验证级别	连接
缺省模拟等级	标识
为参考跟踪提供额外的安全措施	不选定

单击**应用**保存您的更改。

您必须编辑**访问权限**和**启动和激活权限**设置。

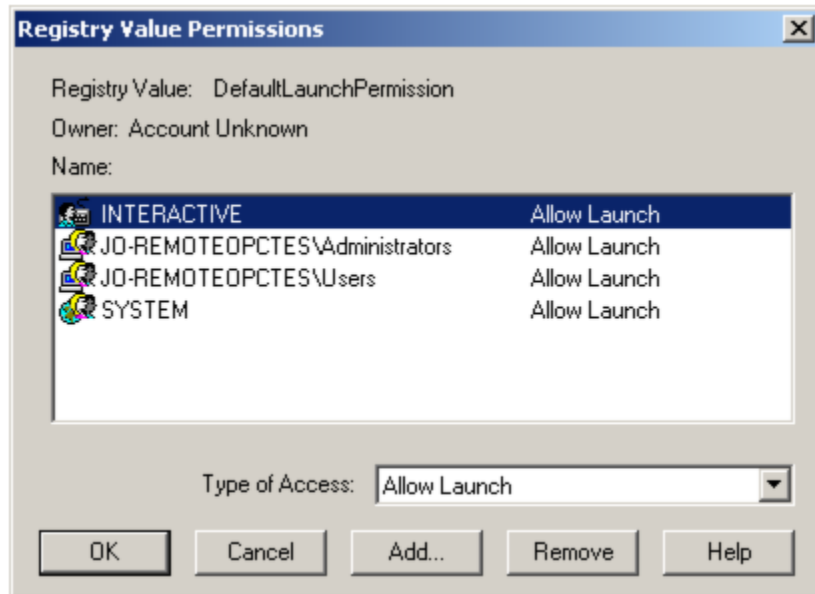
5. 选择**缺省安全设置**分页标签，然后单击**缺省访问权限**部分中的**编辑缺省值**。
 6. 在**注册表值权限**对话框中，选择要编辑权限的用户，然后从**访问类型**菜单中选择**允许访问**。

此设置是 OPCEnum.exe 运行所需要的，对于某些 OPC 服务器和客户机，请将其 DCOM 身份验证级别设置为“无”，以允许匿名连接。如果不使用 OPCEnum.exe，则不需要为匿名用户启用远程访问。

单击**应用**保存您的更改。

现在您必须编辑缺省启动权限。

7. 单击**缺省启动权限**部分中的**编辑缺省值**。确保将所有必需用户的访问类型（包括系统和交互用户）设置为**允许启动**。



单击**确定**保存您的更改。

到现在就完成了 OPC 客户机的配置。

一些比较旧的 OPC 客户机无法找到客户机计算机上的 Vijeo Citect OPC 服务器。如果出现这种情况，请将位于 bin 目录中的 ctopc.reg 文件复制到客户机计算机中。双击此文件以便将 OPC 服务器的特定数据添加到注册表中。

注意： 您必须重新启动此计算机以使设置生效。

Windows Server 2003

此节介绍了如何在 Windows Server 2003 上配置 OPC 客户机。

配置 OPC 客户机：

1. 选择**开始|运行**。键入 **dcomcnfg** 并单击**确定**。
将出现**组件服务**对话框。
2. 在树状窗格中，找到并选择**组件服务**下的**计算机**。
3. 在右侧的窗格中右键单击**我的电脑**，然后从上下文菜单中选择**属性**。
4. 选择**COM 安全设置**分页标签，然后单击**访问权限**部分中的**编辑限制**。
5. 使用**访问权限**对话框选择**匿名登录**，然后选中**远程访问**选项的**允许**。
这是 OPCEnum.exe 正确运行所需要的，对于所有 OPC 服务器和客户机，请将其 DCOM 身份验证级别设置为“无”，以允许匿名连接。如果不使用 OPCEnum.exe，则不需要为匿名用户启用远程访问。
6. 单击**应用**保存您的更改。
您必须随后编辑启动和激活权限。
7. 在“缺省 COM 安全性”分页标签的**启动和激活权限**部分中选择**编辑限制**。

在**组或用户名**列表中，选择**每个人**，然后选中**允许**列中的所有复选框。

8. 单击**确定**保存您的更改。

最后，您就可以编辑参与 OPC 通信的每个用户或组的**访问**和**启动**的缺省权限（例如“OPC 用户”）。

9. 在“COM 安全性”分页标签的**访问权限**部分，单击**编辑缺省值**。选择所需的用户/组并允许**本地访问**和**远程访问**。单击**确定**保存您的更改。

10. 在“COM 安全性”分页标签的**启动和激活权限**部分，单击**编辑缺省值**。选择所需的用户/组并允许**本地访问**和**远程访问**。单击**确定**保存您的更改。

到现在就完成了 OPC 客户机的配置。

一些比较旧的 OPC 客户机无法找到客户机计算机上的 Vijeo Citect OPC 服务器。如果出现这种情况，请将位于 bin 目录中的 ctopc.reg 文件复制到客户机计算机中。双击此文件以便将 OPC 服务器的特定数据添加到注册表中。

注意：现在您必须重新启动此计算机以使设置生效。

创建数据组

从 OPC 服务器访问数据之前，需要定义数据所属的数据组。数据组是将从 OPC 服务器获取的数据组织成相关数据的逻辑选择的一种方法。

注意：OPC 客户机通常允许按组设置属性，如收集数据的速率。因此，应根据这些要求考虑组的安排和命名。

由于此过程特定于 OPC 客户机，因此应参考 OPC 客户机随附的文档。

向组中添加数据项

在每个组中，可以分配一个或多个数据项。可以选择在 Vijeo Citect 运行时中定义的每个变量标签作为数据项。

由于此过程特定于 OPC 客户端，因此应参考 OPC 客户端随附的文档。

故障排除

如果您遇到 OPC 服务器、OPC 客户机或两者的连接问题，请参见 Vijeo Citect 知识库。

Chapter: 35 构建工程

本节介绍如何构建 Vijeo Citect 工程。

另请参见

[编译工程](#)

[运行系统](#)

[通过互联网运行系统](#)

[使用备用INI文件](#)

[调试运行系统](#)

[调试 I/O 设备和协议](#)

[在线重新启动系统](#)

[Vijeo Citect 软件保护](#)

[使用Vijeo Citect内核](#)

[收集运行信息](#)

编译工程

Vijeo Citect 编译器结合使用工程的配置数据库、图形和 Cicode 文件等元素，以创建运行系统。

编译检查项目错误并优化系统，以加快速度并提高操作有效性。工程编译时间取决于工程大小和用户计算机的速度。编译通常只需几分钟。

Vijeo Citect 编译器运行时，它通常以独占模式打开所有文件。在此模式下，只有 Vijeo Citect 具有文件访问权（编译器运行期间）。这提高了编译器的性能，但如果两个人同时编译不同文件同样会导致编译不完整，而这些工程有一个或多个共同的包含工程。

[General] ShareFiles 参数告知编辑器在共享模式下打开所有文件。这个选项允许共享网络用户同时运行编译器，但会增加编译所需时间。

编译工程：

1. 选择工程编辑器。
2. 单击**编译**，或选择**文件 | 编译**。

编辑完成时，结果将显示在一个对话框中。

Citect 编译器结果对话框单独报告错误数和警告数，以避免混淆。即使编译可能会生成警告，但 Vijeo Citect 工程仍可以成功编译，但是，编译错误将导致工程不能成功编译。

如果有编辑错误，必须先将其解决，然后再重新编译。

Vijeo Citect 会在您尝试运行它时自动编译工程（如果尚未编译）。

另请参见

[新增部分编译](#)

[调试编译](#)
[编译错误属性](#)
[编译错误消息](#)
[设置工程编辑器选项](#)

增量编译

用户可以只编译工程的新增部分。利用增量编译，Vijeo Citect 只编译上次编译后增加（或更改）的数据库记录。工程的其余部分不再重新编译。

注意：某些数据库记录是与其他数据库记录相关的。如果更改了相关记录，Vijeo Citect 就会编译整个数据库。

在现场运行一个系统前，用户需进行一次完整的编译（关掉增量编译），这种编译更加严格。与之类似，从软盘重新恢复一个工程时，第一次用户需进行一次完整的编译（关掉增量编译）。

转换成增量编译：

1. 选择工程编辑器。
2. 选择工具 | 选项。
3. 选中**增量编译**复选框，然后单击**确定**。

另请参见
[调试编译](#)

调试编译

如果编译器在编译过程中检查到错误，错误将被写入错误文件。编译器会在编译时向您通告所有错误，您可在任何阶段选择取消编译。如果有多个错误或有严重错误，编译器可能自动取消编译。编译结束后，您可以查找每一编译错误并显示相关信息。

编译器不会验证工程的操作。只因为工程编译并不意味它会在运行时正确工作。例如，编译器检查使用的标签是否被正确定义，以及 Cicode 是否具有可接受的语法。但它并不检查标签是否进行了正确缩放，或 Cicode 是否具有潜在的被零除错误。



警告

不希望的设备操作

- 在工程编译过程中报告的错误被全部解决完之前，请勿尝试运行系统。
- 在将工程提升到测试或实际环境之前，始终应执行完整的编译。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

查看编译错误:

- 选择“工程编辑器”，然后选择文件 | 编译错误。

要获得有关错误的详细信息，请执行以下操作:

1. 点击“编译错误”对话框底部的帮助。
2. 阅读与错误相关的帮助主题。

要查找错误（在工程中）:

- 点击“编译错误”对话框底部的进入。

另请参见

[编译错误属性](#)

编译选项

编译器有多个可用于帮助简化编译问题解决过程的选项。通常，这些选项用于修改编译错误日志的输出，提高其在访问特定错误类型和条目方面的实用性。

此处所列选项可通过 [工程编辑器选项对话框](#)（可从工程编辑器的工具菜单访问）进行调整。

- **编译期间记录“标签未定义”警告**

如果选择此选项，编译器将针对被检测到未在变量数据库中定义的任何标签在错误日志中生成一个“标签未定义”警告。

由于 Vijeo Citectv7.10 现在允许在图形画面上加入未定义的标签，此警告可能是多余的非实用警告。通过取消选中此选项，警告仍将包括在显示的警告计数中，但不会被添加到错误日志中。

- **记录编译期间的不建议使用警告**

如果选择此选项，编辑器将生成一个警告，用以识别在工程中检测到的不建议使用的元素，即不再支持的函数、参数或内核命令。

通过取消选中此选项，警告仍将包括在显示的警告计数中，但不会被添加到错误日志中。

- **就全编译期间未使用的标签发出警告**

通过此选项，可以针对未在 Vijeo Citect 工程中直接使用的标签生成警告条目。当全编译运行时，警告条目包含在编译错误表中。默认情况下，此选项不会被选中。

另请参见

[设置工程编辑器选项](#)

编译错误属性

可能会在编译 Vijeo Citect 工程时遇到错误。可以在“工程编辑器”中使用“编译错误”对话框查看编译错误。

编译器错误具有以下属性:

属性	描述
类型	错误类型。编译时可能出现三种类型的错误。它们是： 错误 - 编译进程继续，但在您纠正错误前，工程不能成功编译。 致命错误 - 此错误的严重程度可导致编译进程终止。纠正错误前无法编译工程。 警告 - 错误没有严重到阻止工程成功编译的程度，但你应该研究并纠正此错误。
记录	已经发生错误的数据库记录数。
名称	在其中检测到错误的图形画面的名称、库或报表格式文件。
域	已经发生错误的数据库域。
表格	已经发生错误的数据库表。
错误	错误简述。
上下文	在数据域、报表格式文件或 Cicode 中检测到错误的位置。错误的上下文包含在大括号 {...}。

另请参见
[编译错误消息](#)

编译错误消息

在工程编译期间，可能会看到下述提醒信息。

错误消息	描述
地址边界错误	当从 I/O 设备的内存中读长型或实型时，所有地址必须是奇型或偶型边界。地址不能混合。可以使用 [General]CheckAddressBoundary 参数禁止检查。
不支持模拟地址	DIGITAL 标签的预期位置定义了 INT 或其它模拟变量。检查标签名是否正确，或是否为该标签定义了 DIGITAL 数据类型。
数组尺寸超界	为标签指定了索引但未将其声明为数组，将标签声明为数组时未指定索引，为数组指定了错误的维数，或为一个数组指定了四个以上的维数。
错误的模拟格式	为模拟变量指定了错误格式。检查变量标签表的格式域。
错误的因数定义	使用包含无效表达式的 Cicode 表达式。检查表达式语法。
错误的浮点值	无法在预期的浮点值位置找到浮点值，或浮点值超界。
错误的 I/O 设备变量	变量标签地址的格式对于所使用的 I/O 设备协议无效：检查地址格式。（请参见各种受支持 I/O 设备的帮助中的 数据类型 主题，获取该设备的相应地址格式列表）。
错误的整型值	无法在预期的整型值位置找到整型值，或整型值超界。
错误的点限制	在 citect.ini 文件中指定了不正确的点限制。点限制必须符合您的 Vijeo Citect 许可证。

错误消息	描述
错误的原始数据类型	指定了无效的原始数据类型或不匹配的数据类型，例如，试图将整数转换为字符串。
错误的字符串参数	在字符串转换中指定了无效的格式参数。检查变量标签表中变量的格式说明。
不能编译所有函数	编译函数库时检测到错误消息。
不能打开文件	文件不能打开。文件不存在或已被破坏，或你的系统没有文件句柄可以利用。
不能读文件	文件不能被读。文件已经到结尾，或它已经被破坏。
不能从空函数中返回参数值	不能在返回参数值的函数中使用 RETURN 语句。删除 RETURN 语句或为函数声明一个返回数据类型。
函数内部不能使用数组	不能在函数内声明数组。数组只能被声明为库变量，即在库文件开始处。
函数体外不能使用 RETURN	只能在函数内使用 RETURN 语句。
不能写入文件	文件不能接受写操作。文件已经被破坏或磁盘已满。
达到 Cicode 数据限制	Cicode 模块中的数组不能超过 60KB。减小数组。
结束括弧缺失	Cicode 语句包含的开始和结束括弧数不等。应向语句中添加一个结束括弧“) ”或”]”。
关闭注释分隔符缺失	以 /* 开始的注释必须以 */ 结束。添加 */ 分隔符，或使用以感叹号 (!) 开始但不包含结束分隔符的分行注释。
结束引号缺失	Cicode 语句中包含不同数量的开始和结束引号。应向语句中添加一个结束引号 (") 。
数据库为空	数据库中没有记录。
数据库找不到	主工程数据库找不到。
数据库表满	数据库已满。如果错误仍然存在，请联系 Citect 支持。
磁盘满	磁盘已满。从磁盘中删除多余的文件，或用较大的磁盘替换现有磁盘。
DO 缺失	WHILE 语句中必须包含 DO 语句。
END 缺失	END 语句必须用在条件语句或函数定义的末尾。
读文件时出错	无法找到或打开在数据库域中指定的包含文件。检查文件名是否正确以及是否已正确指定了文件，即 <@FILENAME>。
表达式太大	表达式对于编译器而言太大。通过划分表达式为两个或更多小表达式来减小表达式尺寸。
文件已在单一模式下打开	文件已经被另一用户打开。将 citect.ini 文件中的 [General] ShareFiles 参数设置为 1，以在共享模式下打开文件。
文件不存在	找不到文件。检查文件名是否正确。
文件只读	曾试图向只读文件中写入内容。检查文件名是否正确或更改文件的属性。

错误消息	描述
文件锁死	文件被其他网络用户使用。
文件没索引	确定数据库已被编入索引，但找不到与数据库关联的索引文件。压缩数据库。
文件大小错误	Cicode 函数文件、报表格式文件或包含文件太大。文件最大不能超过 1MB 。
FUNCTION 缺失	必须用 FUNCTION 关键字声明函数。
不接受 GLOBAL 函数，请使用 PUBLIC	已在 Cicode 中声明了全局函数。全局不是有效的函数类型。必须使用“公共”类型取代之。
群组未发现	群组名缺失。检查群组名是否正确，或是否已在群组表中正确指定了群组。
未发现包含工程	包含工程（在包含工程数据库中指定）不存在。检查包含工程的名称。
不兼容的类型	语句中包含不匹配的数据类型。例如，在数字的预期位置指定了字符串。 在工程或包含工程中，此错误可能会在变量标签被放置在图形画面上时以及标签的名称与 Cicode 函数的名称相同时出现。
函数的参数个数不正确	传递到 Cicode 函数的参数太多或太少。
索引关键字已改变	数据库索引已破坏。压缩数据库。
无效的布尔值	在预期的 TRUE 或 FALSE 值位置发现非整型值。例如， IF 、 WHILE 或 FOR 语句中的控制表达式必须是整数。
无效的字体名	工程中不存在该字体。检查字体名是否正确，或通过字体表指定字体。
无效的群组定义	工程中不存在群组。检查群组名是否正确，或通过群组表指定群组。
无效的时间格式	在报表、事件、趋势标签、 SPC 趋势标签或设备表的时间、周期或采样周期域中错误地指定了时间。 时间必须采用 HH:MM:SS 格式，并且必须处于 0:00:00 - 23:59:59 范围内。只需要小时，例如， 16 表示 16:00 （下午 4:00）。请注意，将出于历史目的接受 24:00:00 ，并将其直接映射到 0:00:00 。 周期格式必须是有效日期或 HH:MM:SS 格式的时间，分钟和秒的范围是 0-59 。只需要秒，例如， 22 表示 22 秒。 采样周期格式必须是毫秒值（例如， 0.200 表示 200 毫秒）或 HH:MM:SS 格式的时间（分钟和秒的范围是 0-59 ）。只需要秒，例如， 22 表示 22 秒。
标记参数错误	参数语法不正确，指定了错误的参数个数，或参数中的字符数不正确。
标记被定义两次	标记名必须是唯一的。检查标记表是否有重复的名称。
标记太大	标记太大。标记的大小不能超过 8KB 。
超过最大报表尺寸	报表文件必须少于 63KB 。减小报表的尺寸或配置两个报表。
不接受 MODULE 函数，请使用 PRIVATE	在 Cicode 中声明了模块函数。模块不是有效的函数类型。必须使用私有类型取代之。

错误消息	描述
必须返回来自函数的参数值	如果已声明 Cicode 函数将返回参数值，它必须包含 RETURN 语句。
没有定义的 I/O 设备	未在工程中定义 I/O 设备。
没有数据库格式	数据库已经被破坏或文件格式未知。压缩数据库。
开始括号缺失	必须在 Cicode 函数中使用圆括号 ()，即使在其没有参数时也如此，例如 MyFunction() 。在工程或包含工程中，此错误可能会在变量标签被放置在图形画面上时以及标签的名称与 Cicode 函数的名称相同时出现。
操作数缺失	Cicode 操作符必须跟着一个操作数。
没有可利用的文件句柄	<p>Vijeo Citect 使用文件句柄打开每一个文件。当试图同时打开多个文件或数据库时，Vijeo Citect 可能需要比可用句柄更多的文件句柄。</p> <p>如果您有许多包含工程，则很有可能没有文件句柄可用。在编译工程时，Vijeo Citect 会同时打开每个包含工程中的数个文件，以增加包含的每一额外工程的文件句柄利用率。如果您在添加其他包含工程后得到此提醒信息，就没有可利用的文件句柄了。要确认此问题，请删除包含工程之一，然后查看 Vijeo Citect 能否编译您的工程。</p> <p>对于联网运行的 Windows，文件句柄数设置位于不同的位置。要增加 DOS 中的文件句柄数，设置位于 CONFIG.SYS 文件中。如果使用的是 Novell Netware，您必须增加 NET.CFG 或 SHELL.CFG 文件中的文件句柄数。还必须通过 [CtEdit]DbFiles 参数增加 Vijeo Citect 使用的文件句柄数。调整相关文件的以下设置：</p> <pre> CONFIG.SYS FILES=120 NET.CFG or SHELL.CFG file handles=120 </pre>
没有可利用的内存	Vijeo Citect 已经没有可利用的内存。增加计算机中的内存空间或使用较小的数据库。
画面名称不能以下划线开始	画面名称必须以字母数字字符 (A - Z, a - z 或 0 - 9) 开始。
达到点限制	已经达到可以参考的最大点数。最大点数限制由 Vijeo Citect 许可证决定。联系 Citect 支持。
不接受 PRIVATE 变量，请使用 MODULE	在 Cicode 中声明了私有变量。私有不是有效的变量类型。必须使用模块类型取代之。
协议缺失	I/O 设备数据库中的协议域是空的。您必须为 I/O 设备选择一个协议。
不接受 PUBLIC 变量，请使用 GLOBAL	您在 Cicode 中声明了公共变量。公共不是有效的变量类型。必须使用全局类型的变量取代之。
已达到表底	已达到数据库表底，或数据库已被破坏。压缩数据库。如果错误仍然存在，请联系 Citect 支持。
此变量不支持读再映射	不能在再映射写被禁止时写入映射变量，不能在再映射读被禁止时读取。检查再映射表。
分号缺失	所有 Cicode 语句必须用分号 (;) 分隔。

错误消息	描述
软件错误	检测到 Vijeo Citect 内部软件错误。联系 Citect 支持。
定义文件无效	系统文件已被破坏或被删除。在系统中重新安装 Vijeo Citect。如果错误仍然存在，请联系 Citect 支持。
语句缺失	Vijeo Citect 缺失了一个语句。检查 Cicode 是否有语法错误。
字符串缺失	数据库域中只能使用字符串。
字符串太大	字符串超出大小限制。字符串最多不得超过 255 个字符。
超级精灵必须在画面上	超级精灵语法 (?) 只能在画面上使用。不能在报表或 Cicode 函数库中使用超级精灵。使用 TagRead() 和 TagWrite() 函数取代之。
搜索符号失败	数据库记录不存在。请检查记录名是否正确。
语法错误	指定了错误的 Cicode 表达式。检查表达式的结构。
标签已经定义	标签名必须是唯一的。检查变量标签表是否有重复的名称。
标签缺失	未在预期的位置找到标签名称，或已将表达式传递至缺少标签的函数。检查标签名是否正确，或通过变量标签表指定标签。
没有发现标签	标签不存在。检查标签名是否存在，或指定标签（通过变量标签表）。如果变量数据库中没有标签，数据库的索引可能不正确。如果使用 Excel 或其它数据库编辑器编辑变量数据库，可能会出现该错误。要为数据库重新编制索引，请选择文件 压缩（在工程编辑器中）。
THEN 缺失	必须在 IF 语句中使用 THEN 语句。
参数过多	在 Cicode 函数中指定了太多参数。最多可以接受 32 个参数。
Cicode 函数过多	定义了超过 4500 个用户函数。要增加允许的函数数目（最多 10000 个），请使用 CtEdit 参数 MaxCicodeFunctions。此错误通常是在多个包含工程中定义 Cicode 函数引起的。扩展该参数可能影响系统性能。仅应根据 Citect 客户服务的建议对其进行设置。
数据库中有太多域	在数据库中指定了太多域。此错误应该仅会在 citect.frm 文件被更改或被破坏后出现。联系 Citect 支持。
打开的文件过多	已经超过可同时打开的最大 .DBF 文件数。请通过更改 [CtEdit] DbFiles 参数增加限制。
太多包含工程	已经定义了超过 240 个包含工程。
数据库中的记录过多	数据库中指定了太多记录。此错误应该仅会在 citect.frm 文件被更改或被破坏后出现。联系 Citect 支持。
Cicode 中有拖尾字符	Cicode 语句中有后跟分号的多余拖尾字符。
名称中有拖尾字符	数据库记录名中包含无效字符。从记录名中删除无效字符。
未预料的文件开始	已经到达数据库开始处或数据库已被破坏。压缩数据库。如果错误仍然存在，请联系 Citect 支持。

错误消息	描述
未预料的文件结尾	已达到数据库结尾处，或数据库已被破坏。压缩数据库。如果错误仍然存在，请联系 Citect 支持。
未知的二进制错误	编译时无法打开输出文件。压缩数据库。如果错误仍然存在，请联系 Citect 支持。
未知的 DBA 错误	检测到 Vijeo Citect 内部软件错误。联系 Citect 支持。
未知域	正在引用不存在的域。数据库已被修改或被破坏。压缩数据库。如果错误仍然存在，请联系 Citect 支持。
未知文件	找不到包含文件。检查包含文件名是否正确，或包含的文件是否位于正确的目录中。
未知 I/O 设备	工程中没有 I/O 设备。检查 I/O 设备名是否正确。
未知协议	协议不存在。

运行系统

对工程进行编译后，可以启动运行时系统。运行系统之前，先运行计算机设置向导。

注意：请记住，Vijeo Citect 软件是受版权保护的。如果尝试运行没有保护键的 Vijeo Citect，Vijeo Citect 会显示提醒信息，用户只能在演示模式下运行。

另请参见

[启动和运行时配置](#)

[独立运行服务器](#)

[通过 Internet 运行系统](#)

启动和运行时配置

可指定要在 Vijeo Citect 启动时自动执行的 Cicode 函数。此 Cicode 会在 Cicode 系统上线时尽快执行。应使用计算机设置向导指定启动函数的名称。

还可以在启动时运行报表。Vijeo Citect 会在它启动时搜索名为“Startup”的默认报表。如果已经配置了名为“Startup”的报表，它会自动运行。您可以使用计算机设置向导更改启动报表的名称（或将其全部禁止）。

可以自定义 Vijeo Citect 运行时和启动行为的多个元素。通常，计算机设置向导已可以满足全部需要，但您也可以使用参数进行更多控制。

要启动 Runtime 系统：

- 点击运行，或选择文件 | 运行。

要在线编译和运行 Vijeo Citect：

- 点击**运行**，或选择**文件 | 运行**。

注意： Vijeo Citect 会在您尝试运行它时自动编译工程（如果尚未编译）。

如果您的操作系统是 Windows Vista，Vijeo Citect 会在运行时运行一次兼容性检查，以查看运行工程使用的驱动程序是否与您的操作系统兼容。

- 如果某 I/O 设备使用标记为与 Windows Vista 兼容的驱动程序，工程将正常地继续运行。
- 如果某 I/O 设备使用标记为与 Windows Vista 不兼容的驱动程序，会显示一个对话框，要求您从工程配置中删除此 I/O 设备或在兼容操作系统中运行此 I/O 服务器。
- 如果某 I/O 设备使用尚未被确认为是否兼容的驱动程序，会显示一个对话框，询问您是否继续运行该工程。

独立运行服务器

可以在多进程系统中运行各个进程，这样特定的服务器或客户机可以单独启动。此特性对硬件测试和系统分析很有用。

通过 **Citect 浏览器**、**图形编辑器**和**工程编辑器**中的**工具**菜单配置工程期间，可以启动运行时管理器。通过这种方式启动运行时管理器表示它处于空闲状态，这样用户可以自行启动特定进程。

启动进程（通过右键单击进程并选择**启动**）后，运行时管理器的监视板将显示选定进程的启动日志并启动运行时。

注意： 只有一个运行管理器的实例可以在计算机中的任意时刻运行。

有关使用运行时管理器的详细信息，请单击其**帮助**按钮。

另请参见
[服务器重定向](#)

通过 Internet 运行系统

如果能够通过计算机进行 Internet 访问，则可以使用计算机通过 Internet 从远程位置运行工程。此计算机然后被称为“Internet 显示客户端”。这基本上只适合运行时 Vijeo Citect 版本；可以从该计算机运行工程，运行过程与从任何正常客户机的运行过程一样。但 Internet 显示客户端不能是服务器，不能用于对配置进行更改，只能运行工程。

Windows Vista 不支持 Internet 显示客户端。它只能在早期操作系统中正确运行，但已 Vista 和以后更高的版本也不打算提供对 IDC 的支持。

注意： Vijeo Citect 还允许用户通过 LAN 连接计算机的网络在标准 Web 浏览器中运行工程。请参见 [Vijeo Citect Web 客户机](#)。

另请参见

[Vijeo Citect Internet 显示客户机](#)

[Vijeo Citect Internet 服务器](#)

[启动和运行时配置](#)

[服务器 - 客户机文件更新](#)

互联网显示客户端

互联网显示客户端是 Vijeo Citect 的一个仅运行期间版本。互联网显示客户端不能是服务器，并且不能用来进行配置更改 - 您只能用它来运行工程。

一台计算机可以在正常安装 Vijeo Citect 的同时安装互联网显示客户端。通过互联网运行工程之前，必须启用 **[Internet]Client** 参数。

还可以同时运行互联网显示客户端的多个实例。这使得用户可以在远程计算机上同时处理多个工程，但只能在运行时模式下。

注意：

- Windows Vista 不支持互联网显示客户端。它仍将在较早的操作系统上正常运行，不过，尚无将来在 Vista 及更高版本操作系统上对 IDC 提供支持的计划。
- 如果用户使用防火墙，必须确认端口 2073 到 2079 是畅通的，以使互联网显示客户端能和互联网服务器通讯。
- 如果 ActiveX 对象具有关联的数据源，则需要确保承载互联网显示客户端的计算机能够定位该数据源。参见主题 [管理关联数据源](#)。
- 如果要通过 IDC 使用过程分析器，必须将过程分析器视图 (.pav) 文件复制到 IDC 才能实现此目的。

另请参见

[Vijeo Citect 互联网服务器](#)

互联网服务器

任何 I/O 服务器均可以是 Vijeo Citect 互联网服务器：只需要使用计算机设置向导。（互联网服务器需要特殊的保护密钥。有关保护密钥的详细信息，请联络 Citect 支持部门。）

如果用户使用防火墙，必须确认端口 2073 到 2079 是畅通的，以使互联网显示客户端能和互联网服务器通讯。

注意： Windows Vista 不支持互联网显示客户端。它仍将在较早的操作系统上正常运行，不过，尚无将来在 Vista 及更高版本操作系统上对 IDC 提供支持的计划。

另请参见

[启动和运行期间配置](#)

启动和运行时配置

可以通过计算机设置向导自定义 Vijeo Citect 运行时和启动行为的许多元素。

还可配置 IDC 安装，以在启动时连接到 Vijeo Citect 互联网服务器。为此，您需要调整 IDC 计算机的 citect.ini 文件中的参数 [Internet]IPAddress、[Internet>Password 和 [Internet>ShowSetupDlg。

citect.ini 文件保存在互联网显示客户端安装的 bin 目录下。

注意：Windows Vista 不支持互联网显示客户端。它仍将在较早的操作系统上正常运行，不过，尚无将来在 Vista 及更高版本操作系统上对 IDC 提供支持的计划。

另请参见
[服务器 - 客户端文件更新](#)

服务器 - 客户端文件更新

当用户登录 Vijeo Citect 互联网服务器时，所有用来运行工程的文件下载到用户计算机。因为这些文件必须是最新的，Vijeo Citect 定期把互联网服务器上的文件与下载到互联网显示客户端上的文件进行比较。（此周期可通过参数 [Internet]UpdateTime 定义）。如果某个文件自上一次更新以来被改变，它就被复制到互联网显示客户端。

注意：Windows Vista 不支持互联网显示客户端。它仍将在较早的操作系统上正常运行，不过，尚无将来在 Vista 及更高版本操作系统上对 IDC 提供支持的计划。

设置用户 Vijeo Citect 互联网服务器：

1. 运行计算机功能配置向导，选择**自定义设置**。
2. 从**网络计算机**部分中选择**服务器和控制客户端**。
3. 在到达互联网服务器屏幕后，选择**互联网服务器**。
4. 输入互联网服务器计算机的 TCP/IP 地址（例如，10.5.6.7 或 plant.yourdomain.com）。在建立一个连接时，这些信息被下载并存储在互联网显示客户端的 citect.ini 文件中。
5. 确定互联网服务器计算机的 TCP/IP 地址：
 - 对于 Windows NT4 或 2000，转到命令提示，输入 **IPCONFIG**，然后按回车。
 - 对于 Windows 95，选择**开始|运行**，键入 **WINIPCFG**，再按下回车（**[Enter]**）键。
6. 完成计算机设置向导后，使用 [Internet]Manager 和/或 [Internet]Display 参数，定义互联网显示客户端使用者必要的密码。
7. 如果互联网服务器上的运行时间工程与任何包括工程有链接，如果包括工程文件和运行时间工程文件存储在相同目录层，互联网显示客户端只能访问包括工程文件。例如，如果当前工程是在目录：

C:\Citect\User*<current project>*

那么任何所包含的工程必须在同级:

C:\Citect\User*<包含工程>*

- 任何可被潜在的 FTP 用户访问的文件需放置在互联网服务器的 Internet 目录, 默认 \Internet 位置是 C:\Citect\User\Internet。这是 Vijeo Citect 保存 idc.exe 文件的位置, 允许远程安装互联网显示客户端。

如果互联网目录和当前运行时间工程共享相同目录层, 互联网服务器上的 Internet 目录只能被潜在 FTP 用户访问。例如, 如果用户使用 Vijeo Citect 的默认设置, 当前工程文件夹将在:

C:\Citect\User*<current project>*

并且此 Internet 目录会在同级:

C:\Citect\User\Internet

如果互联网服务器上的运行时间工程存储在其它地方, 必须建立一个合适的互联网目录。

安装互联网显示客户端:

- 在远程计算机上, 启动互联网浏览器。
- 键入 Vijeo Citect 互联网服务器 (例如 <ftp://sanctus.citect.com.au/idc.exe>) 的 FTP 地址。
- 将文件 (IDC.exe) 保存在一个临时文件夹。
- 进入这个临时文件夹, 双击 **IDC.exe**。
- 按照提示完成安装。

通过互联网运行工程:

- 确认用户安装了互联网显示客户端并设置了互联网服务器。
- 在互联网显示客户端, 双击 Vijeo Citect IDC 程序组的 **Citect 运行** 图标。
- 在 **Citect 互联网客户端设置** 对话框中, 键入 Vijeo Citect 互联网服务器的 TCP/IP 地址 (例如 10.5.6.7 或 plant.yourdomain.com), 然后键入口令。
- 点击 **确定**。所有相关数据将下载到用户计算机, 工程会运行 (如果 Vijeo Citect 互联网服务器在运行)。

一旦工程在互联网显示客户端运行, 要连接某个不同的 Vijeo Citect 互联网服务器。

- 单击运行时间 **控制** 菜单中的 **客户端设置**。
在 **Citect 互联网客户端设置** 对话框里, 完成下列属性:

选项	描述
地址	键入 Vijeo Citect 互联网服务器的 TCP/IP 地址 (例如 10.5.6.7 或 plant.yourdomain.com)。最后一次使用的互联网服务器地址会自动输入。先前使用的所有互联网服务器的地址保留在菜单中 (最近使用的在顶部)。
口令	输入由 Vijeo Citect 互联网服务器管理员提供的口令。用户的口令在传送之前被加密。 如果用户输入一个不正确的口令, 连接失败。

2. 点击**确定**。所有相关数据下载到用户计算机，工程会运行。

使用备用 INI 文件

Vijeo Citect 启动时，会从 `citect.ini` 读取缺省参数值。从 v7.10 起，Vijeo Citect 预定将 `Citect.ini` 文件存放在安装期间选择的 User 的 `config` 文件夹和 `Data` 文件夹中。如果在该位置未找到文件，将不会搜索其它位置，而显示错误。如果需要将 INI 文件存储在其它位置，则应在启动 `citect32.exe` 和 `ctexplor.exe` 时在命令行中指定存放路径。如果要运行多个工程，则可以为每个工程指定指定备用 ini 文件。新的 ini 文件名必须传递给 Vijeo Citect 系统应用程序，即作为命令行参数传递给 `CtExplor.exe` 和 `Citect32.exe`。

如何为 Vijeo Citect 指定备用 INI 文件：

1. 单击用于启动 Citect 工程管理器或运行时的图标。
2. 单击右键显示快捷菜单，选择**属性**。
3. 单击**快捷方式**标签。
4. 使用 `-i` 选项将 INI 文件的名称添加到适当图标的命令行属性。例如，要使用初始化文件 `my.ini` 启动 Vijeo Citect，请输入以下行：

```
C:\Program files\Schneider Electric\Vijeo Citect\Vijeo Citect 7.10\bin\citect32.exe  
-ic:\citect\user\myproj\MY.INI
```

注意：计算机设置向导使用的 `.ini` 与为 Citect 工程管理器指定的相同。可以为运行时和浏览器程序指定不同的 ini 文件。但如果从浏览器初始化运行时，则使用为浏览器指定的 `ini`。

调试运行期系统

本节介绍用于分析和解决系统级错误的工具。

另请参见

[硬件报警](#)

[SysLog.DAT 文件](#)

[调试 I/O 设备和协议](#)

硬件报警

当检测到系统错误时（即影响到 Vijeo Citect 操作的故障），Vijeo Citect 生成硬件报警。硬件报警通常在专用硬件报警画面中显示，可用作标准模板。

硬件报警画面指示 Vijeo Citect 系统中正在发生什么。在发生通信丢失、Cicode 无法执行、图形画面不能正确更新或服务器出现故障时，此画面显示。硬件报警由唯一的描述和错误代码构成。

硬件报警没有详细信息，但可为您指出正确的方向。例如，如果您得到动画冲突报警，Vijeo Citect 不会告知您原因。您必须观察哪个画面引起了此硬件报警，并自行确定此动画的位置。

注意：您的系统应该没有连续的硬件报警。

以下两个硬件报警域并不总显示在硬件报警画面中。ERRPAGE 显示在检测到错误时显示的画面的名称。这对查找由动画不正确编程而引起的错误很有帮助。ERRDESC 提供特定于报警类型的信息。例如，如果报警是一个 I/O 设备错误，ERRDESC 会显示此设备的名称。

另请参见：

[SysLog.DAT 文件](#)

SysLog.DAT 文件

syslog.dat 是由 Vijeo Citect 维护的文件，包含一个有用的 Vijeo Citect 系统信息记录。记录到 syslog.dat 的信息是广泛的；从底层驱动程序流量和核心消息到用户定义的消息。

“I/O 设备属性”对话框中的**记录读**和**记录写**域控制是否记录每一个 I/O 设备。

注意：Vijeo Citect 在运行期间锁定 syslog.dat。然而，您仍可以使用内核中的 SysLog 命令来查看它。

这个文件的大小是有限制的（缺省最大值为 2000 kb）。当该文件达到大小限制时，Vijeo Citect 会重命名 syslog.bak，并启动一个新的 syslog.dat。用户可通过参数 [Debug]SysLogSize 改变这个限制。例如，citect.ini 中的以下行会将 syslog.dat 的大小设置为 30 Mb：

```
[DEBUG]
SysLogSize=30000
```

如果要归档多个系统日志文件，可以将 [Debug]SysLogArchive 设置为 1。这将向文件名中添加一个时间标签。

另请参见

[调试 Runtime 系统](#)

使用地址转发进行服务器重定向

Vijeo Citect 允许使用者临时转换服务器的通讯到另一个计算机上，这样会帮助硬件维护和系统分析。

通过在 `Citect.ini` 文件内包含地址转发项，使用者可以覆盖服务器的工程配置地址，重定向互联网通信量到不同的地址和端口。

例如，如果在项目配置里添加某个新服务器之前用户要测试此新服务器，用户可以在单一的客户机的 `citect.ini` 文件内进行必要的地址转发调整以定向到新的硬件。此服务器不必重新编译工程就能在系统环境里被测试。

使用者必须在所有可能被地址转发影响的计算机的 `citect.ini` 文件里进行必要的调整。

例如，如果计算机配置以运行两个 I/O 服务器，并且使用者可能会临时重定向其中的一个，然后服务器和所有正在连接的客户机必须得到对它们的 `citect.ini` 文件的必要地址转发调整，这样服务器才能知道彼此。

使用 `citect.ini` 文件里的下列语法来实现地址转发：

```
[AddressForwarding]
<ClusterName>.<ServerName>=<ipaddress>:<port>
```

使用者也可以使用下列语法来重定向 I/O 服务器的对等端口连接：

```
[AddressForwarding]
<ClusterName>.<ServerName>_PeerPort=<ipaddress>:<port>
```

对于启用报警属性的报警服务器，可以使用下列方法重定向报警属性连接器：

```
[AddressForwarding]
<ClusterName>.<ServerName>_AlarmProps=<ipaddress>:<port>
```

只是在 `Citect Runtime` 启动期间解释和使用地址转发。推荐（不是必须的）在确认改变的工程运行之前运行计算机设置向导。



警告

不希望的设备操作

- 请勿将地址转发作为指定客户端/服务器通信的主要机制。它绕过由编译器强制执行的正常配置检查。
- 因为一旦重定向不再是必需的，调整必须手动改正，您应仔细跟踪您对 `citect.ini` 文件所做的每一个调整。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

注意： Vijeo Citect Web 客户机使用地址转发来使通信穿过公司的防火墙。任何手工的调整也能影响 Web 客户机与 Web 服务器部署的连接能力。

在线重新启动系统

使用在线重新启动工具，您可以更改工程配置，并检查运行时系统中的结果，而无需关闭 Vijeo Citect。系统运行时，您可以更新系统。

注意： 如果已将 Vijeo Citect 配置为使用多处理器支持，则建议不要使用重启工具，而使用运行管理器。运行管理器允许您分别重启不同的进程，而重启工具则会重启该机器上的所有 Vijeo Citect 进程。

系统转换所需要的时间是由工程的大小和工程变化的范围决定的：

- 如果用户只改变图形画面，Vijeo Citect 会执行 *部分重新启动*（只改变运行时系统的画面）。转换是瞬间的。
- 如果您更改任何数据库（例如，添加一个新的警报标签、趋势标签或 Cicode 函数），Vijeo Citect 将执行 *完全重新启动*来运行更新的工程。

另请参见

[在线重新启动网络上的系统](#)

在线重新启动网络上的系统

如果是在网络上使用 Vijeo Citect，可使用结构化的重新启动步骤在对工厂无负面影响的情况下重新启动 Vijeo Citect。用户可通过网络上的任何 Vijeo Citect 计算机来开始在线重新启动。

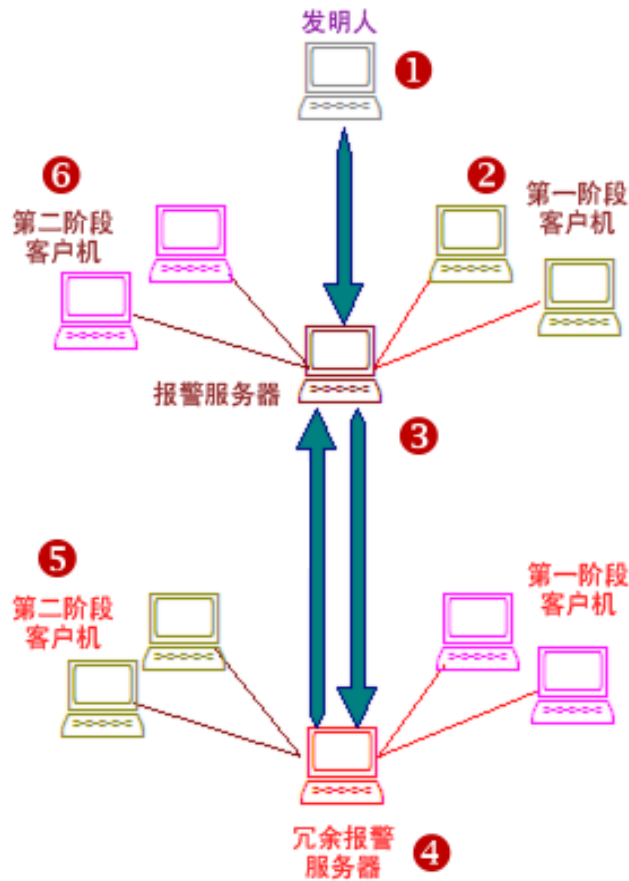
关闭的阶段：

1. 将每个客户机上的 citect.ini 参数 [Shutdown]Phase 设置为 1 或 2，以标识客户机的阶段。
2. 从服务器发出 Shutdown() 命令。
发出 Shutdown() 命令后，设置为阶段“1”的客户机将首先关闭。设置为阶段“1”的客户机重新连接到服务器后，设置为阶段“2”的客户机将关闭。

Vijeo Citect 通过以下步骤自动管理在线重新启动：

1. “创始人”发出 Shutdown("Everybody") 命令。
2. 创始人连接的报警服务器将关闭其第一个阶段客户机。
3. 然后，创始人连接的用于建议其他报警服务器的报警服务器将关闭其自身。
4. 第二个报警服务器关闭其第一个阶段客户机，并等到另一个报警服务器重新启动（运行新工程）。
5. 报警服务器重新联机后，第二个报警服务器关闭其第二个阶段客户机，然后关闭其自身。

6. 第一个报警服务器重新启动其第一个阶段客户机，并关闭和重新启动其第二个阶段客户机。



注意：如果已将 Vijeo Citect 配置为使用多处理器支持，则建议不要使用重启工具，而使用运行管理器。运行管理器允许您分别重启不同的进程，而重启工具则会重启该机器上的所有 Vijeo Citect 进程。

另请参见

[\[Shutdown\]Phase](#)、[\[Shutdown\]NetworkIgnore](#)、[Shutdown\(\)](#)

使用多个工程

使用在线重启工具的最有效方法是使用两个工程。当第二个工程处于发展阶段时，第一个工程成为当前运行时系统。可以按以下方式管理两个工程：



工程 A 是当前的运行时系统，而工程 B 正在开发阶段。



工程 B 完成时，可以使用在线重启工具将运行时系统更改为工程 B。



然后将工程 B 复制到工程 A 以便进一步开发。

注意： 如果已将 Vijeo Citect 配置为使用多处理器支持，则建议不要使用重启工具，而使用运行管理器。运行管理器允许您分别重启不同的进程，而重启工具则会重启该机器上的所有 Vijeo Citect 进程。

开始在线重新启动

开始在线重新启动，由发起人（网络上的任何 Vijeo Citect 计算机）发出一个关机命令和一个关机函数，例如：

```
Shutdown("Everybody", "MyProject", 2);
```

如有可能，平衡两阶段关机的所有客户端。[Shutdown]Phase 参数定义每个 Vijeo Citect 计算机回应的阶段。

用户可通过参数 [Shutdown]NetworkIgnore，将选定的计算机（例如，I/O 服务器）从在线重新启动过程中排除。

安全性方面，用户可通过参数 [Shutdown]NetworkStart，阻止选定的计算机发起在线重新启动过程。

注意：如果已将 Vijeo Citect 配置为使用多处理器支持，则建议不要使用重启工具，而使用运行管理器。运行管理器允许您分别重启不同的进程，而重启工具则会重启该机器上的所有 Vijeo Citect 进程。

使用回调函数

在系统关机前，用户可通过回调函数（和 OnEvent 函数）执行清理工作。通常在主要启动函数（使用 [Code]Startup 参数定义）中调用 OnEvent()。每一次调用 Shutdown()，就运行回调函数。

```
/* A user shutdown procedure. */
INT
FUNCTION
MyStartupFunction()
    ...
    OnEvent(25, MyShutdown);
    ...
END

INT
FUNCTION
MyShutdown()
    STRING sPath;
    // Perform housekeeping tasks
    ...
    sPath = ProjectCurrentGet();
    If sPath = "ProjectA" Then
        ProjectSet("ProjectB");
    Else
        ProjectSet("ProjectA");
    END
    Shutdown("Everybody", sPath, 2);
END
```

软件保护

Vijeo Citect 使用硬件密钥来防止许可证侵权。硬件密钥是插入到计算机的并行端口或 USB 端口的物理密钥。该硬件密钥包含用户许可证的详细信息，如类型和 I/O 点数限制。

另请参见

[CiUSAFE 对话框属性](#)

[演示模式](#)

CiUSAFE 对话框属性

CiUSAFE 对话框具有下列属性：

序列号

计算机硬件密钥的序列号。如果密钥是 2000 年 9 月 11 日之后交付的，或在此时间后更新的，会出现序列号。如果不是这种情况，用户可在硬件密钥的标签上获得序列号。用户需要在 Schneider Electric SA 网站输入序列号才能升级该密钥。

KeyID

用户每次启动 CiUSAFE 时，**KEYID** 字段中将显示一个密钥 ID。升级硬件密钥时，用户可能需要提供密钥 ID 和序列号。这取决于该密钥在 Vijeo Citect 许可证数据库中的状态，如需密钥 ID，会提示用户输入。单击**保存 KeyID** 将密钥 ID 和序列号保存到一个文本文件，用户访问 Schneider Electric SA 网站时，可参阅这个文件。

授权代码

要更新硬件密钥，请输入 106 个字符的授权代码。输入密钥 ID 和序列号并验证许可证和客户服务协议之后，会要求用户提供此代码。单击**更新**以更新硬件密钥。

返回代码

返回代码指出密钥更新的结果：

0	密钥更新成功。
1,3	输入的 KeyID 或授权代码无效。
2	输入的 KeyID 或授权代码被破坏。
4,16	输入的 KeyID 或授权代码无效。
9	找不到硬件密钥。

要关闭程序，请单击**退出**。

演示模式

您可以在演示模式下运行 Vijeo Citect，这样就不需要硬件密钥。在演示模式下，可以正常使用 Vijeo Citect 的全部功能，只不过有运行时间和 I/O 限制。

注意：如果将 Vijeo Citect 配置为在一个或多个 CPU 上运行多个进程，则不能以演示模式使用 Vijeo Citect。如果以一个进程运行 Vijeo Citect，可以跟以前版本的 Vijeo Citect 一样使用演示模式。

以下演示模式适用于：

- 15 分钟内最多有 50,000 个真实的 I/O。
- 10 小时没有静态点，最多允许有一个真实的 I/O。这对使用内存和磁盘 I/O 的演示非常有用。如果没有配置静态点，Vijeo Citect 将以此模式启动。

- 如果您希望以这种模式演示 DDE、CTAPI 或 ODBC 写入 Vijeo Citect，则只能写入一个点。要写入多个点，必须通过创建至少一个静态 I/O 点迫使 Vijeo Citect 以 15 分钟 50,000 个 I/O 演示模式启动。
为使此模式工作，您必须配置一个真实变量标签，并附随 PLC 或 I/O 设备。标签必需在 Cicode 中由画面使用。如果您没有连接真实的 I/O 设备，Vijeo Citect 会报告一个硬件错误，您可以使用 IODeviceControl 函数将它禁止。
- 8 小时最多 42,000 个真实的 I/O。这只能通过特殊的 Vijeo Citect 集成合作伙伴 (CIP) 密钥来提供。

使用内核

可使用 Vijeo Citect 内核执行低级诊断和调试操作，以及 Vijeo Citect 系统的运行时分析。使用它可显示所有低级数据结构、运行时数据库、统计、调试跟踪、网络流量、I/O 设备流量等。还可以调用内置 Cicode 函数或用户编写的 Cicode 函数。



不希望的设备操作

- 请勿将内核用于普通的 Vijeo Citect 操作。内核仅用于诊断和调试目的。
- 将安全性配置成只有经过批准的人员才能查看或使用内核。
- 除非您是 Vijeo Citect 和 Cicode 的专家用户，或者是在 Schneider Electric SA 支持人员的指导下，否则请勿查看或使用内核。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

应限制对内核的访问：使用内核的任何人都可以完全控制 Vijeo Citect（随后可控制工厂和设备）。

访问 Cicode 和缓存命令

为了防止内核中 Cicode 和缓存命令的未授权使用，只有内核用户能够访问这些命令。必须在用户数据库中为要从中访问命令的 Vijeo Citect 工程定义内核用户（用户名为“内核”，密码非空白）。内核用户不需要定义任何区域或权限。

运行期间，如果内核用户尝试访问 Cicode 或缓存内核命令，内核将向内核用户索要密码。如果没有在用户数据库中为该工程定义内核用户（或用户提供的密码不正确），则拒绝对 Cicode（或缓存）命令的访问。

另请参见

[显示内核窗口](#)

[内核内部](#)

[使用内核命令](#)

[内核命令](#)

显示内核窗口

您可以用多种方式显示内核窗口。Vijeo Citect 能够在启动时自动打开窗口，在“控制”菜单中提供命令选项，或者，您可以在需要时定义运行时命令，从而显示“内核”窗口。

- [通过控制菜单显示内核](#)
- [在启动时显示内核](#)
- [定义一个运行时命令](#)
- [关闭内核窗口](#)

通过控制菜单显示内核

要在运行时系统的“控制”菜单中添加**内核**选项，请使用“计算机设置向导”。运行向导，选择**自定义模式**，在“安全设置 | 控制菜单”页中选择**在菜单上的内核**选项。

然后，可以在运行时从“控制”菜单中（左上角）选择“内核”选项来显示“内核”窗口。如果没有显示标题栏，则可以通过按 **ALT-SPACE** 访问控制菜单（确认在“键盘”页的“安全设置”中选择了**已启用 Alt-Space**）。

注意：在试运行之后，请清除这些选项（缺省），以防止意外或未经授权使用内核的行为。

在启动时显示内核

要在 Vijeo Citect 启动时自动显示“内核”窗口，请将 `[Debug]Kernel` 参数设置为 1。“内核”窗口在启动时打开，在关闭时关闭。缺省时不显示（0）。

注意：在试运行之后，请重置该参数，以防止意外或未经授权使用内核的行为。

定义 Runtime 命令

要显示内核窗口，定义一个用于调用 `DspKernel()` 函数的 **Runtime** 命令，给 `iMode` 参数赋值 1：

命令	<code>DspKernel(1);</code>
注释	显示（打开）内核窗口

要关闭内核窗口，再次调用 `DspKernel()` 函数，给 `iMode` 参数赋值 0：

命令	<code>DspKernel(0);</code>
----	----------------------------

注释

关闭内核窗口

注意：您应该向 DspKernel 赋予最高的权限级别，以防止操作员打开内核窗口。

关闭内核窗口

可通过从内核主窗口的“控制”菜单中选择**关闭**来关闭内核窗口。

另请参见
[内核内部](#)

内核内部

显示时，Vijeo Citect 内核由称为“主窗口”的应用程序（客户端）窗口以及一个或多个子窗口组成。启动时，内核窗口会显示有关 Vijeo Citect 启动进程的信息。内核窗口还会显示运行时系统消息，从而提供您 Vijeo Citect 系统的连续操作历史。

内核窗口包含一个命令行界面（类似于命令提示符），您可以在此键入内核命令以执行一个内核操作或显示其他子窗口。

注意：对缓存和 Cicode 内核命令的访问出于安全考虑而设置了密码保护。有关详情，请参见 [访问 Cicode 和缓存命令](#)。

缺省情况下，Vijeo Citect 会在一个处理器上运行一个实例。利用此配置，可在一个内核窗口中报告所有进程。

如果 Vijeo Citect 运行配置为运行多个实例，则每个实例都有单独的内核窗口，显示实例特定的系统消息；例如，趋势相关的系统消息会显示在内核的“趋势”窗口中。可通过使用内核窗口菜单栏上的内核菜单在各个内核窗口之间切换。

在缺省配置下，内核窗口将显示以下消息：

- **初始化子系统** - 启动 Vijeo Citect 的主要部分。
- **初始化字体系统** - 创建 Vijeo Citect 中已经定义的所有字体。这些字体用于显示诸如报警这样的信息以有 5.0 版本以前的动态文本。
- **初始化客户机系统。**
- **启动 I/O 服务器** - 只有在 Vijeo Citect 计算机是 I/O 服务器时才可见。如果计算机是一台 I/O 服务器，但并不显示此消息，则该计算机设置不正确：运行计算机设置向导检查您的配置。**已启动 I/O 服务器** - 服务器已启动并且运行正常。
 - **初始化 I/O 服务器** - 开始检查 I/O 服务器运行所需的配置，并初始化所需的所有卡。
 - 在客户机上，这些消息将被正在调用已连接的‘<I/O 服务器>’替换。
- **初始化 Cicode 系统** - 所有 Cicode 均已装载到内存中并准备运行。
- **初始化串行通讯端口系统** - 检查所有端口和硬件是否均能响应并正确运行。

- **初始化请求系统** - 这一系统处理从 Vijeo Citect 的客户端部分到 Vijeo Citect 的服务器部分的请求。
- **初始化趋势客户机系统** - 趋势客户机与正常客户机微有不同，因此它需要单独初始化。
- **启动趋势服务器** - 如果 Vijeo Citect 计算机是一台趋势服务器，您才能看见这些消息。如果计算机是一台趋势服务器，而那些消息并没有显示，最可能是因为计算机设置不正确。您应该运行计算机设置向导检查您的配置。
 - **趋势启动** - Vijeo Citect 将检查所有趋势文件，如果不能找到就创建一个新的。
 - **初始化趋势 Acq 系统** - 你定义的每个趋势都有它自己的采样率。此时，Vijeo Citect 正在设置系统以使它能够按照正确的采样率为每一个趋势笔获得数据。

在客户机上，这些消息将被正在调用已连接的‘<趋势服务器>’替换。

- **初始化报警系统。**
- **装载报警数据库** - 仅当 Vijeo Citect 计算机是一台警报服务器时，您才能看见这些消息。它把所有报警数据都装入存储区。如果计算机是一台报警服务器，而这些信息并没有显示，最可能是因为计算机设置不正确。您应该运行计算机设置向导检查您的配置。

• 打开报警存储文件

装载报警存储文件

报警存储文件被装载 - Vijeo Citect 得到报警存储文件（它在指定的目录中创建），并且为了看到已存在报警的状态而检查它。如果有冗余报警服务器，这个报警服务器将询问另一个报警服务器，而不是使用报警存储文件 - 因为另一个服务器上的文件比任何一个文件都新。

开始报警处理 - 服务器现在正在处理报警数据。

在客户机上，这些消息将被正在调用已连接的‘<报警服务器>’替换。

- **初始化报表系统。**
- **启动报表服务器** - 如果 Vijeo Citect 计算机是一台报表服务器，您才能看见这些消息。如果计算机是一台报表服务器，而此消息并没有显示，最可能是因为计算机设置不正确。您应该运行计算机设置向导检查您的配置。在客户机上，此消息将被正在调用已连接的‘<报表服务器>’替换。
- **初始化画面系统** - Vijeo Citect 现在将显示启动画面。此时，如果它被显示，Vijeo Citect 将掩盖内核。
- **初始化函数** - 执行已定义为在启动时运行任何 Cicode 函数。

信息的下一行是启动时间和 Vijeo Citect 版本号。

• 通道端口号在线

通道端口号在线

通道端口号在线 - 仅当 Vijeo Citect 计算机为 I/O 服务器时，您才能看到这些信息。这些信息告诉您在 I/O 服务器上定义的端口已经上线。如果你被告知端口没有上线或不能打开，检查你的工程配置。*端口号*是在“端口”视窗中指定的端口名称。

- **单元 'UNIT#' 端口 PORT# 上线**
单元 'UNIT#' 端口 PORT# 上线

单元 'UNIT#' 端口 PORT# 上线 - 仅当 Vijeo Citect 计算机为 I/O 服务器时显示。这指示单元号为 *UNIT#*（在“I/O 设备”视窗中定义的）的 I/O 设备被连接到端口 *PORT#*。

- **通讯系统在线** - Vijeo Citect 已完成启动操作，现在运行完全正常。

要寻找什么

Vijeo Citect 中的所有系统都应顺畅地启动。当试运行一个系统时，您应检查内核。如果任何元件在启动时没有正确初始化或报告错误，则说明您的 Vijeo Citect 系统工作不正常，您应该进行调查。

启动错误的共同原因有：

- 不正确的计算机设置（通常可由计算机设置向导解决）。
- 网络错误或硬件问题。
- 通讯错误（通常只是配置问题）。

使用主窗口检查所有的 I/O 设备在启动时都已经正确联机。首先端口必须初始化，然后 I/O 设备本身将联机。如果设备未按预期那样被初始化，Vijeo Citect 将显示：“PLC 没有响应”、“I/O 设备脱机”或相似的信息。

一些 I/O 设备可能尝试联机两次。如果是这样，Vijeo Citect 将等待（通常 30 秒）然后重试。如果 I/O 设备在两次尝试后仍然不能联机，则应检查您的配置（两端）和电缆。

注意：内核不断在主窗口中报告 I/O 设备状态的变化。这些信息也可作为报警报告给硬件报警页。

另请参见

[使用内核命令](#)

使用内核命令

命令在命令行界面发出（类似于 DOS 提示符），通常来自“主内核”窗口。某些命令在“主内核”窗口中显示结果，其它命令打开信息显示的子窗口或进一步的命令。一次最多可以打开五个窗口。

可以使用几个键盘键从命令历史中扫描和重新使用命令，以加速内核命令的发出。（命令历史是以前发出的命令的列表）。这些键盘键如下所列：

键	描述
上箭头	通过命令历史向后扫描。（命令显示在命令行中。）
下箭头	通过命令历史向前扫描。（命令显示在命令行中。）
F3	在命令行中放置发出的最后一个命令。
左箭头	在命令行中向后移动光标，一次一个字符。

键	描述
右箭头	在命令行中向前移动光标，一次一个字符。
Delete	在命令行中删除光标右边的字符。
Backspace	在命令行中删除光标左边的字符。
Insert	在命令行中从覆盖模式切换到插入字符模式。

注意：在主窗口中键入命令时，如果命令中间显示消息，则仍可以正常执行命令。使用 Shell 命令可打开新的命令窗口。

另请参见
[内核命令](#)

内核命令

下表介绍了各种内核命令。

命令	描述
Cache	改变每个 I/O 设备的缓存超时。
Cicode	打开可用于调用 Cicode 函数的子窗口。
Cls	清除主窗口或 Cicode 窗口的所有文字信息。
Debug	实现对 Vijeo Citect 与驱动程序之间原始数据传送的调试。
DriverTrace	列出 I/O 服务器上等待发送给驱动程序的驱动程序控制块(DCB)。
Exit	关闭 Cicode 或命令窗口。
Help	列出内核中的部分可用命令。
INI	显示本机 citect.ini 文件。
Kernel Alarm	版本 7.0 已经不支持
Kernel Trend	版本 7.0 已经不支持
Kernel Report	版本 7.0 已经不支持
Kernel IOserver	版本 7.0 已经不支持
Kernel Client	版本 7.0 已经不支持

命令	描述
Log	启用或禁用 I/O 设备读写的日志记录。
NetBIOS	v7.10 中已经不用。
Page General	显示常规统计信息。
Page Driver	显示有关 Vijeo Citect 系统中每个驱动程序的信息。
Page Memory	显示内存调试堆。
Page Netstat	版本 7.0 中已经不用
Page Table	显示有关 Vijeo Citect 的内部数据结构的信息。
Page RDB	显示有关 Vijeo Citect 的运行时数据库的信息。
Page Unit	显示有关 Vijeo Citect 系统中每个 I/O 设备的信息。
Pause	暂停调试输出。
Probe	版本 7.0 中已经不用
Shell	打开新的命令窗口。
Stats	重新设置所有系统统计。
SysLog	显示 syslog.dat 文件。

缓存

更改 Vijeo Citect 与之通讯的每个 I/O 设备的缓存超时设置。

语法

Cache <I/O 设备名称> <超时>

其中：

<I/O 设备名称>

（使用“I/O 设备”视窗）在工程中定义的任意有效的 I/O 设备，*表示所有 I/O 设备。

<超时>

以毫秒为单位的超时时间，0（零）表示禁用超时。

使用此命令可以调整 I/O 服务器与 I/O 设备通讯时的缓存超时。如果将超时设置为 0，则表示禁用缓存。如果为已禁用缓存的 I/O 设备指定缓存超时，则会启用缓存。

只有在 I/O 服务器正在运行同时对 I/O 设备所做的更改才会应用。如果重新启动 I/O 服务器，缓存超时将恢复到在工程中配置的值。一旦确定最优缓存超时后，就应通过在“I/O 设备”视窗（针对特定 I/O 设备）中设置该值将此更改固定下来。

另请参见

[内核命令](#)

[显示内核窗口](#)

Cicode

在本地或远程计算机上，打开一个可以调用 Cicode 函数的子窗口。在这个窗口中，可以调用任何内置或用户自定义的函数。

语法

Cicode [<名称>]

其中：

<名称>

可以是 Citect 服务器（例如，报警、报表、趋势服务器）的名称，也可以是客户计算机名称。

如果输入 Cicode 命令时不带“名称”参数，则会创建本地 Cicode 窗口。所有 Cicode 命令都在本地计算机上执行。

在多集群系统中连接到服务器时，需要指定下列语法：

<集群名>.<服务器>

例如，要连接到 Cluster1 上的报警服务器，则使用：

```
cicode Cluster1.Alarm
```

如果使用服务器或计算机名称作为“名称”参数，就可以创建到远程服务器或计算机的 Cicode 窗口。在远程 Cicode 窗口中输入的 Cicode 命令都在远程计算机上执行。例如，要创建一个所有命令都在报警服务器上执行的 Cicode 窗口，可使用：

```
Cicode Alarm
```

如果从服务器上发出命令，就可以创建到 "MyComputer" 计算机的窗口：

```
Cicode MyComputer
```

如果远程计算机可以找到，它的名字就会显示在 Cicode 窗口的标题栏上，否则就会创建本地窗口。

注意：如果在服务器上发出命令，只能指定一个计算机名称。此函数只支持客户机到服务器或服务器到客户机的连接。

每个 Cicode 命令和它自己的 Cicode 任务一起执行，因此你可以启动需要花很长时间完成的任务。Cicode 提示会在 Cicode 任务启动后立即返回，任务将在后台继续运行。如果函

数立即完成，则会显示函数的返回结果。如果函数继续运行，结果不显示并且不能返回 - 但会返回“任务仍在运行”消息和任务句柄。

注意：记住，没有对从此窗口发出的命令的权限检查，因此你对系统有完全的访问权。

Cicode 提示 0:>显示任何对象与之关联的当前窗口号。要改变当前窗口，请使用 WinGoto() 函数（或影响当前窗口的其他任何 Cicode 函数）。

Cicode 窗口不识别任何变量名称，因此当你调用 Cicode 函数时，只能传递常量（例如，数字或字符串）。当你调用需要字符串的函数时，应该传递一个字符串常量，例如，Prompt("Hello from the Kernel")。如果该字符串只是一个单字，就不需要使用分隔符，例如，Prompt(Hello)。Cicode 窗口会尝试将您输入的内容（作为参数）转换为正确的数据类型。如果不能转换这些参数，它就传递 0 或空字符串给函数。

注意：有些 Cicode 函数被编译器作为标签宏来实现。当函数的参数数目改变时，这些宏允许向后兼容。因为 Cicode 窗口不扩展宏，所以你不能直接调用这些函数。你必须使用宏扩展。如果无法找到你试图使用的函数，在函数名前面增加一个下划线（_）后再试一次，例如，_DevClose(1)。还可以通过使用 Shutdown() 函数从此窗口关闭 Citect 系统。

另请参见
[内核命令](#)
[显示内核窗口](#)

Cls

从主窗口或 Cicode 窗口中清除所有文本，并将光标移到左上角。

语法

Cls

当窗口（因为显示调试数据或太多命令）而变得很混乱时，使用此命令可清除当前窗口。

另请参见
[内核命令](#)
[显示内核窗口](#)

调试

针对在 Citect 和选定驱动程序之间传送的所有原始数据启用调试。所有的协议流量都显示在内核窗口中，并记录在 SysLog.DAT 文件中。

语法

Debug<Port> <Mode> [*<Display>*]

其中：

<Port>

在工程中配置的端口名称（使用端口表）。

<Mode>

模式：

ALL - 跟踪所有低层通讯流量直至内核窗口。

READ - 跟踪所有读命令的低层通讯流量直至内核窗口。

WRITE - 跟踪所有写命令的低层通讯流量直至内核窗口。

ERROR - 跟踪所有包含协议错误的低层通讯流量。此模式仅在检测到错误时才产生跟踪，因此您可以保持此选项长时间开启（以查找困难的问题）。

OFF - 停止任何类型命令的调试跟踪。

注意：“TO”和“FROM”模式现已废弃。

<Display>

显示部分（可选）：

MONO - 向单色监视器（如果已连接上）和内核发送调试。

MONOONLY - 只向单色监视器发送调试。

要将某端口置于调试模式下，输入 `DEBUG`，紧随其后输入端口名称和所需模式。如果不知道端口名称，输入 `DEBUG`（不带任何参数），Citect 会显示一个由所有可用端口构成的列表。

只有 I/O 服务器能与 I/O 设备通讯，所以此命令通常仅适用于 I/O 服务器。

当启用调试跟踪模式时，Citect 会在内核窗口中显示所有的协议流量并将其记录在 `SysLog.DAT` 文件中。这会降低 Citect 的性能（因为可能会有很多数据），因此，不要在对当前操作非常重要的 I/O 服务器上启用调试跟踪。仅在试运行期间或不重要的段使用此命令。过度使用此命令可能导致 I/O 设备离线。



警告

不希望的设备操作

请勿在工作环境中启用或使用调试跟踪模式。此模式仅用于调试过程。

忽视这些说明可能导致死亡、严重人身伤害或设备损坏。

当向单色监视器发送调试追踪时，它直接通过驱动程序的中断程序显示，因此速度很快。当处于活动状态时，此跟踪（如果使用 MONOONLY）可以降低 Citect 的 CPU 负荷，并

提供即时输出。如果您正在自行开发驱动程序，并且您的驱动程序在调试追踪显示在“内核”中之前崩溃，此方法很有用。

可以在跟踪处于活动状态时使用 [Shell](#) 命令创建额外的命令窗口。这样，您将可以输入更多命令（在新的窗口中）。

可以使用 [暂停](#) 命令停止调试输出（以查看数据）。

另请参见
[内核命令](#)
[显示内核窗口](#)

DriverTrace

列出 Vijeo Citect 向驱动程序发出的命令及信息：在调试驱动程序时，这些命令和信息非常有用。例如，它可以帮助确定某驱动程序无法完全初始化的原因。

注意：运行 `syslog` 追踪将占用大量 CPU 使用率。如果执行大量追踪操作，您需要监视对 CPU 性能的影响。

语法

DriverTrace [OFF|CMDS|VER[BOSE]]|PORT=<name>|MASK=<xxxxxxxx>|DUMP]

其中：

空

返回当前的 DriverTrace 设置。这对于在实现新的追踪前，确定当前的掩码或端口设置非常有用。仅在内核中键入 `DriverTrace`，然后点击回车键，查看当前设置。

OFF

关闭 DriverTrace。

CMDS

启动 DriverTrace，但不会显示数据缓冲区的内容。

注意：特定驱动程序命令必须使用内核命令 `DriverTrace[Mask]` 或 `DriverTraceMask` INI 参数才能启用。

VER[BOSE]

启动 DriverTrace，但可以显示数据缓冲区的内容。

注意：特定驱动程序命令必须使用内核命令 `DriverTrace[Mask]` 或 `DriverTraceMask` INI 参数才能启用。

PORT

允许将追踪限制为某特定驱动程序端口，方法为：定义端口名。缺省设置 `PORT=*` 表示将对所有端口进行追踪。

MASK

一个 4 字节的十六进制数，表示一个位掩码，可用于包含驱动程序命令，或从 DriverTrace 中排除驱动程序命令。这些驱动程序命令及其值如下显示：

命令	比特位置
CTDRV_INIT	00000001
CTDRV_OPEN	00000002
CTDRV_INIT_CHANNEL	00000004
CTDRV_INIT_UNIT	00000008
CTDRV_READ	00000010
CTDRV_WRITE	00000020
CTDRV_CONVERT	00000040
CTDRV_CANCEL	00000080
CTDRV_CPU	00000100
CTDRV_DATABASE	00000200
CTDRV_STOP_UNIT	00000400
CTDRV_STOP_CHANNEL	00000800
CTDRV_CLOSE	00001000
CTDRV_FORMAT	00002000
CTDRV_STATS	00004000
CTDRV_DEBUG	00008000
CTDRV_INFO	00010000
CTDRV_STATUS_UNIT	00020000
CTDRV_INIT_CARD	00040000
CTDRV_UPDATE_INFO	00080000
CTDRV_UI_READ	00100000
CTDRV_UI_WRITE	00200000
CTDRV_EXIT	00400000
CTDRV_LINE_STATUS	00800000

命令	比特位置
CTDRV_SEND_BREAK	01000000
CTDRV_REINIT_CHANNEL	02000000
CTDRV_SET_PARAM	04000000
CTDRV_GET_PARAM	08000000
CTDRV_OPEN_PORT	10000000
CTDRV_CLOSE_PORT	20000000
CTDRV_STATUS_DISCONNECT	40000000

例如，仅能够用于包含 CTDRV_OPEN、CTDRV_INIT_UNIT 和 CTDRV_READ 命令的值为：0000001A。

由于经常调用 CPU 函数，因此大多数用户都不希望对其进行调用。要取消对 CPU 函数的调用，可以将掩码设置为 7ffffeff。

缺省的 <掩码> 为 7FFFFFFF。

DUMP

列出 I/O 服务器上等待发送给驱动程序并可从驱动程序进行操作的驱动程序控制块 (DCBs)。

另请参见

[内核命令](#)

[显示内核窗口](#)

退出

关闭 Cicode 或命令行窗口。

语法

Exit

注意： 您不能使用这个命令关闭主窗口。请参见 [关闭内核窗口](#)。

要关闭所有其他窗口，请从窗口的控制菜单框中选择“关闭”，或按 Esc 键，或双击控制菜单框。

另请参见

[内核命令](#)

[显示内核窗口](#)

帮助

显示由内核中提供的某些命令组成的列表。

语法

帮助

另请参见：

[内核命令](#)

[显示内核窗口](#)

INI

显示本机 `citect.ini` 文件。（如果你正通过因特网使用 Citect 因特网显示客户机运行工程，将显示 `bin` 目录中的 `.ini` 文件）。你也可以使用上下翻页键和上下箭头在文件中移动，但不能编辑和存盘。

语法

INI

注意：这个窗口和显示 `syslog.dat` 文件的一样。你可以在这个窗口中显示 `citect.ini` 或 `syslog.dat` 文件，但你不能同时显示两个文件。

另请参见

[内核命令](#)

[显示内核窗口](#)

Log

启用或禁用 I/O 设备读取和写入到 `syslog.dat` 文件的日志记录。

语法

Log<I/O Device> <Mode>

其中：

<I/O Device>

要记录日志的 I/O 设备的名称。

<Mode>

下列值之一：READ、WRITE 或 OFF

另请参见

[内核命令](#)

[显示内核窗口](#)

综合页

显示 Citect 总体性能的综合统计信息。

语法

综合页

综合统计

服务器名	这台计算机的服务器名或，如果它是客户机，这台客户机与之通讯的主服务器名。
节点名	这一计算机的计算机名。你可以通过计算机设置精灵设置它。仅在网络配置中包含 Citect 时使用。
时间	当前的时间和日期。
CPU 索引	计算机性能指示。这个数字提供了计算机性能的大体指示。对 Compaq 486/25M，它是 25。
运行	Citect 运行的总时间。
统计数据重置	重置统计数据的 Cicode 命令的运行时间。
总内存	自由内存的数量（包括虚拟内存）。
物理内存	自由虚拟内存的数量。在计算机上自由的物理 RAM（不包括虚拟内存）。
内存资源	自由的 Windows 系统资源的百分比。
调度程序周期	<p>Citect 调度程序正在寻找任务执行的时间。一个计算机可以多快响应事件的好的指示。这一值依赖于计算机的速率和 Citect 使用的 CPU 资源。</p> <p>这个值是完全不同的，依赖于你是用 32 位或 16 位 Citect。这是因为 Citect 可以使用 32 位操作系统完成调度程序，而程序本身是运行在 16 位环境中的。</p> <p>16-位：Citect 越忙，此数值越低。典型值从 10, 000 到 50, 000；高负载系统可能降到 1000 以下。</p> <p>32-位：Citect 越忙，此数值越高。典型值从 100 到 1000；高负载系统可能升到 5000 以上。</p>
CPU 利用率	<p>正在使用的计算机处理能力的利用率。</p> <p>当百分比增加到较高的值时，Citect 的性能可能会下降或非常缓慢。如果 CPU 利用率一直非常高（大于 70%），您的系统可能有问题或 CPU 过载了。为了最好的运行，您的系统应该运行在 0% 到 40% 之间。</p>
每秒任务	Citect 调度程序执行时每秒的任务数，显示 Citect 有多忙。每秒应该有 30 到 200 个任务。如果计算机是输入输出服务器，任务数高一些（因为每一个协议使用很多任务）。
错过的错误	Citect 在本机错误缓冲区中跟踪内部错误。当检测到错误并且没有缓存来放置错误信息时，“错过的错误”计数器增加。正常时应该为 0（零）。如果这个值增加，您的系统可能有问题。

I/O 服务器统计

下面的统计数据仅在计算机被配置成 I/O 服务器时增加。

读请求	<p>第一个数是从所有的客户机 Citect 计算机（包括本地 Citect 客户机）向 I/O 服务器发送的读请求总数。这个值从 0（到十亿）不断增加，依赖于客户机请求的速度和服务器运行多长时间。</p> <p>秒数更有用 - 它记录每秒的读请求。这个值也依赖于客户机从 I/O 服务器请求数据的速度，应该在每秒 5 到 400 个请求之间。</p>
物理读	<p>第一个数是对 I/O 设备所作的物理读的总数。因为 I/O 服务器优化对 I/O 设备所做得读请求，这个数通常小于读请求数。这个数小于读请求数并且永远增加。这个数小于读请求数并且永远增加。</p> <p>第二个数是每秒物理读数（I/O 服务器正在执行）。这一值依赖于客户机和服务器与 I/O 设备通讯能有多快（典型值是 5 到 200 之间）。</p>
块读	对输入输出设备同一地址所作的相同的读的次数。服务器将两个请求合成一个块作为优化。
数字读	I/O 服务器从 I/O 设备读的数字点的数目。
每秒数字读	I/O 服务器每秒处理数字读的个数。它提供了一个性能的总的指示。它依赖于协议。
写请求	<p>第一个数是从所有的客户机 Citect 计算机（包括本地 Citect 客户机）向 I/O 服务器发送的写请求总数。第二个数是每秒的写请求数（I/O 服务器正在运行）。</p> <p>请求数依赖于客户机发送写请求到 I/O 服务器的频率（典型值在 0 到 20 之间）。如果请求数总是超过 10，这可能引起性能下降（通常是由 Cicode 向 I/O 设备写入太多内容引起的）。</p>
物理写	<p>第一个数是向 I/O 设备所作的物理写的总数。因为 I/O 服务器优化对 I/O 设备所做得写请求，这个数通常小于写请求数。第二个数是每秒物理写数（I/O 服务器正在执行）。</p> <p>请求数依赖于客户机发送写请求到 I/O 服务器的频率（典型值在 0 到 20 之间）。如果请求数总是超过 10，这可能引起性能下降（通常是由 Cicode 向 I/O 设备写入太多内容引起的）。</p>
块写	对输入输出设备同一地址所作的相同的写的次数。服务器将两个请求合成一个块作为优化。
寄存器读	I/O 服务器从输入输出设备读的寄存器点的数目。
每秒寄存器读	I/O 服务器正在运行时每秒读寄存器的数目。它提供了一个性能的总的指示。它依赖于协议。
缓存读	服务器从读缓存中读请求的数目。
读缓存 (%)	<p>从缓存中读的比率。如果缓存读的百分比很大（例如，大于 40%），输入输出设备的缓存超时可能设得太高了。试着减少 I/O 设备缓存超时值使得百分比降到 40% 以下。</p> <p>缓存读百分比依赖于 Citect 系统的配置和连接的客户机的数目。如果你有许多客户机需求相同的 I/O 设备数据，这个百分比就很高，这通常并不会影响性能。例如，如果你有 10 个 Citect 客户机浏览相同的画面，90% 的缓存读是可以接受的。</p>
缓存超界	因为写请求指向了相同的位置引起缓存区域超界的次数。当您向 I/O 设备写入内容并且这个地址位于读缓存中，Citect 将冲掉来自缓存的数据。下一次数据被读时，它从 I/O 设备重新加载来反映新值。
缓存提前读	提前读更新缓存区的次数。当提前读缓存被使能时，I/O 服务器试着去读任何一个 I/O 设备接近缓存超时值的数据。I/O 服务器也仅在对 I/O 设备的通讯通道空闲时读数据 - 给其它的读请求更高的优先权。
缓存区	已被分配的激活缓存区的数目。从 I/O 设备读的每一块数据块要存放在缓存中时都会请求缓存区。一次激活的缓存区的数目依赖于 Citect 计算机的动态操作和它们的工程配置（典型值从 0 到 100）。

缓存短缺	缓存需要分配附加的区域但没有区域可以利用的次数。当这个发生时，服务器不能存取数据。它不产生错误但降低性能。如果这个值增加的太快，就增加可利用的存储区。
响应时间	<p>I/O 服务器处理读写响应的时间（例如，从请求到达服务器到客户机返回的时间）。这个时间依赖于 I/O 设备的物理响应时间和请求在队列中等待其它请求完成的时间。在服务器的负载增加时，这个时间增加，是系统响应良好的指示。</p> <p>显示平均，最小和最大时间。经典值依赖于输入输出设备协议，然而，Citect 也有 500 到 2000 毫秒的平均响应时间。较慢的协议或重负载系统使响应时间加长，但即使对于非常大的系统，你也可以在少于两秒内得到响应。</p>
点	在 Citect 系统中的最大 I/O 点数。静态和动态计数的总和不能超过这个数。点的限制是 Citect 软件保护的一部分，并且被编程在您的硬件密钥上。
最大全权限	可以在您的 Citect 系统中存在的最大全功能 Citect 计算机（全权限）数。这个数是 Citect 软件保护的一部分，并且被编程在你的硬件密钥上。
当前全权限	当前在您的 Citect 系统中运行的全功能 Citect 计算机（全权限）数。
记录最高全权限	因为 Citect 重启，你的 Citect 系统经历的最高权限。
静态	静态 I/O 点数。
最大管理客户机	在您的 Citect 系统中，仅查看客户机的最大数目。这个数是 Citect 软件保护的一部分，并且被编程在你的硬件密钥上。
当前管理客户机	运行在您的 Citect 系统中的仅查看客户机的当前数目。
记录最高管理客户机	因为 Citect 重启，您的 Citect 系统经历的仅查看客户机的峰值数目。
动态	动态 I/O 点数。
最大显示客户机	在您的 Citect 系统中，仅用于显示的客户机的数目。这个数是 Citect 软件保护的一部分，并且被编程在你的硬件密钥上。
当前显示客户机	运行在您的 Citect 系统中的仅用于显示的客户机的当前数目。
记录最高显示客户机	因为 Citect 重启，您的 Citect 系统经历的最高仅用于显示的客户机的数目。

另请参见
[内核命令](#)
[显示内核窗口](#)

画面驱动程序

显示有关 Citect 系统中每一个驱动程序的信息。仅当 Citect 计算机配置成附带 I/O 设备的 I/O 服务器时，此窗口才显示。

语法

画面驱动程序

使用上下翻页键扫描驱动程序列表。

驱动程序统计

端口名称	驱动程序用于通讯的物理端口名。
协议	协议名。
标题	协议标题和版本信息。Citect 版本 3.xx、4.xx 和 5.xx 的所有协议驱动程序是版本 2.0 类型驱动程序。
读请求	第一个数是从所有的客户机 Citect 计算机（包括本地 Citect 客户机）向驱动程序发送的读请求总数。 第二个数是每秒读请求数（I/O 服务器正在运行）。
物理读	第一个数是对输入输出设备所作的物理读的总数。因为 I/O 服务器优化对 I/O 设备所做得读请求，这个数通常小于读请求数。这个数小于读请求数并且永远增加。 第二个数是每秒物理读数（输入输出服务器正在执行）。
块读	对 I/O 设备同一地址所作的相同的读的次数。服务器将两个请求合成一个块作为优化。
数字读	I/O 出服务器从 I/O 设备读的数字点的数目。
每秒数字读	I/O 服务器每秒处理数字读的个数。它提供了一个性能的总的指示。它依赖于协议。
写请求	第一个数是从所有的客户机 Citect 计算机（包括本地 Citect 客户机）向驱动程序发送的读写求总数。 第二个数是每秒的写请求数（I/O 服务器正在运行）。
物理写	第一个数是向 I/O 设备所作的物理写的总数。因为 I/O 服务器优化对 I/O 设备所做得写请求，这个数通常小于写请求数。 第二个数是每秒物理写数（I/O 服务器正在执行）。
块写	对 I/O 设备同一地址所作的相同的写的次数。服务器将两个请求合成一个块作为优化。
寄存器读	I/O 服务器从输入输出设备读的寄存器点的数目。
每秒寄存器读	I/O 服务器正在运行时每秒读寄存器的数目。它提供了一个性能的总的指示。它依赖于协议。
缓存读	服务器从读缓存中读请求的数目。
读缓存	从缓存中读的比率。如果缓存读的百分比很大（例如，大于 40%），输入输出设备的缓存超时可能设得太高

(%)	<p>了。试着减少 I/O 设备缓存超时值使得百分比降到 40% 以下。</p> <p>缓存读百分比依赖于 Citect 系统的配置和连接的客户机的数目。如果你有许多客户机需求相同的 I/O 设备数据，这个百分比就很高，这并不会引起问题。例如，如果你有 10 个 Citect 客户机浏览相同的画面，90% 的缓存读是可以接受的。</p>
错误数	驱动程序发生的总的错误数。
存储区短缺	服务器需要为驱动程序申请存储区但没有区域可以利用的次数。它降低通讯性能并且导致丢失请求。如果这个值增加的太快（即，一分钟超过 10 次），就增加可利用的存储区。
驱动程序错误	在重试发生前，底层驱动程序发生错误的数目。因为驱动程序重试和它完成了第二次重试，因此，即使没有错误报告，这个值也可能不断增加。如果这个值增加的太快（每分钟 10 次或更多）而没有其它错误被报告，就可能有影响性能的底层错误。
存储区超界	驱动程序需求存储区但没有可利用的次数。驱动程序必须放弃数据。如果这个值增加的太快，就增加驱动程序能够处理的未决控制的数目。这个值只和客户机驱动程序增加。I/O 设备不需要未决存储区。
超时	驱动程序在操作中发生的超时数。这个值在当驱动程序重试而没有可见错误的情况下不断增加。如果这个值增加的特别快，就可能有通讯问题。
重试次数	驱动程序执行的重试数。如果这个值增加，就可能有通讯问题。
最大未决控制	被保持在存储区中等待协议处理的请求数。每个协议都有提前定义的缺省值。如果一个协议一次可以处理多请求或多命令，这个值可能很高。如果协议一次只能处理一个命令，这个值可能是 1 或 2。
打包尺寸（字节）	<p>这是 I/O 驱动程序在运行时将读或写请求打包的可以使用范围。缺省值是在多次的试验后设置的。为了从 I/O 设备更快的得到响应，这个值被计算作为 I/O 设备可以响应的数据的最佳范围。</p> <p>例如，如果打包尺寸是 100，在一个图形页面上有地址 1 和地址 99，Citect 将在一个请求中读两个地址。如果在画面上有地址 1 和地址 101，Citect 将发出两个独立的请求。打包尺寸不能大于协议的最大包尺寸，因为 I/O 设备的特定类型响应小需求比响应大需求更快。</p> <p>使用这种方法时需要特别注意，许多 I/O 设备必须有用户创建的存储表。如果用户没有在一个范围内定义所有的存储地址，Citect 就可能从 I/O 设备中读一块不存在的存储区（有一个硬件报警）。这是因为 Citect 将请求一个从起始地址到结尾地址的整个范围。所以，在先前的例子中，I/O 设备中没有地址 76，它将返回给 Citect 它不能读地址 76。I/O 设备不知道它不需要这个地址，仍将重试这个命令，在许多情况下将最终使 I/O 设备离线。</p> <p>确认已经定义了 Citect 需要的存储地址。</p>
超时期（毫秒）	如果没有从 I/O 设备来的回答，这是再请求数据之前 I/O 驱动程序需要等待的时间。
最多重试次数	<p>I/O 驱动程序为了从 I/O 设备得到数据而重试的次数。</p> <p>在 Citect 设置 I/O 设备离线之前，重试和超时给你的所有时间。</p> <p>如果超时是 2000，重试是 2，Citect 为了得到响应将等待 2 秒，然后重试，等待 2 秒，重试，等待 2 秒，离线。失去通讯到离线一共需要 6 秒。您可以修改这个参数，但如果您设置的太慢，就会产生不必要的重试并且可能得到 I/O 设备离线的消息。</p>
循环时间（毫秒）	这是 Citect 检查端口寻找数据或向端口写数据的时间，单位毫秒。如果是 0，协议以中断方式运行。
传输延时（毫秒）	这是 Citect 在收到最后一个请求发出的响应和发出新的请求之间保持数据的时间。通常为 0，然而，一些协议成为饱和并且开始混乱。在这种情况下，缺省值在协议测试时就计算好了，减小这个值可能引起问题。然

	而，增大它也会对你的系统的整体响应时间产生一点影响，但可使通讯更加稳定。
观察时间 (秒)	这是 Citect 在 I/O 设备离线之后，重新建立连接之前等待的时间。通常是 30 秒。它可以设的小一些，但不能小于超时和重试的值，否则，您将不能和 I/O 设备重新建立连接。
响应时间	驱动程序处理读和写请求的时间（即处理对 I/O 设备单个读或写操作所花的时间）。这个时间依赖于 I/O 设备的物理响应时间，因为不包括队列等待的时间。这个值反映通讯通道的调整（例如，增加波特率将减少响应时间）。 显示平均，最小和最大时间。
通道利用率	这个值显示正使用的硬件通道整体能力的百分比。I/O 服务器会试图保持尽可能高的利用率，然而，如果客户机 Citect 计算机请求数据比通道提供的慢，总数将低于 100%。通道利用率高于 100% 也是可能的，因为许多 I/O 设备的驱动程序可以同时处理多个命令（同时有多个命令使用通道）。
每秒字节	驱动程序每秒传输的字节数。当调整驱动程序时，这个值提供了一个简单的性能指示。
专门变量	通过使用详细模式（按 V）或按下箭头，显示 20 个专门变量。那些变量是驱动程序专用的。

另请参见

[内核命令](#)

[显示内核窗口](#)

画面内存

显示内存调试信息。此窗口专为 Citect 专家设计，需要具备高级专门技术才能使用。

语法

画面内存

另请参见

[内核命令](#)

[显示内核窗口](#)

队列页

队列是 Vijeo Citect 中使用的主要数据结构，队列页允许系统中各种队列活动进行显示。要查看不同的队列，请按上下翻页键。

在每一页上，将显示正在被查看队列的句柄、名称和长度。页中显示的队列数量和条目都完全取决于在 Citect32 和单个工程中的自定义配置。队列格式可以变化，但有若干通用格式。

范例

队列休眠

句柄 5 长度 49

名称	句柄	状态	优先级	Cpu	最少	最多	平均	计数
U Anm.Animate	9	休眠	用户	0.4	0.000	0.003	0.000	1134
U Tran.Task.Delay	15	休眠	用户	0.0	0.000	0.000	0.000	2231
U DISKDRV\WatchDog	22	休眠	用户	0.0	0.000	0.000	0.000	7
U Alarm.Heart	57	休眠	用户	0.0	0.000	0.000	0.000	1
U Alarm.ServerMoni	69	休眠	用户	0.0	0.000	0.000	0.000	3
U Report.Heart	72	休眠	用户	0.0	0.000	0.000	0.000	1
U Alarm.HardRelease	70	休眠	用户	0.0	0.000	0.000	0.000	3
U Trend.Client	55	休眠	用户	0.0	0.000	0.000	0.000	95
U Spl.Task	45	休眠	用户	0.0	0.000	0.000	0.000	96

这是最常见的队列类型。列的意义如下：

模式（名称被隐藏）：U 表示用户用户任务，S 表示系统任务。

名称：任务的名称。

句柄：任务的内部句柄。

状态：任务的状态，可以是下列状态之一：空闲、发生、就绪、休眠、等待、暂停和调试。

优先级：队列中项目的优先级。它可以下列值之一：低、用户和高。

CPU：显示该任务所用 CPU 时间的百分比。

最少：对此任务的最少响应时间（来自统计）。

最多：对此任务的最多响应时间（来自统计）。

平均：平均响应时间。

计数：此任务已运行的次数。

另请参见

[内核命令](#)

[显示内核窗口](#)

页面 RDB

显示每一个 Citect 运行数据库的信息 (RDB)。

语法

页面 RDB

使用上下翻页键在表格间移动。

运行数据库包含已编译工程配置信息。有两种类型的 RDB：常驻和非常驻。

常驻数据库在 Citect 启动时装入并留在内存中。常驻数据库的例子是报警、趋势、报表和函数。常驻数据库的名字以下划线 (_) 开始。

非常驻数据库和页面联系。数据库在需要时—当页面显示时被装入或卸载。

每个数据库被划分成大约 10 个表格； <name>, TEXT, REQUEST, PIECE, CODE, SCALE, RUN, WRITE, FUNC, SYMB 和 <name>。

另请参见

[内核命令](#)

[显示内核窗口](#)

表格页

包含 Citect 运行时数据结构信息。这个域非常广泛并且在开始时有点难定位。当前大约有 50 个表格信息 - 绝大多数仅与特定环境相关，否则没有用。

语法

表格页 [<名称>]

其中：

<名称>

任选的显示表格的名字。

使用上下翻页键浏览整个表格。使用上下键滚动表格数据。

MASTER_TABLE (表格 1) 列出所有表格。下列表格非常有用：

表格页统计

这个有用的表格包含了在 Citect 上运行的每一个任务执行的时间和循环时间。执行时间是整个任务运行的时间。循环时间是任务启动和再启动之间的时间。CPU 是任务运行所占整个 CPU 的使用率（快速任务经常是 00 CPU）。

Citect 0 条目是主窗口的显示任务（图形页更新）。即，Citect 从 I/O 服务器请求数据，I/O 服务器得到数据并送回给客户机，客户机更新显示所需要的总时间。

注意： Citect 0 协调主窗口的显示任务。Citect 1 为第一个子窗口，Citect 2 为第三个，等等。

进入 CodeX 协调 Cicode 任务，X 是任务句柄。你可以通过浏览 Cicode 表发现哪一个任务对应于哪一个句柄。

注意： 进入 Trend Acq entry 获得在工程中定义的每一个不同趋势的采样时间。

表格页 Cicode

这个表格包含了当前运行的所有 Cicode 任务。它包含任务名、句柄、运行状态和一些统计信息。CPU_Time 是任务已经运行的所有时间—每次任务运行它都增加。CPU 是任务运行所占整个 CPU 的使用率（快速任务经常是 00 CPU）。

表格页传输

此表显示了 Vijeo Citect 组件之间全部通信通道的列表。确切的两个独立组件之间存在 tran 协议。客户端 tran 启动一个连接，服务器 tran 等待连接。客户端和服务在此上下文环境中与拥有该 tran 协议的组件类型没有任何关系。

有两种模式可以查看 tran 表：“标准”和“详细”。“标准”模式以表格形式显示该传输的全部相关信息。它可以使用上/下方向键进行滚动，顶行指示当前所选的 tran。所选 tran 的扩展信息可以通过按“v”键切换至“详细”模式进行查看。进入“详细”模式后，可以使用 Page Up/Page Down 键来改变要显示的 tran。

在标准模式中，tran 表包含下列各列：

域	描述
名称	<p>截短至 20 个字符的通信通道的名称和类型：</p> <p>客户端 格式：<集群名称><服务名称> 示例：Cluster1Alarm, Cluster1Report, Cluster1Trend</p> <p>服务器 格式：<服务器名称><集群名称> 示例：AlarmServer1Cluster1, ReportServer1Cluster1, TrendServer1Cluster1</p> <p>专用（客户端和服务） 格式：@@<集群名称><服务器名称> 示例：@@Cluster1.AlarmServer1, @@Cluster1.ReportServer1, @@Cluster1.TrendServer1</p>

域	描述
节点	<p>平台（客户端和服务端） 格式：<服务器名称><集群名称> 示例：AlarmServer1Cluster1, IOserverCluster1</p> <p>要么是 tran 所连接的节点，要么是连接的状态：</p> <p><呼叫> 客户端 tran 试图进行连接。</p> <p><侦听> 服务器 tran 等待被连接。</p> <p><禁用> tran 当前被禁用。</p> <p><i>节点名称</i> tran 已经连接且已在线。其值为 tran 要连接的节点/计算机的名称。这可能是当前的计算机或其他计算机，取决于工程中的 TCP/IP 配置。</p>
类型	<p>这是其中之一：</p> <p>客户端 - 表示客户端 tran。此 tran 主动地尝试连接至服务器 tran。</p> <p>服务器 - 表示服务器 tran。此 tran 被动地等待来自客户端 tran 的连接尝试。</p> <p>SerRnd - 表示服务器到服务器的冗余 tran。这是当工程被配置为服务器冗余时主服务器与备用服务器之间的连接。</p>
模式	<p>这是其中之一：</p> <p>本地 此 tran 连接处于同一进程的两个组件。例如，当 Vijeo Citect 运行在单一进程模式时，所有服务器组件都与客户端运行在同一进程模式。此时，这些组件之间的连接应标记为“本地”。</p> <p>OutPro（超出进程范围） 此 tran 要么已通过 TCP/IP 在两个组件之间建立了一个连接，要么正试图建立连接。各组件处于不同的进程。它们可能处于同一台计算机，也可能处于不同的计算机。</p> <p>OutPrD（超出进程范围专用） 当 Vijeo Citect 运行于多进程模式时，单一机器上的客户端与各个服务器进程之间存在的一种专用连接。这些 tran 使用管道连接来确认某一台计算机上运行的所有 Vijeo Citect 组件之间存在一个通信路径，忽略工程的 TCP/IP 配置。专用连接在不同的计算机之间不存在。</p>

域	描述
	<p>Platfo（平台 Tran）</p> <p>此 tran 使用平台网络模块作为组件之间的传输层。对于 v7.0，这仅用于连接到 I/O 服务器。</p> <p>注意：有一些客户端报警服务器平台 tran 会在 Tran 表中显示。v7.0 中不实施或要求服务器端报警服务器平台，因此，这些客户端 tran 将始终保持在连接状态。它们可以被安全地忽略。</p> <p>远程</p> <p>此服务器 tran 通过 TCP/IP 等待连接尝试。连接上之后，此模式变更为 OutPro。</p>
hMsg	此值表示 Vijeo Citect 的信息传输层所用的会话句柄。专用或平台 tran 不使用它是因为它始终保持为 -1。
Cnt	<p>此值指示 tran 已建立连接的次数。它特别地对 tran 接收到 MSG_OPEN 信息的次数进行计数。此值对本地 tran 没有任何意义，因为它保持为 0。</p> <p>如果数字高了，这表明网络在失去连接后重新建立连接。然而，也能意味着服务器已经长时间运行，而且许多客户端已经启动并停止，因此关闭和打开会话。</p>
发送	此值显示已由 tran 发送的消息数量。
Rec	此值显示已由 tran 接收的消息数量。
等待	此值表示 tran 为获得缓冲以便发送消息而不得不等待的次数。
堆栈	<p>此值指示协议数量。从历史上看，此值综合了 NetBIOS LanA 的数量和 TCP/IP 协议堆栈。但是，v7.0 已经删除了对 NetBIOS 的支持，而对 TCP/IP 的支持有所增强，包括对冗余网络界面卡 (NIC) 的支持。</p> <p>堆栈值显示的是一个指数（以 1 为底），该指数指示使用冗余 IP 地址的服务器 tran 是正在侦听还是已经连接。其值仅用于超出进程范围的 TCP/IP 服务器 tran；它对所有其他 tran 都没有任何意义。</p>
服务	此列显示了 tran 所用的服务类型（忽略 tran 模式）。有效服务有报警、IO、报表和趋势。
状态	这是 tran 的当前状态。有效状态有“在线”、“离线”、“连接中”（仅限客户端 tran）、“断开连接中”、“侦听中”（仅限服务器 tran）和“禁用”。

表格页 CSAtoPSI.Subs

显示客户端标签预定的列表。

另请参见

[内核命令](#)

[显示内核窗口](#)

单元页

显示有关 Citect 系统中每个输入输出设备的信息。此信息在 Citect 计算机被配置作为输入输出服务器或仅配置为客户机时显示。如果计算机是一台客户机，则显示所有输入输出服务器的输入输出设备。如果计算机是一台输入输出服务器，则只显示这台输入输出服务器的输入输出设备。可通过使用 Verbose 模式（按 V 可启用 Verbose 模式）显示其他输入输出服务器的输入输出设备。如果计算机是客户机，则并不是所有输入输出设备信息都会被更新（因为只有输入输出服务器具有此信息）。

语法

单元页

如果 Citect 计算机是客户机，状态代码和错误代码仅仅是本地计算机的，并不反映输入输出服务器上输入输出设备的真实状态。请注意，所有已配置的输入输出设备都显示在此窗口中，并不是只显示特定输入输出服务器的输入输出设备。（任何远程输入输出设备并不反映该输入输出设备的真实状态）。

使用上下翻页键可浏览输入输出设备列表。

输入输出设备信息

I/O 设备	工程中（使用输入输出设备表单）定义的输入输出设备的名称。
I/O 服务器	服务于此输入输出设备的输入输出服务器的名称。
注释	工程中（使用输入输出设备表单）定义的输入输出设备的描述。
输入输出设备号	工程中（使用输入输出设备表单）定义的输入输出设备号。
PLC 号	工程中（使用输入输出设备表单）定义的物理输入输出设备地址。
端口名称	输入输出设备连接到的通信端口。
协议	与输入输出设备通信所使用的协议。
服务器状态和客户机状态	<p>输入输出设备的状态。仅当计算机是输入输出服务器并且服务于此输入输出设备时，该服务器状态才是有效的。客户机状态字段只对客户机有效，并指示挂接到输入输出服务器的输入输出设备的状态。输入输出设备状态可以是下列一种：</p> <p>运行 - 表示与输入输出设备的通信连接正常。</p> <p>备用 - 表示与输入输出设备的通信连接正常，但与该输入输出设备的通信目前是通过另一个端口进行的。此端口处于备用模式。</p> <p>开始 - 表示服务器目前正在建立（与输入输出设备的）通信连接。</p> <p>停止 - 表示服务器目前正在放弃（与输入输出设备的）通信连接控制。</p> <p>离线 - 表示服务器不能与输入输出设备建立通信通道。如果有备用端口或服务器可以使用，Citect 会试着使用该端口与输入输出设备通信。</p> <p>远程 - 表示输入输出设备状态正常，但目前没有连接。</p>

主	指示输入输出设备是否处于主模式下; Yes = 主, No = 辅 。如果输入输出设备处于主模式下, 只要服务器被激活, 它就会立即启动与输入输出设备的通信通道。如果输入输出设备处于辅模式下, 当服务器启动时, 输入输出设备会保持不被激活 (直到主输入输出设备失败)。
客户机正在使用	此客户机正在使用的输入输出服务器的名称。这使您可以识别主辅输入输出服务器。
通用错误	驱动程序返回的最后一个通用错误代码。由于所有协议驱动程序都有自己的特殊错误, 因此输入输出服务器无法识别它们。驱动程序会将这些特殊错误转换成服务器可以识别的通用错误。
错误句柄	错误句柄是输入输出服务器为每个错误指定的。 Citect (此时) 不使用这个句柄。
驱动程序错误	驱动程序特定的错误代码。每个驱动程序都有自己的特殊错误代码。有关每个错误代码的说明, 请参考驱动程序特定的错误 (对特定协议)。
错误消息	与普通错误代码关联的提醒信息。
错误数	来自输入输出设备的错误数。
重启次数	服务器试图与输入输出设备建立连接的次数。通常是 1 , 因为服务器在启动时建立一个连接。如果这个值显示大于 1 , 通信通道就有问题。
响应时间	<p>驱动程序处理读写请求的时间 (例如, 处理对输入输出设备单个读或写操作所花的时间)。此时间取决于输入输出设备的物理响应时间, 因为不包括队列等待时间。这个值反映通讯通道的调整 (例如, 增加波特率两倍, 则响应时间减半)。显示平均、最小和最大时间。</p> <p>注意: 一个响应慢的输入输出设备可以减慢您的整个系统。例如, 如果您的输入输出设备的响应时间为 2000 毫秒, 在您系统中使用这个设备的数据的页面的最小更新时间将是 2000 毫秒。</p>
缓存	此字段指示是否缓存输入输出设备数据。
缓存超时	如果缓存输入输出设备, 则此字段会显示缓存超时值。数据在被丢弃和再次从输入输出设备读取之前, 数据在缓存中将保留超时期这么长的时间。只缓存读数据。
打包常数	协议中为此输入输出设备指定的当前打包常数值。
拨号连接	<p>拨号连接的状态和历史。</p> <p>成功 - 拨号成功的次数。</p> <p>失败 - 拨号失败的次数。</p> <p>总数 - 拨号的总次数。</p> <p>下一次 - 预定下次拨号的时间。</p>

另请参见[内核命令](#)[显示内核窗口](#)**暂停**

暂停内核窗口的调试输出。

语法

暂停

另请参见

[内核命令](#)

[显示内核窗口](#)

Shell

打开新的命令窗口。

语法

Shell

您可以使用命令窗口象使用主窗口一样。在主窗口显示调试跟踪数据时，命令窗口对于显示调试信息或输入命令是很有用的。你可以通过选择从 Windows 系统图标的关闭或用 [Exit](#) 命令退出命令窗口。

另请参见

[内核命令](#)

[显示内核窗口](#)

统计数据

将所有系统统计数据（使用在一般页面、驱动程序页面、表格页面、输入输出设备窗口页面）复位至 0。

语法

统计数据

可以通过此命令在 Citect 运行很长时间后复位统计数据，进而提供现在的统计数据指示（而不是 Citect 已运行的总时间内的平均值）。

注意：某些 I/O 服务器的统计数据每隔几分钟自动重置一次。

另请参见

[内核命令](#)

[显示内核窗口](#)

SysLog

显示本地 SysLog.DAT 文件。SysLog.DAT 在被显示前从磁盘被读取，但一旦被显示就不再更新。您必须关闭并且再打开窗口迫使它更新。你也可以使用上下翻页键和上下箭头在文件中移动，但不能编辑和存盘。

语法

SysLog [删除]

使用可选的“删除”键清除 SysLog.DAT 文件的内容。

注意： 这个窗口与用于显示 Citect.INI 文件的窗口一样。您可以在这个窗口中显示 Citect.INI 或 Syslog.DAT 文件，但不能同时显示这两个文件。

另请参见

[内核命令](#)

[显示内核窗口](#)

收集 Runtime 信息

Vijeo Citect 内核可以显示有关 Runtime 系统的信息。以下区域是收集信息的理想位置：

- **常规** - 关于 Vijeo Citect 的总体性能的统计和信息。例如，此画面显示内存使用情况、协议和 I/O 设备统计数据概要以及 CPU 使用情况。可通过 Page General 命令访问该项。
- **表格** - 包含有关 Vijeo Citect Runtime 数据结构的信息。此区域非常广泛并且开始时难于操控。然而，Page Table 统计数据很有深度。可通过 Page Table 命令访问该项。
- **驱动程序** - 有关在 I/O 设备上运行的单个协议的特定统计数据和信息。每一个端口都有其自己的信息画面。可通过 Page Driver 命令访问该项。
- **设备** - 与驱动程序信息类似，显示有关每一 I/O 设备的特定统计数据和信息。可通过 Page Unit 命令访问该项。

另请参见

[系统调整](#)

系统调整

Vijeo Citect 为实现最佳性能而设计，所以大多数用户不需要调整其系统。然而，特殊的环境可能需要您调整自己的系统以获得最佳性能。内核允许您定位于需要调整的区域，并且通过参数调整自身。例如，您可以通过 [Page]ScanTime 和 [Alarm]ScanTime 参数提高客户端的性能。

缓存调整

应该将缓存调整到足够大以致不会产生不必要的读取，同时要调整到足够小以便当通讯通道繁忙时不会返回旧数据。如果缓存太大，通讯通道会在一段时间内闲置因而浪费带宽。而且，如果缓存太大，Vijeo Citect 客户机可能会开始缩短读取请求周期，它将产生不必要的网络或内部拥堵。

当客户机向 I/O 服务器请求数据时就会缩短读取周期，而数据从缓存中返回，返回的速度就会非常快。客户机将处理数据（在屏幕上显示它），然后再次请求相同的数据。如果

I/O 服务器从缓存中再次返回相同的数据，客户机将再次处理相同的数据，这就是多余的并且浪费 CPU 和网络资源（传输请求和响应）。当短周期发生时，CPU 和网络负载将上升同时 PLC 通讯量将下降。

要调整缓存，必须平衡不必要的读取和短周期之间的缓存时间。下述方法假设您知道如何使用 Vijeo Citect 调试内核。

1. 关闭所有单元的缓存，使用内核中的 CACHE 命令以便不必重新编译工程。
2. 网络上只运行一个 Vijeo Citect 客户机，将 I/O 服务器上的该客户机用于测试。
3. 显示为你的系统产生正常 PLC 负载的典型画面。
4. 在内核中，使用 STATS 命令重新设置所有 Vijeo Citect 统计数据。
5. 在内核中，显示画面 "PAGE TABLE STATS"。此画面会显示各种 Vijeo Citect 任务的周期和执行时间，其中一些消耗 PLC 数据。这些称之为 "Citect n "（其中， n 为一个编号）的任务是从 PLC 获取数据并显示在屏幕上。看一下平均周期时间，从左边数第三列。假设平均周期时间是 1200 毫秒。这意味着当前画面正以 1200 毫秒收集所有 PLC 数据并显示在屏幕上。
6. 应始终将缓存时间设置成低于平均周期时间以使短周期发生几率最小化。通常，它应当少于这一时间的一半，例如，600 毫秒。
7. 将缓存时间设置为周期时间的一半（600 毫秒）。在单个客户机上可能看不到性能的提高，因为缓存只有在多客户机环境下才能提高性能。如果同时还在运行需要相同数据的趋势、报警或报表，有可能看到性能的提高。
8. 增加正在显示相同数据的另一台 Vijeo Citect 客户机。重置 STATS 并检查平均周期时间。每台新客户机都不应增加周期时间，它应该稍微有些下降。同时看一下 PAGE GENERAL，检查每台新客户机正在从缓存中进行读取，即缓存读取的百分比在增加。
9. 如果平均周期时间降到少于最初时间的一半，短周期将发生，你应该减少缓存时间直到它停止。

调整缓存是一个反复试验的过程 - 因为你改变它，读取周期时间也会随之改变。缓存时间还取决于当前 PLC 流量的大小。当前流量是动态的，因为 Vijeo Citect 将只读取当前正在运行的画面、趋势、报警和报表所需要的数据。始终应监视低负载状况下的平均周期时间，并将缓存设置尽可能低，以停止或防止短周期发生。

使用 Web 客户机

本节包含有关 Web 客户机的信息，并介绍以下内容：

[Vijeo Citect Web 客户机](#)

[系统架构](#)

[入门](#)

[选择要使用的 Web Server 平台 - IIS 或 Tomcat?](#)

[准备用于部署的工程](#)

[配置部署](#)

[实现多语言支持](#)

[Web 客户机升级问题](#)

[常见问题](#)

Chapter: 36 Web 客户端

通过 Vijeo Citect Web 客户端，可以在 Web 浏览器中查看实时的 Vijeo Citect 工程。它通过局域网连接、需要读/写访问当前生产信息的用户提供了轻松访问 Vijeo Citect 运行期间的手段。

例如，在不必进行大量下载或软件安装的情况下，高级经理可以从局域网上的任意一台计算机监控设备并访问当前生产信息。

注意： 在 64 位 Windows 操作系统（XP 64 位和 Vista 64 位），有两个 Internet Explorer 选项。一个用于 32 位模式，一个用于 64 位模式。Web 客户端必须在 Internet Explorer 32 位模式下使用。

如果在运行 Windows Vista 操作系统的同一台计算机上先启动显示客户端，再启动 Web 客户端，则会显示提醒信息。这种情况对使用 ActiveX 可执行组件的所有工程（例如，CSV 示例工程）都会发生。为了使用这些工程，必须在 Internet Explorer 中将 Web 服务器地址添加为受信任的站点。

另请参见
[系统架构](#)

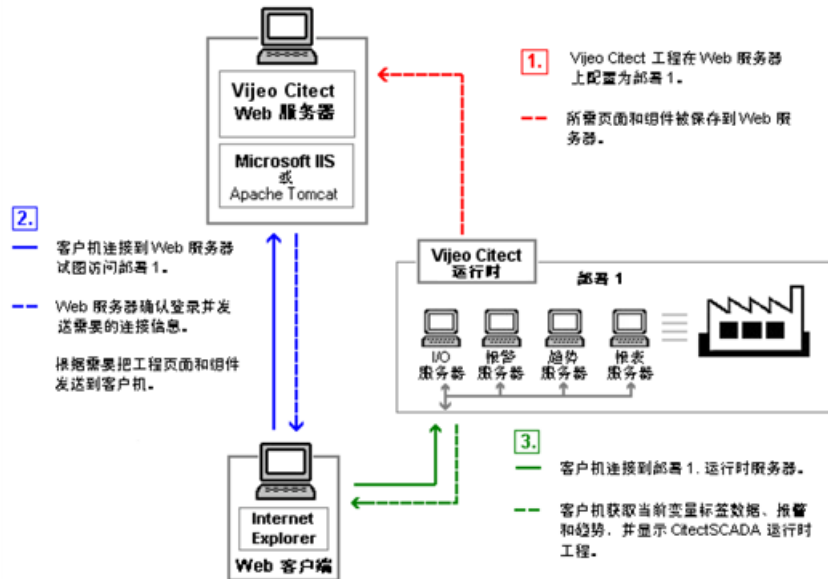
系统架构

要在互联网浏览器中显示实时的 Vijeo Citect 工程，必须使用标准的、基于 Web 的通讯协议合并工程页面的内容和当前显示在这些页面上的数据。要了解 Vijeo Citect Web 客户端的通讯架构，最简单的方法考虑以下每一组件在归档此输出量时发挥的作用：

- **Vijeo CitectWeb 服务器** - 执行系统的服务器端功能。它通过接收来自客户端的请求进行工作，并在客户端详细信息得到授权时响应客户端。然后将客户端定位到 Vijeo Citect 工程的图形和功能内容，以及运行服务器中的位置。当 Vijeo Citect 工程被配置为“部署”时，此信息将存储在 Web 服务器上。一个 Vijeo Citect Web 服务器可以包含多个部署。
- **Vijeo Citect 运行服务器（包括 I/O 服务器、报警服务器、趋势服务器和报表服务器）** - 监视物理生产设施，包含 Web 客户端显示的实时变量标签数据、报警和趋势。
- **Web 客户端** - 提供可将已部署工程的页面和内容与从 Runtime 服务器提取的原始数据相结合的平台。还需要标准 Web 技术，以便客户端可以使用 Microsoft Internet Explorer。

下图显示这些组件如何交互。

Vijeo Citect Web 客户端通讯架构。



在 Web 客户端连接到 Runtime 服务器之后，第 2 步和第 3 步成为一个连续的过程，期间，所需内容随着用户浏览工程页面而被调用。

Web 客户端使用的 citect.ini 文件设置是在连接时从 Web 服务器上的 citect.ini 文件中获取的。

此图包含在不同计算机上设置的系统组件，其唯一目的是解释通讯模型。实际上，架构的灵活性使这些组件可以被分配到任何所需配置中；它们甚至可以共享一个共用位置。

入门

Vijeo Citect Web Client 帮助的设计初衷是引导您成功完成设置 Web Client 系统的所需步骤。

有关安装和配置 Web 服务器的详细信息，请参见《Vijeo Citect 安装和配置向导》，该向导以 PDF 格式通过安装 CD 提供，或通过 Vijeo Citect 安装的 Documentation 文件夹提供。

为方便您的安装，应首先熟悉 [系统架构](#)，然后执行以下步骤，它们将合理地引导您完成正确的设置过程。

注意：要执行 Web Client 解决方案，必须首先在 Web 客户机上安装 Microsoft .NET Framework 2.0。

1. 决定要使用的 Web Server 平台

Microsoft IIS 和 Apache Tomcat 都是受支持的 Vijeo Citect Web 服务器平台。决定要使用的平台是设置系统的重要的第一步。

- [选择要使用的 Web Server 平台 - IIS 或 Tomcat?](#)
- 2. **准备用于部署的 Vijeo Citect 工程**
本部分介绍在 Web 服务器上部署之前，需要对 Vijeo Citect 工程执行的调整。
 - [准备用于部署的项目](#)
- 3. **配置部署**
本部分介绍如何通过识别工程的源位置和关联服务器来在 Web 服务器上部署工程。
 - [配置部署](#)
- 4. **多语言支持**
如果使用 IIS 作为 Web 服务器平台，可以在 Web 服务器界面上实现数个语言选项。
 - [实现多语言支持](#)

如果您已按上述步骤操作，但部署的工程似乎没有如期执行，请借助于 [常见问题](#) 部分解决可能遇到的所有问题。

选择要使用的 Web 服务器平台 - IIS 或 Tomcat

可以将以下两种技术作为 Vijeo Citect Web 服务器的平台：

- Microsoft 的 **Internet 信息服务器 (IIS)**
- The Apache Jakarta Project 的 **Tomcat Web 服务器**

设置 Web 服务器之前，应决定将使用哪种平台，因为两者的设置差异极大。

您决策的依据应为是否喜欢开放源代码解决方案（即 Apache Tomcat），或者是对其中一种技术的熟悉程度。

不过，做决定时还可以考虑以下因素。

- 建立在 Tomcat 上的 Vijeo Citect Web 服务器不支持部署的名称，Vijeo Citect 工程或文件（例如，图形画面）包含非 ASCII 字符。

注意：有关如何在其中一种平台上安装和配置 Web 服务器的说明，请参阅 [Vijeo Citect 安装和配置指南](#)。

准备用于部署的项目

在 Web 服务器上部署工程之前，您需要在 Vijeo Citect 配置环境中进行调整，以使它可用于基于 Web 的交付。为准备进行这些调整，应考虑下列事项：

- [Web 客户机平台的功能限制](#)
- [准备要交付的工程用户文件](#)
- [运行 Web 部署准备工具](#)

注意：要使用 Web 客户机，您的 Vijeo Citect 系统必须被配置为使用 TCP/IP 通信。这意味着您的工程中的网络地址必须用标准 IP 寻址方法定义。

Web 客户机平台的功能限制

由于架构需要支持基于 Web 的 Vijeo Citect 工程的执行，因此 Web 客户端未提供标准 Vijeo Citect 系统的全部功能。

如果工程性能出现下降，应考虑所访问的下列不支持的特性和 Cicode 函数。需要对这些参数进行调整。

功能限制

下列功能不受支持：

- Cicode 调试器
- 远程关闭
- 模糊逻辑
- 内核窗口
- 与 Internet Explorer 的键盘快捷键冲突的键盘快捷键。
- Web 客户端无法用作 Vijeo Citect 服务器
- 基于默认菜单画面模板的画面将仅显示以前访问的画面的按钮
- 一般模板上的**画面选择**按钮仅列出以前访问的画面
- CSV_Include 工程的更新画面菜单项将失效

注意：如果您的工程基于 CSV_Include 模板，则必须创建一个用于从菜单条访问画面的自定义菜单。

Cicode 函数的限制

有几种 Cicode 函数不能用于 Web 客户机，或者其能力受限制：

集群函数	不支持集群功能
DebugBreak	不支持 Cicode 调试器
DelayShutdown	不支持编程关闭
FTP 函数	所有 FTP 函数均不支持
模糊逻辑函数	已因大小原因从控制中删除
KerCmd	不支持内核窗口
ProjectRestartGet	不支持编程关闭
ProjectRestartSet	"
ProjectSet	"
Shutdown	"

ShutdownForm	"
SwitchConfig	配置环境不可用
TraceMsg	不支持内核窗口
UserCreate	必须在编译和自动部署工程的计算机上对用户配置文件进行更改。
UserCreateForm	"
UserDelete	"
UserEditForm	"
UserPassword	"
UserPasswordForm	"
GetWinTitle	仅限主窗口之外的窗口
WinFree	"
WinMode	"
WinMove	"
WinPos	"
WinSize	"
WinTitle	"
WndShow	"
WndViewer	调用多媒体应用程序。不支持的功能

另请参见
[准备要交付的工程用户文件](#)

准备要交付的工程用户文件

如果您的 Vijeo Citect 工程中包含用户创建的文件（如 DBF 文件、HTML 文件或 CSV 文件），您必须手动将这些文件放入名为 **Misc.zip** 的特殊压缩文件才能将其交付到 Web 服务器。与之类似，如果工程中包含 ActiveX 项目，也必须将它们加入名为 **ActiveX.zip** 的压缩文件。

要准备任何要部署的用户创建的文件，请执行以下操作：

1. 识别与要部署的工程关联的用户创建的所有文件。
 这些文件可能包括与工程页面上显示的表关联的 CSV 或 DBF 文件，或 HTML 内容。
2. 使用压缩工具将这些文件压缩到一个名为 **Misc.zip** 的文件中。

3. 将 Misc.zip 放置在工程的主文件夹中。例如，在范例工程中，这将是：
[User]\Example

注意：如果您的工程中有使用 ActiveX 对象的包含工程，确认将这些对象压缩到此包含工程的目录下的 Activex.zip 文件中。

现在，可以在 Web 服务器上部署这些文件了。

要准备任何要部署的包含 ActiveX 对象，请执行以下操作：

1. 识别与要部署的工程关联的所有 ActiveX 对象。
2. 使用压缩工具将这些文件压缩到一个名为 **ActiveX.zip** 的文件中。
3. 将 ActiveX.zip 放置在工程的主文件夹中。例如，在 CSV_Include 工程中，这将是：
[User]\CSV_Include

注意：如果 ActiveX 对象具有关联的数据源，确认托管 Web 客户端的计算机可以找到该数据源。请参见 [管理相关数据源](#) 主题，该主题位于 Vijeo Citect 用户指南帮助中有关 ActiveX 对象的部分下面。

另请参见

[运行 Web 部署准备工具](#)

运行“Web 部署准备”工具

准备要部署的工程的最后一步是通过“Web 部署准备”工具运行工程。这样会刷新已编译的工程，并为基于 Web 的传递创建所需的文件和目录。

如何通过“Web 部署准备”工具运行工程：

1. 确认要部署的工程为传递压缩了所有关联的用户文件和 ActiveX 对象（请参见 [为传递准备工程的用户文件](#)）。
2. 在 Citect 浏览器中查找要部署的工程，然后刷新编译。
3. 转到 Citect 浏览器的 **工具** 菜单，并选择 **Web 配置准备**；或者单击浏览器工具栏上的以下图标：



4. 此时将显示进度指示器。工程的大小对此进程花费的时间长短影响很大；文件很多的大型工程处理时间会超过十分钟，这取决于硬件。（如果需要，可以中断部署准备。）
5. 准备完成后，会出现一个对话框，表明准备成功。单击 **确定**。
工程此时已对在 Web 服务器上进行部署准备就绪。如果更改工程，则必须刷新编译并再次运行“Web 部署准备”工具。

注意：在编译工程时，可以自动运行 Web 配置准备工具。要执行此操作，请转到 Citect 工程编辑器的工具菜单并选择选项。选择准备 Web 部署选项并单击确定。请注意，这样会增加每个编译花费的时间，尤其是对大型工程。

配置部署

部署代表 Vijeo Citect 工程在 Web 服务器上的实施。它会结合显示工程所需的文件和组件，并记录生成 Vijeo Citect 运行时数据的服务器的位置。

Web 服务器上配置的部署在 Web 客户机主页上列出，当您初次登录时该页面会显示。通过点击部署名称左侧的小加号 (+) 图标可以显示部署的配置详细信息。

可以对部署实施的动作类型取决于您的登录权限。例如，如果您以仅查看客户端登录，则只能查看部署。如果您是管理员，则可以编辑部署和创建新部署。

下面的列表说明了主页上显示的每个图标的关联功能：

	添加新部署 - 带您进入部署配置页面，在其中可以创建新部署（仅限管理员客户）。
	帮助 - 显示该帮助页面，说明如何配置和使用 Web 客户机。
	编辑部署 - 带您进入部署配置页面，以便编辑选定部署（仅限管理员客户）。
	删除部署 - 删除选定部署（仅限管理员客户）。
	启动控制客户端 - 使用控制客户端权限显示选定部署（仅限控制客户端和管理员客户端）。
	启动仅查看客户端 - 使用仅查看客户端权限显示选定部署

此外，系统消息面板提供影响 Web 服务器当前状态的事件通知。

注意：Web 客户机使用的 citect.ini 文件设置是连接时从 Web 服务器上的 citect.ini 文件获取的。其中包括 Web 客户机可以访问哪些簇。要在使用 Web 客户机查看的工程中切换簇，请使用函数 ClusterActivate 和 ClusterDeactivate。

另请参见

[准备用于部署的工程](#)

[创建新部署](#)

[从 Vijeo Citect 中部署工程](#)

[显示部署](#)

[编辑现有部署](#)

[更新部署以反映工程更改](#)

[删除部署](#)

创建新部署

要在 Web 服务器上部署一个 Vijeo Citect 工程，必须以管理员权限登录到 Web 客户端。这将确保您能访问主页的所有功能。

添加新部署

1. 点击**添加新部署**图标。



随即会显示“部署配置”画面。

2. 在**部署**文本框中键入名称，如果需要，输入**描述**信息。部署名称不能包含以下字符：
\
*
?
|
.
/
"
'
:
;
<
>

&

注意：如果已经升级了 Web 客户端版本，仍可以查看之前使用 6.1 版或更早版本的 Web 客户端创建的原有部署。有关详细信息，请参见 [Web 客户端升级注意事项](#)。

3. 标识**工程路径**字段中的 Vijeo Citect 工程内容的来源。
如果工程位于 Web 服务器本地，可以使用标准路径地址。路径必须直接指向 Vijeo CitectUser 目录。例如，MyExample 工程的位置应为：

```
[User]\MyExample
```

注意：如果远程管理 Web 服务器并且使用本地路径地址，请确认该路径表示工程在 Web 服务器计算机上而不是在当前正使用计算机上的位置。

如果工程不在 Web 服务器上，则需要使用标识网络主机和能够找到它的目录的 UNC 地址。例如：

```
\\ComputerName\<path to application data>\User\MyExample
```

注意：必须共享工程所在目录，以允许 Web 服务器访问它。理想状态下，应从目录创建共享（例如，称为 WebShare），然后使用以下工程路径：

```
\\计算机名称\WebShare
```


记住，如果试图从远程计算机上访问工程目录，“本地”管理员登录不会在另一台计算机上提供相应的访问权限。应使用网络用户配置文件，它能够被相同域中的其他计算机所识别。

4. 确定是否存在任何与工程相关的 I/O、报警、报表或趋势服务器受到防火墙的保护。如果存在，则需要与防火墙管理员确认 Vijeo Citect 端口是否已经打开并允许直接访问，或者防火墙已经正在使用地址转发。如果正在使用转发，则需要通过在**服务器**字段中按照下列格式键入名称来标识每台服务器：

```
<ClusterName>.<ServerName>
```

如果在报警服务器上启用了报警属性，则需要将报警属性连接器配置为单独的服务器，以便让 Web 服务器知道哪个端口正在运行。要在**服务器**字段键入报警服务器的名称，请使用以下格式：

```
<ClusterName>.<AlarmServerName>_AlarmProps
```

例如：

```
ClusterOne.AlarmServerOne_AlarmProps
```

按照防火墙管理员提供的信息，为每个服务器键入**地址**和**端口**。

注意：Web 客户端会将通过此种方式重定向的任何服务器自动添加到本地 citect.ini 文件的 [AddressForwarding] 部分。有关详细信息，请参见 [使用地址转发](#)。

可通过选择**添加新服务器**图标将其他服务器添加到列表。

5. 显示部署时，可使用**客户端控制**文本框指定 Web 客户端组件的特定版本的使用法。菜单中列出了所有当前安装在 Web 服务器上的 Web 客户端控制的不同版本。通常情况下，应当选择与您编译的 Vijeo Citect 工程相匹配的控制版本。
6. 点击**应用更改**。



这一步很重要，如果直接跳回到主页，所做的更改将会丢失。

所有工程文件都是从指定的路径获取，并复制到 Web 服务器以备 Web 客户端访问。一旦设置完成，有关工程大小的信息即会显示在“工程路径”字段上方的“文件路径”栏内。左边的数字表示工程中包含的文件的个数，右边的数字表示工程大小。

保存部署。通过点击主页图标返回到 Web 客户端主页后，即会列出您的新部署。

另请参见

[从 Vijeo Citect 中部署工程](#)

[显示部署](#)

从 Vijeo Citect 内部部署工程

通过 Web 客户端架构可以从 Vijeo Citect 配置环境内部部署工程，而不必使用 Web 客户端界面设置系统。

此过程要求在 Citect.ini 文件里调整两个参数：

- [WebServer]WebClientCab
- [WebServer]DeployRoot

这些参数可识别出用于工程的客户端组件和部署根目录的位置。在工程已编译且准备进行部署后，它将被直接置于 Web 服务器上。

注意：

- 如果已经将 Web 客户端升级到版本 7，您仍然可以查看遗留的部署。有关详细信息，请参见 [Web 客户端升级问题](#)。
- 在应用此选项时，请注意 citect.ini 文件的配置，因为这些参数的错误非常难于诊断。为了避免输入错误，请使用 Web 部署工具，该工具位于 Web 服务器的 Web 部署图形用户界面上的“Citect 工程管理器”工具栏中。
- 如果工程名称包含非 ASCII 字符，则从 Vijeo Citect 内部进行部署时可能会不成功并显示错误。在这种情况下，请使用 Web 服务器界面来创建部署。

从 Vijeo Citect 内部部署工程：

1. 确定已对 Vijeo Citect 系统进行了配置，使其使用 TCP/IP。如果运行计算机设置向导，那么网络页将识别出正在使用的通信协议。
2. 调整 citect.ini 文件内的 [WebServer]DeployRoot 参数。此参数代表部署在 Web 服务器上所在的目录。
如果已经设置了基于 IIS 的 Web 服务器，那么缺省位置是已安装目录内的“Deploy”目录。例如：

```
[webserver]
DeployRoot="C:\Program Files\Schneider Electric\Vijeo Citect\Vijeo Citect 7.10\WebServer\deploy"
```

如果您使用的是 Apache Tomcat Web 服务器，则此目录是：

```
[webserver]
DeployRoot="C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps\Vijeo Citect\deploy"
```

注意： 在设置 [WebServer]DeployRoot ini 参数时，路径的最后一个子文件夹的名称必须包含“deploy”，否则部署可能不成功。如果是从 Windows 2000 系统部署到网络位置，则请使用映射的驱动器，而不要使用 UNC 地址。不要将驱动器直接映射到部署位置，因为路径必须以“deploy”子文件夹结尾。

3. 调整 citect.ini 文件内的 [WebServer]DeployRoot 参数。此参数代表目录路径和部署运行时的客户端组件，它与已安装的客户端目录相关。例如：

```
[webserver]
WebClientCab=700/CitectSCADAWebClient_7_0_176.cab
```

请注意，本例在定义的路径中使用了正斜杠。

4. 编译工程并准备对其进行部署。转到 Citect 工程管理器的**工具**菜单，并选择 **Web 配置准备**或在“浏览器”工具栏上选择以下图标。



您的工程应立即显示为下一次登录到 Web 客户端主页上时的一个部署。

注意：在编译工程时可以自动运行 Web 配置准备进程。要执行此操作，请转到 Citect 工程管理器的**工具**菜单并选择**选项**。选择**预备网络配置**选项并单击**确定**。但请注意，这可能增加编译工程所需的时间。

5. 确定是否存在任何与工程相关的 I/O、报警、报表或趋势服务器受到防火墙的保护。如果存在，则需要与防火墙管理员确认 Vijeo Citect 端口是否已经打开并允许直接访问，或者防火墙已经正在使用端口转发。
如果正在使用端口转发，则将需要以管理员身份登录 Web 客户端，选择工程，然后单击**编辑部署**按钮：



将转到部署配置页面。

6. 在**服务器**字段键入名称，确定每个服务器的端口转发是否正在使用，请使用以下格式：

```
<ClusterName>.<ServerName>
```

如果在报警服务器上启用了报警属性，则需要将报警属性连接器配置为单独的服务器，以便让 Web 服务器知道哪个端口正在运行。要在**服务器**字段键入报警服务器的名称，请使用以下格式：

```
<ClusterName>.<AlarmServerName>_AlarmProps
```

例如：

```
ClusterOne.AlarmServerOne_AlarmProps
```

7. 键入每个服务器的**地址**和**端口**，这些信息应由防火墙管理员提供。

注意：Web 客户端将对以此方式被重定向到本地 citect.ini 文件的 [AddressForwarding] 部分的所有服务器进行自动添加。有关详细信息，请参见 [使用地址转发](#)。

通过选择**添加新服务器**图标可以向列表中添加其他服务器。

另请参见
[显示部署](#)

显示部署

在显示部署时，将从 Web 服务器下载必需的 Web 客户端组件文件，以便可以在 Web 浏览器中运行相关的 Vijeo Citect 工程。

注意： Web 客户端使用的 citect.ini 文件设置是在连接时从 Web 服务器上的 citect.ini 文件获取的。其中包括 Web 客户端可以访问哪些集群。要在使用 Web 客户端查看的工程中切换集群，请使用函数 ClusterActivate 和 ClusterDeactivate。

如何显示部署：

1. 在可用的部署列表中定位想要显示的部署。
2. 点击相应的图标（**启动控制客户端**或**启动仅查看客户端**）以显示部署。



可用的显示选项根据您的登录权限会有所不同。如果选择“仅查看客户端”图标（带有金色锁的图案），则只能读取 Vijeo Citect 工程的当前值。

在需要的工程文件和组件下载完成后，将显示 Vijeo Citect 工程。现在，您可以按要求在工程页面中进行导航了。

注意：在下载完新的或更新的组件文件（.cab 文件）后，如果客户端计算机上的当前用户不具备 Windows 管理员权限，则可能出现一条提醒消息。如果运行新的部署或者需要下载更新的 .cab 文件，请确认当前 Windows 用户具有管理员权限。

另请参见
[编辑现有部署](#)

编辑现有部署

如果需要，您可以对部署的设置进行编辑。例如，您可以更改某部署的名称，或者为运行时服务器指定新地址。

要编辑某部署的设置，您必须以管理员客户的身份进行登录。

要编辑现有部署，请执行以下操作：

1. 在可用部署列表中，选择要编辑的部署。
2. 点击**编辑部署**图标。



您将进入“部署配置”画面。

按需要对字段进行更改。有关字段的说明，请参见 [创建新部署](#)。

注意：如果您为某个部署设置了新名称，它将与旧名称并存，而不会进行更新及覆盖。这使您能够轻松地复制现有部署；但是，如果初始部署不再需要旧名称，为了避免混淆，您应将其删除。

3. 点击**应用更改**。（这一点非常重要，因为如果您直接跳转回主页，则所做的更改将丢失。）



Web 服务器将检索 Vijeo Citect 工程的全新画面和组件，包括任意最新的更改。

另请参见
[更新部署以反映工程更改](#)

更新部署以反映工程更改

如果更改了源 Vijeo Citect 工程，您必须更新其相关部署，以将这些更改发布到在 Web 服务器上。

部署的更新将最新的工程页面和组件上传到 Web 服务器进行分发。这一点很重要，因为在工程画面和从运行时服务器轮询的数据（如果内容不是最新的）间可能存在差异。

如何更新部署：

1. 确认使用 Web 部署准备工具在 Vijeo Citect 中编译并处理了要更新的工程。请参见 [运行“Web 部署准备”工具](#)。
2. 选择要更新的部署。
3. 点击**编辑部署**图标。



您将进入“部署配置”画面。

4. 点击**应用更改**。



Web 服务器将检索 Vijeo Citect 工程的全新画面和组件，包括任意最新的更改。

另请参见
[编辑现有部署](#)
[删除部署](#)

删除部署

要从 Web 服务器中删除部署，必须以管理员客户的身份登录。

要从 Web 服务器中删除部署，请执行以下操作：

1. 选择要从可用部署列表中删除的部署。
2. 点击**删除部署**。



会出现一个对话框，要求您确认是否要删除此部署。点击**确定**。

另请参见
[配置部署](#)

实现多语言支持

Web 客户机部署配置界面可以使用英语之外的语言显示。缺省支持以下语言：

- 法语
- 德语
- 西班牙语
- 中文
- 日语
- 韩文

还可通过翻译定义所显示文本的资源文件来实现其他语言。对于上面所列的语言，该文件已经存在每种语言的翻译版本，存储在安装的 `locales` 文件夹中。

另请参见
[如何设置缺省语言](#)
[使用不同于当前系统区域设置的语言](#)
[设置非缺省语言](#)

如何设置缺省语言

当一台客户计算机连接到Web服务器时，Web 页面中的脚本会自动检测当前定义为浏览器缺省值的 [语言代码](#)。此代码通过客户机上**控制面板 | 区域和语言选项**中定义的系统区域设置来绘制。

一旦确定了浏览器的语言代码，脚本就会尝试将它与 Web 服务器上可用的语言代码进行匹配。如果匹配成功，该关联语言会自动应用于 Web 客户机部署配置界面。如果匹配不成功，缺省设置为英语。

例如，如果 Windows **区域设置**为中文（中国），则您浏览器的语言代码就会设置为 "**zh-cn**"。下面对照显示的是 Web 服务器上的当前语言代码列表，缺少设置如下：

语言	Windows 语言代码
英语	en

语言	Windows 语言代码
法语	fr
德语	de
西班牙语	es
简体中文	zh
日语	ja
韩文	ko

如果未能找到匹配 "zh-cn" 的语言，脚本会尝试装载简体中文语言 "zh" 作为匹配。界面也将自动以中文显示。

另请参见

[使用不同于当前系统区域设置的语言](#)

使用不同于当前系统区域设置的语言

可以使用不同于计算机当前系统区域设置的语言显示 Web 客户机部署配置画面中的内容。方法是：在浏览器的地址字段中使用 URL 查询字符串。

如何切换到另外一种语言：

1. 确定要使用的语言及其相关语言代码。（请参见 [如何实现默认语言](#)，以获取 Web 服务器支持的默认语言的代码列表。
例如，如果您要使用中文，所需的代码应为 **zh**。

注意： 如果要使用的语言不受支持，则必须创建并转译您的消息文件。请参见 [实现非默认语言](#)。

2. 使用 URL 查询，表示要用于 Web 客户机部署画面的语言。例如，如果浏览器的当前地址字段中为：

`http://localhost/VijeoCitect`

则添加 `"/?lang="` 查询到地址的末端。例如，中文为：

`http://localhost/VijeoCitect/?lang=zh`

注意： 如果使用的代码代表默认语言的区域变量，且该特定代码没有匹配项，则 Web 服务器只能实现可用的默认语言版本。例如，如果使用语言代码为简体中文 "zh-cn"，则简体中文使用 "zh"。

现在，您可以使用适当的语言显示 Web 客户机的部署配置画面。

另请参见
[设置非默认语言](#)

设置非缺省语言

如果需要在 Web 客户端的部署配置界面上使用不同于 Web 服务器所支持缺省语言之外的语言，可以使用您自己翻译的定义所显示文字的消息文件。

显示缺省支持语言之外的语言：

1. 使用支持要编辑语言的文本编辑器，打开位于 Web 服务器的 locales 目录中的一个已存在消息文件；缺省路径为：
C:\Program Files\Schneider Electric\Vijeo Citect\Vijeo Citect 7.10\Web Server\locales
打开的文件应包括容易翻译的语言。每个文件开头的语言代码可用来标识每个文件表示的语言；例如，英语文件名为 **enmsg.xml**。
2. 使用名称中相应的语言代码将该文件保存回 locales 目录。
要正确命名该文件，请检查 [Windows 语言代码](#) 获取相应的代码。如果您的翻译资源文件 (XXmsg.xml) 与当前系统区域设置相符，则在启动 Web 客户端主页时，会自动装载该翻译资源文件。
例如，要在 Web 客户端的配置画面上使用希伯来语，则应将您的文件命名为 **hemsg.xml**。要使用台湾繁体中文，应将该文件命名为 **zh-twmsg.xml**。
3. 接下来更改文件的内容。首先，设置正确的编码格式。
编码格式在该文件的第一行定义，其形式如下：
`<?xml version="1.0" encoding="iso-8859-1" ?>`
如果语言使用英语字符，使用的格式应是 ANSI，它定义为 "iso-8859-1"（参见上面的示例）。
如果语言使用非英语字符，应使用 Unicode，它定义为 "UTF-8"（参见下面的示例）。
`<?xml version="1.0" encoding="UTF-8" ?>`
4. 现在，翻译出现在 Web 客户端界面上的文字。
需要翻译的内容在文件中分为两个部分：“标签”和“消息”。标签部分包含用来描述和标识界面元素的内容；消息部分包含出现在系统消息面板中的通知。
要翻译这两部分，请改变 XML 分隔标记之间的文字。但不要改变这些标记本身。
XML 标签可定义每个标签的使用位置。

注意： 确保保留任何 "%" 字符，因为它们用来插入系统信息。

例如，英文文件：
`<!-- Labels -->`
`Vijeo Citect Web Client Deployment`
`System Messages`
`Deployment`
`Description`


```

<span id="ACTION">Action</span>
<!-- Messages -->
<sysmsg id="DELOK">% deleted.</sysmsg>
<sysmsg id="DELCAN">% will NOT be deleted.</sysmsg>
<sysmsg id="DEPNUL">You can't % an empty deployment.</sysmsg>
...
  将以西班牙语显示如下:
<!-- Labels -->
<span id="TITLE">Despliegue del Cliente Web vijeo Citect</span>
<span id="SYSMSG">Mensajes del Sistema</span>
<span id="DEP">Despliegue</span>
<span id="DESC">Descripción</span>
<span id="ACTION">Acción</span>
<!-- Messages -->
<sysmsg id="DELOK">% eliminado.</sysmsg>
<sysmsg id="DELCAN">% NO será eliminado.</sysmsg>
<sysmsg id="DEPNUL">No puede % un despliegue vacío.</sysmsg>

```

翻译完该文件并用适当的名称将它保存到 `locales` 文件夹后，Web 服务器将能够支持该语言。

注意：保存文件时，确保您使用的文本编辑器根据所用的语言编码将该文件保存为正确的格式，即 ANSI 或 Unicode (UTF-8)。请参见上述步骤 3。

Web 客户机升级注意事项

如果您已经将您的 Web 客户机从较早版本升级到了版本 7.0，请阅读下列部分，了解如何安装和如何在现有部署下使用升级的 Web 客户机工具。

安装

如果计划在 Windows 2000 或 Windows 2003 Server 上使用 Web 客户机，您必须首先在计算机上安装最新的 Windows Installer 模块。这一步骤可通过 Windows 2000 或 Windows 2003 Server 上的 Windows Update 功能完成。

升级到 Web 客户机的新版本会在 WebServer 文件夹的 `client` 文件夹中添加一个名称为 700 的新文件夹。

如果计划安装 Web 客户机版本 7.0 但希望查看传统部署（即版本 7.0 之前），则在安装新的版本前，请备份旧的部署到永久存储介质并将它放在安全的位置。然后，在安装新版本之后，将旧的部署复制回 `deploy` 文件夹（见上文），将传统 `.cab` 文件复制到 `client` 文件夹中的相应文件夹中；这将使您的旧部署可用。

创建新部署

创建新的部署时，请注意您针对部署使用的 .cab 文件（**客户机控制**）必须对应于要访问的工程的正确版本，否则部署不会成功。例如，要创建基于版本 6.0 Vijeo Citect 工程的部署，请从**客户机控制**菜单中选择 **600/filename.cab**；要创建基于版本 7.0 Vijeo Citect 工程的部署，请从**客户机控制**菜单中选择 **700/filename.cab**。

从 Vijeo Citect 中部署工程

如果从 Vijeo Citect 中部署项目，请编辑 citect.ini 文件中的 [webservice] 部分，针对正在使用的 Web 客户端的版本指定正确的 cab 文件。例如，对于版本 7.0 部署，指定位于 700 文件夹中的 .cab 文件；对于 6.0 版本，指定 600 文件夹。

常见问题

本节解答关于 Web 客户端的常见问题。一部分专门关于 Windows 2003 Server 的注意事项，另一部分关于常规注意事项：

- [Windows 2003 Server 相关注意事项](#)
- [常规注意事项](#)

Windows 2003 Server 相关注意事项

此节介绍了有关 Windows 2003 Server 产品的注意事项。

问：Web 客户端部署画面在 Windows 2003 Server 上显示不正确：“显示服务器详细信息”缺失，开始客户端、删除部署和编辑部署图标也缺失。如何修正此问题？

答：有两种可能答案：

- 当安装 IIS 6.0 时，它默认设置为“锁定”模式，这意味着它只能服务静态内容。ASP、ASP.NET 和 FrontPage 服务器扩展都被禁用，您必须明确地分别将它们启用。Vijeo Citect Web 服务器需要在 IIS 上启用 ASP。

在 Windows 2003 Server 上启用 IIS6 的 ASP：

1. 选择**开始 | 控制面板**，然后双击**添加或删除程序**。
2. 在“添加或删除程序”对话框中，单击**添加/删除 Windows 组件**。
3. 在“Windows 组件向导”对话框中，选择**应用程序服务器**，然后单击**详细信息**。
4. 在“应用程序服务器”对话框中，选择**Internet 信息服务(IIS)**，然后单击**详细信息**。
5. 在“Internet 信息服务 (IIS)”对话框中，选择**万维网服务**，然后单击**详细信息**。
6. 在“万维网服务”对话框中，选择**当前服务器画面**选项。
7. 在 Windows 2003 Server 中，缺省设置是使除了 localhost 之外的所有 Web 位置都成为非受信站点。因此您必须修改浏览器的安全设置。

更新 Windows 2003 Server 的受信任站点设置：

1. 选择**工具 | Internet 选项**。
2. 单击**安全性分页标签**，然后单击**受信任站点 | 站点**。
3. 在**将该网站添加到区域中**字段中，添加 Web 服务器的 IP 地址如下：

http://<ip address>

注意： 只要 Web 客户端页面没有正确地装载、显示或执行，您都应验证您的安全设置，即使运行的不是 Windows 2003 Server。

问： 当我尝试启动 Web 客户端时，我得到提醒信息“启动 Citect Web 客户机失败：无法初始化 citect 系统”，然后 Web 客户端浏览器窗口对我的输入停止响应。如何修正此问题？

答： 首先检查您没有在无意中删除安装的 Web 服务器目录中的 #DisplayClient 文件夹，因为这将产生此错误。缺省情况下，此目录位于：

```
C:\Program Files\Schneider Electric\Vijeo Citect\Vijeo Citect
7.10\WebServer\deploy\#displayclient
```

如果不是这种情况，这是由于 MIME 设置不正确：初始化文件在 Windows 2003 中没有被识别为注册的文件扩展名。要纠正此问题，可添加正确的 MIME 类型扩展名，方法如下：

1. 运行 IIS 管理器，转到**网站|缺省网站|部署**。
2. 从文件夹的右键单击菜单中选择**属性**。
3. 转到**HTTP 头|MIME 类型**。
4. 在“扩展名”字段中，输入“.*”。
5. 在“MIME 类型”字段中，输入“all”。
6. 重新启动 Web 服务器和客户机。

常规注意事项

本节介绍与 Web Client 产品相关的一般注意事项。

问： 当尝试在 Internet Explorer 中运行部署时，我得到了以下错误消息：“此画面存在的问题可能导致其无法正确显示...”。原因是什么？

答： 在首次尝试运行部署时，如果与之关联的客户端组件 (.cab 文件) 尚未出现在本地机器上，Citect 会试图下载该组件。如果客户端组件的下载操作被触发，而当前登录的用户没有客户端中的 Windows 本地管理员权限，则会出现此提醒信息。

解决办法是确认第一次运行部署的人是客户端的 Windows 本地管理员。在组件被下载后，任何用户都将能够访问和运行部署。除非 .cab 文件被更新，否则此提醒信息会再次出现。

问： 我在 Vijeo Citect 中用合适的 citect.ini [WebServer] 参数部署了一个工程，但此工程未出现在 Web 服务器上的部署列表中。一个对话框通知我此部署已成功。发生了什么？

答： 如果您在使用 [WebServer]DeployRoot 参数语法时犯错，可能会出现此情况。例如，如果您使用圆括号替代方括号（例如“[WebServer]DeployRoot”），编译器将无法读取此参数，部署文件将被发送到 Vijeo Citect 工程目录：

```
[User]\<Project Name>
```

部署被标记为成功，但 Web 服务器却无法找到它。

如果您已部署工程但它没有出现在 Web 服务器的列表中，请检查上述位置中是否有一个名为“Web Deploy”的子文件夹。如果这个文件夹存在，则 `citect.ini` 文件中所用的语法是正确的。

问：我将某用户从配置了 Web 服务器访问权的用户列表中删除了，但他仍能登录。如何拒绝他访问？

答：某些时候，即使用户帐户已被删除，相应的用户仍能连接到 Web 服务器。当操作系统没有立即向 Web 服务器报告用户删除时可能发生这种情况。删除用户与限制访问之间需要大约半小时。

解决方法是在删除用户前，拒绝他们的全部访问权限。这样，他们将无法获得访问权。参见《Vijeo Citect 安装和配置指南》的“Web 客户端”部分中的“删除用户帐户”主题。

问：在首次尝试运行 Web Client 组件时，我得到了“系统设置改变”消息，指示我重新启动计算机。应该怎么办？

答：此信息会出现在包含了某些旧版本系统文件（Web 客户端控制需要）的计算机上。如果这些文件在安装时正在被其它应用程序所使用，就会出现“系统设置改变”消息。点击**确定**重启计算机，以便所需文件的较新版本在系统重启过程中安装。除非其他应用程序重新加载这些旧文件，否则此提醒信息还会再次出现。

问：我的系统中包含的一个 ActiveX 对象无法找到其关联数据源。从哪可以找到？

答：如果 ActiveX 对象具有关联的数据源，则需要确认托管 Web 客户端的计算机可以找到该数据源。有关详细信息，请参见主题 [管理关联数据源](#)。

问：在我试图通过 Web Client 部署配置画面编辑部署时，出现了提示“客户机控件 (CitectSCADAWebClient_7_0_xxx.cab) 不在选项列表中”的弹出式窗口，为什么？

导致此对话框出现的原因有以下两个：

1. 您可能错误地设置了 [Web 服务器] 参数 `WebClientControl`。网页可能未识别出 `.cab` 文件的名称或版本。还应检查是否已在 `cab` 文件名前使用了正斜杠，因为这是 Web 服务器查找正确文件所必需的。
2. 用户可能已从 `C:\Program Files\Schneider Electric\Vijeo Citect\Vijeo Citect 7.10\WebServer\client\700`（或指定的位置）删除了所需 `.cab` 文件。因此，网页无法找到该文件。

问：在我翻译了语言源 DLL 之后，过程分析器界面以外语正常显示，但它在 Web Client 平台上却显示为英语。如何修正此问题？

答：在 Vijeo Citect Web Client 中运行的过程分析器也支持运行期语言切换，但您必须配置 Web Client 要下载到客户机的语言。

要配置要下载的语言，请执行以下操作：

1. 在 `C:\Program Files\Schneider Electric\Vijeo Citect\Vijeo Citect 7.10\WebServer\client\700` 中创建一个名为 `bin.zip` 的压缩文件。

2. 将此压缩文件添加到希望客户机下载和使用的语言源 DLL 文件。（您可以在 \Program Files\Common Files\Schneider Electric Shared\Vijeo Citect 文件夹中找到此文件。）

注意： bin.zip 文件及其内容不能进行版本检查。这意味着如果您的服务器包含更新的 bin.zip 文件，您必须从 Web Client 机器中手动删除 bin.zip。为此：

1. 在 Web Client 机器上查找 Analyst.dll 文件的安装目录，然后在此目录中查找名为 **bin.zip** 的文件。
2. 删除此文件。
3. 重新连接到 Web 服务器以下载最新的 bin.zip 文件。

问：我在自己的 Vijeo Citect 工程中配置了键盘快捷键，但当工程是在 Web Client 中配置的时候，它们无法正常工作。这是为什么？

答：针对 Internet Explorer (IE) 配置的键盘快捷键优先级高于在 Vijeo Citect 工程中配置的键盘快捷键。例如，范例工程将 F11 指定为调用有关图形画面上选定动点的帮助。如果工程作为 Web Client 部署运行，F11 会将视图切换至全屏，正常情况下，这与在 IE 中相同。

这是使用 Internet Explorer 托管 Vijeo Citect 工程的局限。最简单的解决方法是返回 Vijeo Citect 配置环境并指定快捷键，以防止冲突发生。有关预配置的键盘快捷键的详细信息，请参见 Internet Explorer 帮助。

问：我无法从 Web Client 打印。为什么？

答：您可以从 Web Client 打印，但不能使用浏览器的文件 | 打印命令执行该操作。而是应该在 Vijeo Citect 工程中创建一个使用 Cicode WinPrint() 函数打印所需画面的“打印”控件。

问：我添加到 Vijeo Citect 工程中的新画面未显示在 Web Client 的画面选择列表或缺省菜单画面中。如何修正此问题？

答：如果您添加到 Vijeo Citect 工程中的画面未显示在 Web Client 中，您可以在画面选择列表中手动键入此画面名称，以查看这一新画面。在这一版本的 Web Client 中，新画面不会被添加到缺省菜单画面中。

问：Web Client 任何处理用户工程需要的 ActiveX 控件（例如 CiMeterX.ocx）和用户文件（例如 Recipes.dbf）？

答：如果您的用户工程需要此类文件，则需要创建特殊的压缩文件以将其包含其中。创建 ActiveX.zip 文件以包含工程需要的 ActiveX 文件，创建 Misc.zip 文件以包含工程需要的其他文件；例如 recipes.dbf、Chinese.dbf、Japan.dbf 等等。将这些文件添加到主工程路径下（例如 [User]\Example）。

注意：可以在压缩文件中包含子文件夹，但工程必须被配置为使用同一相关路径结构。

编译期间，Vijeo Citect 工程需要的支持文件所在的任何压缩文件都将被复制到 Webdeploy 子文件夹中。启动时，Web Client 会检查所有压缩文件的时间标签，以确定压缩文件是否已更新；如果这些文件已更新，将下载压缩文件。

问：我的工程是用 CitectFacilities 创建的，并且纳入了 Citect TimeScheduler。当我在 Web Client 上运行此工程时，TimeScheduler 无法工作。这是为什么？

答：如果要在 Vijeo Citect Web Client 上运行时间调度程序，您必须确认用来登录的用户个人资料对配置工具和配置文件的位置拥有适当的网络访问权限。用户必须能够执行配置工具并写入配置文件。

问：Web Client 部署画面错误地显示在 Windows 2000 Advanced Server 上。“显示服务器详细信息”缺失，开始客户机、删除部署和编辑部署图标也缺失。这是为什么？

答：这可能是因 Windows 自动更新在安装操作系统后一次安装了数个组件引起的。尽管 Internet Explorer 可能已升级到最近的版本（如 6.0.2800.1106），但它可能仍作为第 5 版浏览器运行；例如，它只能对“iframe”提供有限的支持。如果您从帮助菜单中调用**关于 Internet Explorer**，第 5 版风格的对话框将以第 6 版的编号显示，您的计算机将以这种方式受到影响。

完整卸载/重新安装 Internet Explorer 可以修正此问题。

Chapter: 37 Windows 语言代码

代码	Windows 区域设置	代码	Windows 区域设置
af	南非荷兰语	hu	匈牙利语
sq	阿尔巴尼亚语	is	冰岛语
ar-sa	阿拉伯语 (沙特阿拉伯)	id	印度尼西亚语
ar-iq	阿拉伯语 (伊拉克)	it	意大利语 (标准)
ar-eg	阿拉伯语 (埃及)	it-ch	意大利语 (瑞士)
ar-ly	阿拉伯语 (利比亚)	ja	日语
ar-dz	阿拉伯语 (阿尔及利亚)	ko	韩文
ar-ma	阿拉伯语 (摩洛哥)	ko	朝鲜语 (JoHab)
ar-tn	阿拉伯语 (突尼斯)	lv	拉脱维亚语
ar-om	阿拉伯语 (阿曼)	lt	立陶宛语
ar-ye	阿拉伯语 (也门)	mk	FYRO 马其顿语
ar-sy	阿拉伯语 (叙利亚)	ms	马来西亚语
ar-jo	阿拉伯语 (约旦)	mt	马耳他语
ar-lb	阿拉伯语 (黎巴嫩)	no	挪威语 (伯克梅尔)
ar-kw	阿拉伯语 (科威特)	no	挪威语 (尼诺斯克)
ar-ae	阿拉伯语 (阿拉伯联合酋长国)	pl	波兰语
ar-bh	阿拉伯语 (巴林)	pt-br	葡萄牙语 (巴西)
ar-qa	阿拉伯语 (卡塔尔)	pt	葡萄牙语 (葡萄牙)
eu	巴斯克语	rm	Rhaeto 罗曼语
bg	保加利亚语	ro	罗马尼亚语
be	白俄罗斯语	ro-mo	罗马尼亚语 (Moldavia)
ca	加泰罗尼亚语	ru	俄语
zh-tw	中文 (台湾)	sz	拉普兰语 (萨摩斯岛)

代码	Windows 区域设置	代码	Windows 区域设置
zh-cn	中文 (中国)	sr	塞尔维亚语 (西里尔文)
zh-hk	中文 (香港特别行政区)	sr	塞尔维亚语 (拉丁文)
zh-sg	中文 (新加坡)	sk	斯洛伐克语
hr	克罗地亚语	sl	斯洛文尼亚语
cs	捷克语	sb	索布语
da	丹麦语	es	西班牙语 (传统)
nl	荷兰语 (标准)	es-mx	西班牙语 (墨西哥)
nl-be	荷兰语 (比利时)	es-gt	西班牙语 (危地马拉)
en	英语	es-cr	西班牙语 (哥斯达黎加)
en-us	英语 (美国)	es-pa	西班牙语 (巴拿马)
en-gb	英语 (英国)	es-do	西班牙语 (多米尼加共和国)
en-au	英语 (澳大利亚)	es-ve	西班牙语 (委内瑞拉)
en-ca	英语 (加拿大)	es-co	西班牙语 (哥伦比亚)
en-nz	英语 (新西兰)	es-pe	西班牙语 (秘鲁)
en-ie	英语 (爱尔兰)	es-ar	西班牙语 (阿根廷)
en-za	英语 (南非)	es-ec	西班牙语 (厄瓜多尔)
en-jm	英语 (牙买加)	es-cl	西班牙语 (智利)
en	英语 (加勒比海)	es-uy	西班牙语 (乌拉圭)
en-bz	英语 (伯利兹)	es-bo	西班牙语 (玻利维亚)
en-tt	英语 (特立尼达)	es-sv	西班牙语 (萨尔瓦多)
et	爱沙尼亚语	es-hn	西班牙语 (洪都拉斯)
fo	法罗语	es-ni	西班牙语 (尼加拉瓜)
fa	波斯语	es-pr	西班牙语 (波多黎各)
fi	芬兰语	sx	Sutu
fr	法语 (标准)	sv	瑞典语
fr-be	法语 (比利时)	sv-fi	瑞典语 (芬兰)

代码	Windows 区域设置	代码	Windows 区域设置
fr-ca	法语 (加拿大)	th	泰语
fr-ch	法语 (瑞士)	ts	Tsonga
fr-lu	法语 (卢森堡)	tn	茨瓦纳语
gd	盖尔语 (苏格兰)	tr	土耳其语
gd-ie	盖尔语 (爱尔兰)	uk	乌克兰语
de	德语 (标准)	ur	乌都语
de-ch	德语 (瑞士)	vg	Valley Girl
de-at	德语 (奥地利)	ve	文达语
de-lu	德语 (卢森堡)	vi	越南语
de-li	德语 (列支敦士登)	xh	班图语
el	希腊语	ji	意第绪语
he	希伯来语	zu	祖鲁语
hi	印度语		

使用 CSV_Include 工程

本部分包含有关 CSV_Include 工程的信息，并介绍以下内容：

[CSV Include 简介](#)

[使用页面和模板](#)

[创建新工程](#)

[Vijeo Citect 技术参考中的“CSV Include 参考”](#)

Chapter: 38 CSV_Include 简介

CSV_Include 工程是一个预先配置的工程，包含用于 Windows XP 环境的一组模板和页面风格。它有助于减少用来配置新工程的时间。

当创建新 Vijeo Citect 工程时，CSV_Include 工程会作为包含工程自动合并进来。这意味着，在图形编辑器中创建图形画面时，所有工程模板及相关内容均可用于实施。

在包括用于创建现场摹拟的标准图形画面的同时，该工程还包括预定义的趋势和报警显示画面、管理工具画面、用于显示文本和 RTF 文件的文件画面以及弹出窗口选择。所有画面均具有相同导航和报警菜单，使得整个工程的“外观”都一致。该工程甚至支持多监视器显示，从而可以在多个计算机屏幕上同时显示几个图形画面。

注意：请不要修改 CSV_Include 工程作为运行时工程使用。它将无法成功编译，应当将它保留，以作为新工程的模板使用。Vijeo Citect 升级会安装 CSV_Include 工程的新版本，因此升级时将丢失对工程所做的更改。

在何处查找信息

使用下面的链接获取有关 CSV_Include 工程中预配置的内容的详细信息：

- 预定义的画面和模板。请参见 [使用画面和模板](#)。
- 通用工具栏。请参见 [通用工具栏](#)。

它还介绍了用于创建基于 CSV_Include 的工程的过程：

- 创建新工程。请参见 [创建新工程](#)。
- 创建画面。请参见 [创建画面](#)。
- 创建自定义菜单。请参见 [创建自定义菜单](#)。
- 创建报警组。请参见 [创建报警组](#)。
- 创建趋势组。请参见 [创建趋势组](#)。

还包括参考材料，介绍了可用于定制工程的 citect.ini 参数和 Cicode 函数。

Chapter: 39 使用页面和模板

基于 CSV_Include 创建新工程时，以下模板和页面可在创建图形页面时使用。本部分介绍页面及其按钮、菜单和工具。

CSV_Include 工程包含以下模板：

模板名称	描述
正常	标准图形页面
报警	活动报警页面
禁止	禁止的报警页面
汇总	汇总报警页面
硬件	硬件报警页面
趋势	趋势页面（8 支笔）
双重趋势	分隔页面趋势页面（16 支笔）
文件	显示文件
管理工具	工程工具页面
Poptrend	弹出式趋势
Instanttrend	即时趋势
Popup_Small	弹出式小窗口
Popup_Mid	弹出式中窗口
Popup_Large	弹出式大窗口
Popup_Xlarge	弹出式超大窗口

工程还包括以下预先配置的页面：

页面名称	描述
CSV_AdminTools	工程工具页面

页面名称	描述
CSV_Alarm	活动报警页面
CSV_AlarmDisabled	禁止的报警页面
CSV_AlarmSummary	汇总报警页面
CSV_AlarmHardware	硬件报警页面
CSV_Trend	趋势页面（8 支笔）
CSV_TrendDouble	分隔页面趋势页面（16 支笔）
CSV_File	文件页面

另请参见
[创建新工程](#)
[创建页面](#)

普通画面模板

此模板设计用于创建用户所需内容和出厂模拟。此画面包含的标准漫游工具条和报警工具条是所有 CSV_Include 模板的重要特征。如果您要创建的工程以 CSV_Include 为基础，可以将此模板用于您的标准图形画面。

使用普通模板创建的画面显示格式为 1024 x 768，无标题栏，这是 CSV_Include 工程中所有画面的缺省大小。

另请参见
[创建画面](#)

报警画面模板

包括以下类型的报警显示模板：

- **活动报警画面（报警）**：用于创建一个画面，其中显示未确认、已确认以及仍处于报警状态下的所有听得见的报警。预先配置的画面 CSV_Alarm 基于此模板。
- **硬件报警画面（硬件）**：用于创建一个画面，其中显示未确认、已确认以及仍处于报警状态下的所有系统错误的详细信息。例如，如果出现通讯丢失，如果 Cicode 无法执行，如果图形画面没有正确更新，或者如果服务器出现故障，则此画面将提醒您注意该问题。预先配置的画面 CSV_AlarmHardware 基于此模板。
- **禁用的报警画面（已禁用）**：用于创建一个画面，以显示当前在系统中禁用的所有报警。预先配置的画面 CSV_AlarmDisabled 基于此模板。由于维护关闭、损害条带而从正常的报警系统禁用这些报警时，这些报警将出现此画面上。
- **报警汇总画面（汇总）**：用于创建一个画面，其中显示已出现的报警的历史记录日志。预先配置的画面 CSV_AlarmSummary 基于此模板。此画面可用于进行故障排除。

通用功能

所有报警画面模板共享报警列表左侧的面板，该面板支持以下功能：

确认任务

提供用于执行以下任务的选项：

- 确认当前画面上的所有报警
- 仅确认选取的报警
- 为声音报警静音。

这是“活动报警”画面和“硬件报警”画面上的功能。

报警画面任务

允许用户每次浏览一个画面上的报警列表。如果有多个画面可供查看，则用蓝色框表示。

报警列表过滤器任务

允许用户根据工厂区域或按报警分类对当前的列表进行过滤。要应用特定过滤器，首先应配置报警组。

趋势画面模板

CSV_Include 工程包括以下四种类型的趋势显示：

- **单趋势：** 八只趋势笔趋势显示。预先配置的画面 CSV_Trend 基于此模板。
- **双趋势：** 分隔屏幕中显示 2 x 8 只趋势笔趋势显示。预先配置的画面 CSV_TrendDouble 基于此模板。
- **弹出式趋势：** 四只趋势笔的弹出式显示，可通过其他图表画面启动。
- **即时趋势笔：** 弹出窗口即时显示变量标签。

注意： 要在您的工程中实现即时趋势功能，必须将 CSV_InstantTrend 工程添加为包含工程。（请参见 [在当前工程中包含另一个工程](#)。）

使用单趋势、双趋势和弹出式趋势显示，您可以在一个颜色编码的线性图表中选择并显示 Vijeo Citect 工程的趋势标签。以来自 Vijeo Citect 趋势服务器的信息为基础，这些显示可以由能够重新调用的（如果需要）存储历史数据支持。

即时趋势显示具有唯一性，因为它允许选择显示多达四种变量标签。这使您能够直观地监视某个标签，而无需在 Vijeo Citect 工程中将其设置为趋势标签。您甚至可以直接将标签加载到显示中，方法是：将鼠标放置在图表画面的标签值上并键入加号（请参见参数 [TrendX]KeySeq）。

但是这一功能也有其限制，只有启动显示时数据才可用，一旦关闭显示，数据将丢失。即时趋势功能的持续时间设置限制了显示保持打开状态的时间长度，您可以通过参数 [TrendX]Duration 来调节默认的设置。

实现即时趋势的详细信息，请参见 [设置即时趋势](#)。

通用功能

趋势显示模板共享支持以下功能的通用控件：

- [选取的趋势域](#)
- [范围/刻度标记](#)
- [跨度标记](#)
- [设置跨度按钮](#)
- [趋势游标](#)
- [历史模式](#)
- [缩放](#)
- [自动缩放](#)
- [刻度默认值](#)
- [导出到文件](#)
- [粘贴到剪贴板](#)
- [绘制趋势](#)
- [趋势组](#)

选取的趋势域

显示关于目前在趋势表格中正在被变换的趋势标签的信息，包括名称，当前值和刻度范围。

Trend	Current Value	Scale Range
➔ CITECT CPU USAGE	42.0 %	0.0 - 100.0%

为了匹配图形显示，域以颜色编码，因此所列域有了颜色特性。

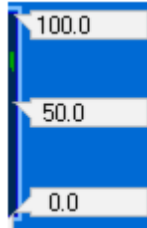
若给选取的趋势域加载某个趋势，请右键单击此域。出现一个菜单，允许您选择趋势或清除此域。一旦加载了所有选择的趋势，使用者可以通过点击此域来制作特殊的笔（图形的焦点）。域左面的箭头表明它是当前焦点趋势。

注意使用瞬时趋势时，此域显示稍微不同，因为图形会显示变量标签的当前数据，而不是趋势标签的当前数据。在这种情况下，所选域显示如下，带有标签名称，当前值，描述和显示过的采样周期。

Variable Tag	Current Value	Description	Sample Period
➔ CPU_Usage	21 %	Percentage CPU usage	1

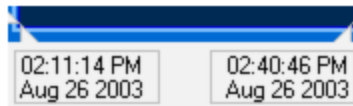
范围/刻度标记

这些标记沿垂直轴显示，并显示当前所选趋势笔的显示刻度，高亮显示刻度范围内的最低参数值到最高参数值。



跨度标记

这些标记显示在水平轴旁边，用于及时显示趋势显示的跨度。左侧标记显示当前的开始时间和日期，右侧标记显示当前的结束时间和日期。右侧标记通常显示当前的时间；但是，这可能会受到水平模式下的缩放或显示操作的影响。



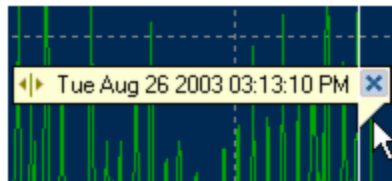
设置跨度按钮

显示一个设置当前屏幕跨度的输入对话框。当输入某个时间单位后，显示会覆盖定义的以当前时间结束的周期。



趋势光标

您可以选择一个图形趋势显示的位置，并确定该特定点的时间和日期。使用显示左端的箭头可以向左或向右移动光标。右侧的“退出”图标可以关闭趋势光标。

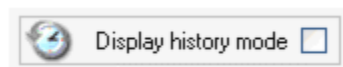


历史模式

允许您向前或向后滚动浏览趋势标签的历史图。要切换到历史模式，选中**显示历史模式**复选框。显示改为较低版本的控件（显示在左侧）。

可使用四个按钮在趋势图上向前或向后移动，使用两个外部按钮一次切换一页，使用两个内部按钮一次切换半页。可通过时钟图标编辑历史显示的结束时间。

取消选中此框将返回正常趋势视图。





缩放

当前显示的放大或缩小。放大 (+) 增加显示的焦距，连续点击持续放大。缩小 (-) 返回默认大小。



自动缩放

自动缩放当前视图，这意味着比例调整到所能达到的最低值和最高值。



刻度缺省值

返回要为当前选择的趋势缺省显示的刻度。



导出到文件

将当前显示的趋势数据导出到文件。将出现“另存为”对话框，允许您将数据另存为 D Base III 文件 (.DBF 文件)、逗号分隔值文件 (.CSV 文件) 或文本文件 (.TXT 文件)。



粘贴到剪贴板

将当前所显示趋势的数据发送到 Windows 剪贴板。



绘制趋势

将趋势绘制到打印机。使用此按钮可打印趋势图而不是“打印画面”按钮。可通过对话框配置打印机设置。



趋势组

您可以加载趋势组。对话框列出了当前配置的组，并允许您清除当前显示的变量标签。请参见 [创建趋势组](#)。



文件画面模板

使用此模板可以创建能够显示文本 (.txt) 或 RTF (.rtf) 文件的画面。

要理解文件画面如何工作，必须先区分文件画面和该画面上出现的文件；因为文件画面仅仅是多种文件的一个空模板。

一旦调用 `CSV_Nav_File` 功能，即会显示文件画面。当执行时，`CSV_Nav_File` 决定要显示在该文件画面上的文件、其位置、用于该画面的标题，以及该文件是否可以编辑。

例如，可以配置一个调用以下项的菜单项：

```
?CSV_Nav_File(MyPageTitle,[Run]:\file.txt,2)
```

它会调用文件画面，在标题栏上放上标题“`MyPageTitle`”，从运行目录载入名为 `File.txt` 的文件，并允许编辑该文件（将最后一个参数设为 2，文件即可保存）。

参数 `[Navigation]FilePage` 决定了此过程中用作模板的画面。预先配置的画面 `CSV_File` 为默认项。可以更改 `Navigation[FilePage]` 的设置，但是，必须要确保为此参数指定的任何画面都基于 `CSV_File` 模板，否则，`CSV_Nav_File` 将无法正常运行。

管理工具页面模板

此模板用于创建一个页面，该页面提供网络和本地计算机统计信息，并允许用户启动 Citect 配置工具和 Windows 应用程序。预先配置的页面 `CSV_AdminTools` 基于此模板。

此页面左侧的面板用于启动以下配置工具：

- **Windows 应用程序：** 允许用户从运行时环境中启动列出的 Windows 应用程序。
- **Citect 配置：** 允许用户从运行时环境中启动列出的 Citect 配置工具。这包括查看和编辑 `Citect.INI` 文件，运行“计算机设置向导”，创建报警组、趋势组，以及启动“菜单配置”工具。
- **Citect Kernel：** 启动 Citect Kernel 的各种组件，允许从运行时环境中调试系统。
- **系统/硬件：** “I/O 设备统计信息”可启动一个对话框，其中显示 I/O 设备统计信息。借助“下一步”和“上一步”按钮，可以逐步将 I/O 设备连接至系统，而“搜索”按钮允许您查看特定设备的统计信息。

借助“标签调试”工具，可以浏览可用标签的列表并读取选定标签的当前值。根据访问权限，您还可以向标签中写入新值。

此页面的右侧显示有关 Vijeo Citect 系统的统计信息：

- **系统信息：** 提供当前运行时计算机的 CPU 使用情况、内存使用情况和可用磁盘空间的详细信息。

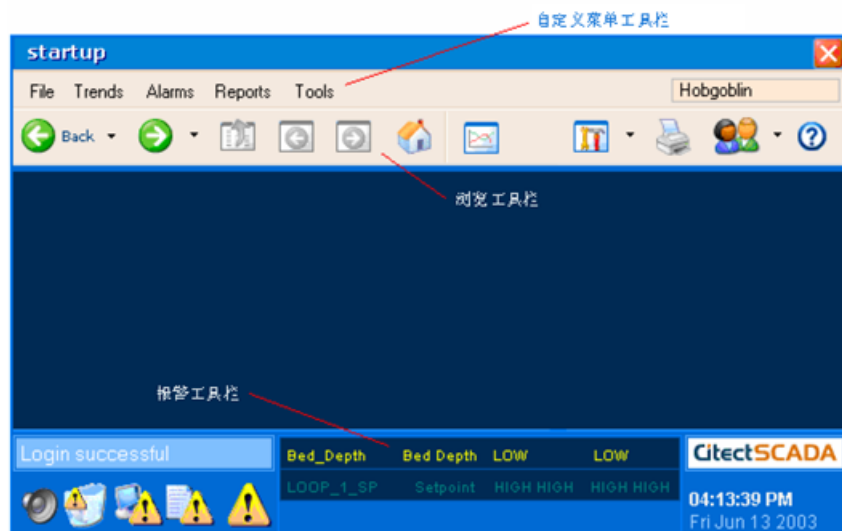
- **Citect 信息:** 提供有关 Vijeo Citect 系统的信息，包括版本信息以及当前连接的趋势、报警和报表客户机的数目
- **I/O 服务器:** 提供有关系统连接至的 Vijeo Citect I/O 服务器的状态的信息。I/O 服务器计算机的名称出现在此面板的标题栏中。显示的信息包括：最大值 - 整体最大响应时间（毫秒）
 - **平均值:** 整体平均响应时间（毫秒）。
 - **最小值:** 整体最小响应时间（毫秒）。
 - **读请求:** 每秒的读请求总数。
 - **读物理:** 每秒的物理读请求总数。
 - **写请求:** 每秒的写请求总数。
 - **写物理:** 每秒的物理写请求总数。
 - **复位按钮:** 清除当前值并重新计算这些值。

通用工具栏

为便于浏览、功能访问和保持外观一致性，CSV_Include 项目中的画面包含通用工具栏。

操作期间，以下三个树状工具栏位于屏幕上：

- **漫游工具条:** 提供浏览按钮，可由此直接访问一些重要画面（如“趋势”画面和“管理工具”画面）。
- **报警工具栏:** 可由此访问“报警”画面，并显示最近的三个活动警报。
- **自定义菜单工具栏:** 提供可以导航到特定画面或调用 Cicode 函数的菜单。这些菜单的内容是在运行时使用查找表生成的。



另请参见
[漫游工具条](#)
[报警工具栏](#)

[创建自定义菜单](#)

漫游工具条

漫游工具条包括允许用户在工程的画面之间移动的按钮。如果当前用户权限不足或未对特定按钮配置任何选项，则按钮对用户不可用。

图标	名称	描述
	后退	退回当前画面之前显示的画面。通过按钮右侧的箭头，可以从最近访问过的画面的菜单中进行选择。通过调整参数 <code>[Navigation]LastPageStackSize.</code> ，可以设置此菜单中包括的画面的最大数量。
	前进	前进到按“后退”按钮之前显示的页面。通过按钮右侧的箭头，可以从最近返回到的画面的菜单中进行选择。通过调整参数 <code>[Navigation]LastPageStackSize</code> ，可以设置此菜单中包括的画面的最大数量。
	父画面	更改当前画面的“父画面”的显示。可以通过在图形编辑器中设置环境变量来为图形画面分配父画面。为此，需打开要为其分配父画面的画面。转到画面的属性对话框（ 文件 属性 ），然后单击 环境 标签。使用父画面名称的参数值添加名为“ParentPage”的新变量。
	上一个/下一个	如果配置了次序，通过浏览次序中的画面向后或向前移动。要配置浏览次序，请转到次序中每个画面的“画面属性”对话框（ 文件 属性 ），并在 常规 标签中相应设置 下一个 和 上一个 字段。
	主页	显示“主页”画面。缺省情况下，此画面是启动画面 <code>CSV_Start</code> 。要将主页更改为其它画面，请使用 <code>[Navigation]HomePage</code> 参数。
	趋势画面	显示“趋势”画面。缺省情况下，按此按钮时会显示预先配置的画面 <code>CSV_Trend</code> 。要显示其它画面，请调整参数 <code>[Navigation]TrendPage</code> 。
	网络画面	显示名为“网络”的画面（如果存在）。缺省情况下，此画面不存在，这表示该按钮也不会显示。要使用此按钮调用函数或调用其它名称（非“网络”）的画面，请调整参数 <code>[Navigation]NetworkPage</code> 。
	工具	显示名为 <code>CSV_AdminTools</code> 的“管理工具”画面。可以调整参数 <code>[Navigation]ToolsPage</code> 来调用包含此按钮的函数或更改显示的画面，但不建议这样做。此按钮还提供启动“标签调试”工具和“即时趋势”显示的菜单。通过此按钮，还可以切换工具提示的打开与关闭。
	打印画面	打印当前画面。需要为打印选择正确配置参数 <code>[Printer]Port</code> 。

图标	名称	描述
	登录	显示标准 Vijeo Citect 登录提示。右侧菜单提供额外的选项，例如“登出”、“更改口令”、“编辑用户”（受登录限制）和“创建用户”（受登录限制）。
	帮助	此按钮可以配置为显示工程的用户帮助信息。缺省情况下，此按钮指向 Vijeo Citect 帮助中的 在画面中添加用户帮助 。此主题介绍如何通过禁止按钮或将其配置为连接到各自 HTML 帮助文件来在各自的工程中更改此缺省行为。



另请参见
[报警工具栏](#)
[创建自定义菜单](#)

报警工具栏

通过报警工具栏，可以访问当前报警信息和用于单击访问报警画面的导航按钮。此外，该工具栏还包括标准 Vijeo Citect 提示符以及当前日期和时间。



报警工具栏包含以下功能：

图标	名称	描述
n/a	最后的报警	该工具栏的中心面板显示在系统中触发的最后三个报警，为操作员提供一个了解报警出现情况的直观线索。 可以修改此列表的格式并控制出现的报警。例如，可以将该列表设置为仅显示特定类型、分类或优先级的报警。 有关详细信息，请参见 <code>LastAlarmFmt</code> 、 <code>LastAlarmCategories</code> 、 <code>LastAlarmPriorities</code> 和 <code>LastAlarmType</code> 参数。
	激活报警	将显示更改为激活报警画面， <code>CSV_Alarm</code> 。此按钮具有动画效果，当系统中存在未确认的报警时将会闪烁。通过此按钮执行的动作由 <code>[Navigation]AlarmPage</code> 参数确定。
	报警汇总	将显示更改为报警汇总页面 (<code>CSV_AlarmSummary</code>)，该画面提供报警出现的历史记录日志。通过此按钮执行的动作由 <code>[Navigation]SummaryPage</code> 参数确定。

图标	名称	描述
	硬件报警	将显示更改为硬件报警画面，CSV_AlarmHardware。这是一个有动画效果的按钮，在出现未确认的硬件报警时将会闪烁。通过此按钮执行的动作由 [Navigation]HardwarePage 参数确定。
	禁止的报警	将显示更改为禁用报警画面，CSV_AlarmDisabled。通过此按钮执行的动作由 [Navigation]DisabledPage 参数确定。
	报警静音	使报警出现时发出声音的报警蜂鸣器静音。请参见 实现听得见的报警 。

另请参见
[创建自定义菜单](#)

Chapter: 40 创建新工程

基于 CSV_Include 工程创建工程非常简单；默认情况下，它被综合为包含所有新建工程的工程。也就是说，无论何时开始新建工程，都可以根据需要使用 CSV_Include 画面和模板。

为了使对工程（基于 CSV_Include 工程）进行配置更加容易，请安装下列步骤操作：

1. 创建一个特权用户；请参见 [创建特权用户](#)。
2. 运行计算机设置向导；请参见 [运行计算机设置向导](#)。
3. 设置即时趋势；请参见 [设置即时趋势](#)。
4. 设置多监视器显示；请参见 [在多个监视器上显示工程](#)。
5. 实现声音报警；请参见 [实现声音报警](#)。

请不要修改 CSV_Include 工程作为运行期间工程使用；而应将其作为新工程的模板使用。

如果基于 CSV_Include 模板创建工程，请不要包含使用不同风格的画面，包括以前的包含工程。这样做有可能影响功能。

另请参见
[创建画面](#)

创建特权用户

某些 CSV_Include 工程内容通过用户登录加以保护。如果没有进行有效登录，一些工程功能将被禁用。例如，如果你以受限用户身份登录，工具画面大多处于未激活状态。

默认情况下，CSV_Include 工程中的以下元素受全局权限的保护。

元素	全局权限	相关函数
管理工具画面	8	[Privilege]EngTools
编辑用户	8	[Privilege>EditUser
工程关闭	0	[Privilege]Shutdown
确认报警	1	[Privilege]AckAlarms
禁用报警	8	[Privilege]DisableAlarms

配置 CSV_Include 工程时，请确保用户拥有访问可用功能的适当权限。确认用户在需要时能确认报警，并且能够访问管理工具画面的所有功能。

要为上面列出的元素调整全局权限以实现更复杂的安全体系结构，请调整 `citect.ini` 文件中的 `[Privilege]` 参数。有关详细信息，请单击上表中的相关链接。

另请参见

[运行计算机设置向导](#)

运行计算机设置向导

对于所有 Vijeo Citect 工程，必须在运行 `CSV_Include` 工程的任何计算机中完成计算机设置向导。有关详细信息，请参见 [运行计算机设置向导](#)。

如何使用向导设置计算机主要取决于用户，由于需要一些必需的设置，因此必须将计算机设置向导作为自定义设置运行。在“向导”页面中，必须注意几个问题：

事件设置画面

`CSV_Include` 工程包含三个预配置的事件。

- `CSV_TrendXClient`：启用即时趋势功能。
- `CSV_TrendXServer`：启用即时趋势功能。
- `CSV_AlarmClient`：启用声音报警器。

选择并启用所需的上列事件。要支持即时趋势功能，您必须在作为趋势服务器的计算机上启用 `CSV_TrendXClient` 和 `CSV_TrendXServer` 事件。在您的运行时计算机上只需启用 `CSV_TrendXClient`。要支持声音报警，请启用 `CSV_AlarmClient`。

您应选择所有事件，除非您需要禁用相关的功能。

安全设置—控制菜单页面

提供显示标题栏选项。当 `CSV_` 包含 XP 风格标题栏的页面，取消此选项会允许工程在全屏模式下成功地运行（全屏模式运行 `CSV_Include`-based 工程）。

另请参见

[创建特权用户](#)

[设置瞬时趋势](#)

[实现声音报警](#)

设置瞬时趋势

通过瞬时趋势，可以在不将标签配置为趋势标签的情况下，直观地从 Vijeo Citect 工程中监视变量标签。由于不支持方便使用此功能的 Vijeo Citect 趋势服务器，因此瞬时趋势要求您进行下列设置：

1. **包括 `CSV_InstantTrend` 工程。**要在自己的一个工程中实现瞬时趋势功能，必须将 `CSV_InstantTrend` 工程添加为包含工程。（参见 [在当前工程中包含另一个工程](#)。）
2. **启用 `CSV_TrendX` 事件。**正如 [运行计算机设置向导](#) 中的描述一样，您必须启用事件 `CSV_TrendXClient` 和 `CSV_TrendXServer` 才能使瞬时趋势发挥作用。确认在趋势服务

器计算机上启用两个事件，并在将使用瞬时趋势的任何运行机器上启用 CSV_TrendXClient。

3. **调用瞬时趋势显示。** 确认您配置来启动瞬时趋势显示的任何按钮或菜单项将调用函数 CSV_TrendX_Display()。您应当不设置任何参数调用此函数，因为这样将实现缺省设置。这样可装载具有适当显示设置的正确页面。
4. **设置缺省显示持续时间。** “瞬时趋势”窗口只在设定的时间期间保持打开。使用 [TrendX]Duration 参数可根据需要调整缺省时间期间。
5. **配置标签选择对话框。** 可以决定使用瞬时趋势时是否将当前标签的列表加载到“标签选择”对话框中。如果您的工程有太多的标签，没有标签显示在选择对话框中可能很有用。参见计算机设置编辑器联机帮助中对参数 [TrendX]TagListEnable 的介绍。
6. **检查用于加载标签的键序列。** 通过将鼠标光标置于图形画面的某个标签值上按加号 (+) 键，可将标签直接加载到瞬时趋势显示中。不过，请确保此键序列在用于其他应用程序功能时不会崩溃。要改变此功能的键序列，请参见参数 [TrendX]KeySeq。

瞬时趋势不支持类型为 LONG 或 REAL 的变量标签。

另请参见
[趋势页面模板](#)

在多个监视器上显示工程

CSV_Include 工程可同时在多个计算机屏幕上显示几个画面。如果您的硬件可以通过单个计算机运行多个监视器，则 CSV_Include 工程可以同时在多达六个屏幕上显示不同的画面。

要跨多个监视器成功地运行工程，请调整下列参数：

- **[MultiMonitors]Monitors.** 调整此参数以指示工程需要运行在多少台监视器上。
- **[MultiMonitors]StartupPage n .** 此参数决定每台监视器在启动时显示哪个画面。请注意，需要为每台监视器重复此参数。例如，StartupPage1 决定在启动时显示在第一台监视器上的画面，StartupPage2 决定在启动时显示在第二台监视器上的画面，依此类推。

许多类型的硬件都支持多监视器显示。CSV_Include 工程已经通过了单显卡多输出 PC 上的测试，但未曾在其他架构上进行测试。受到未经测试的硬件架构的影响，在多个屏幕上显示工程画面的能力会有所不同。

实现声音报警

CSV_Include 工程提供声音报警支持。可以将工程配置为使选定的 wav 文件无论何时触发特定优先级的报警都能发出声音。您甚至可以为不同报警优先级指定不同的声音，从而能够从报警的声音判断其紧急程度。

在您的工程中实现声音报警：

1. 确认将要触发声音警告的报警分配到某个分类并指定特定的优先级。
2. 调整参数 [Alarm]Sound n 。此参数根据报警的优先级 (n) 确定报警响起时使用的 wav 文件。例如，[Alarm]Sound1 可标识在触发优先级为 1 的报警时将响起的 .wav 文件。

3. 如果需要，调整参数 [Alarm]SoundInterval。此参数根据报警的优先级 (n) 确定选定的 wav 文件响多长时间。例如，[Alarm]SoundInterval 设置优先级为 1 的报警发出声音的时间长度。
4. 在要从其发出声音提示的任何机器上启用 CSV_AlarmClient 事件。您可以从计算机设置向导的“事件设置”画面启用此事件。参见 [运行计算机设置向导](#)。
5. 确认用户有适当的登录权限来确认报警，否则他们无法将报警置为静音。参见 [创建特权用户](#)和参数 [Privilege]AckAlarms。

另请参见
[报警画面模板](#)

创建画面

考虑工程所需的画面时，首先要确定是否可以使用 CSV_Include 工程中任何预定义的画面。

画面名称	描述
CSV_Trend	默认的 8 笔趋势画面，通过“漫游”工具条“工具栏”上的“趋势”按钮调用。
CSV_Alarm	当前报警画面，从“报警”工具栏上调用。
CSV_AlarmHardware	硬件报警画面，从“报警”工具栏上调用。
CSV_AlarmDisabled	禁用报警画面，从“报警”工具栏上调用。
CSV_AlarmSummary	报警汇总画面，从“报警”工具栏上调用。
CSV_AdminTools	管理工具画面，通过“漫游”工具条上的“工具”按钮调用。

“漫游”工具条还提供了几个可根据需要创建的画面。确定下列画面是否对您的工程有用，可以用适当的名称来创建它们：

画面名称	描述
主页	通过“漫游”工具条上的“主页”按钮调用。
网络	通过“漫游”工具条上的“网络”按钮调用。注意，如果名为“网络”的画面不存在，将不显示这个按钮。

所列画面与调用它们的按钮之间的链接，可通过 citect.ini 文件中的 [Navigation] 参数设置加以定义。如果要通过按钮调用具有其他名称的画面，请调整这些参数设置。有关详细信息，请参见 Vijeo Citect 技术参考中的“CSV Include 参数”。

创建新画面

使用最少内容设计标准和弹出模板，以呈现客户所需的信息和现场摹拟。

画面基于图形编辑器中的模板创建，从“使用模板”对话框（文件 | 新画面）中选择所需模板。用户可通过从**风格**字段中选择 **csv_xp_01** 来访问 CSV_Include 模板。

注意：如果要从 Citect 工程管理器中加载 CSV 模板，就必须定位到 [Project Name]/Graphics/Templates/csv_xp_01/XGA 下。

CSV_Include 工程中无须绘制和配置多个元素即可在画面中添加滚动按钮。通过将函数 `DspButtonRollOver()` 添加到按钮的可见属性，用户可以在画面中添加文本，鼠标经过时后面将显示一个按钮。不需要参数。

另请参见

[向页面中添加用户帮助](#)

向页面中添加用户帮助

CSV_Include 工程包括“漫游”工具条上的“帮助”按钮。创建页面时，可以配置此按钮，以向用户提供有关页面内容及其支持功能的信息。

为操作员提供的信息必须作为各个 HTML 页面创建并编译为 HTML 帮助文件 (.chm)。创建 .chm 文件所需的编译器可以从 Microsoft 开发人员网络 (MSDN) 下载。请转到 <http://msdn.microsoft.com> 并搜索 "HTML Help Workshop"。SDK 可用，其中包括有关创建 HTML 帮助工程所需的下载内容和信息。

创建 .CHM 文件后，建议将其置于工程目录中。

为工程页面配置帮助按钮

1. 选择已在工程浏览器中创建的工程。
2. 转到“工程编辑器”，并从**系统**菜单中选择**参数**。此时将出现“参数”对话框。
3. 在**部件名称**字段中，键入 "HelpButton"。
4. 在**名称**字段中，使用以下格式标识要向其中添加帮助的页面：

```
SetTopic:<PageName>
```

其中，<PageName> 是 Vijeo Citect 工程中包括的页面的名称。请不要在冒号后面加空格。

5. 在**值**字段中，使用以下格式标识关联的 HTML 帮助文件和主题：

```
<HelpFileName>|<HTMLPageName>
```

其中:

- `<HelpFileName>` 是已创建的编译好的 HTML 帮助文件的名称 (例如 `MyHelpFile.chm`)
 - `<HTMLPageName>` 是要作为帮助主题显示的编译好的 HTML 页面的名称 (例如 `Help_Topic.html`)。
6. 如果需要, 请添加**注释**。
 7. 点击**添加**。

帮助按钮将立即启动已标识的 HTML 文件, 同时显示选定的主题。

禁用工程页面的帮助按钮

1. 选择已在工程浏览器中创建的工程。
2. 转到“工程编辑器”, 并从**系统**菜单中选择**参数**。此时将出现“参数”对话框。
3. 在**部件名称**字段中, 键入 "HelpButton"。
4. 在**名称**字段中, 使用以下格式标识要禁用其帮助按钮的页面:

```
SetTopic:<PageName>
```

其中, `<PageName>` 是 Vijeo Citect 工程中包括的页面的名称。请不要在冒号后面加空格。

5. 在**参数值**字段中, 键入 DISABLED。
6. 如果需要, 请添加**注释**。
7. 点击**添加**。

现在, 已标识页面上的帮助按钮在运行时期间将不可用。

注意: 请确保不要为单个工程页面复制此参数。

另请参见

[使用页面和模板](#)

创建自定义菜单

自定义菜单工具栏, 紧挨在画面标题栏下方。可用于创建下拉菜单, 以调用 `Cicode` 函数或导航到特定画面 (亦即到特定的模拟现场)。

可通过调整参数 `[Page]MenuDisable` 来禁用所有画面上的菜单栏。如果要禁用特定画面上的菜单, 可以将其作为一个环境变量指定这个参数。要完成此步骤, 请在“图形编辑器”中打开请求的画面, 转到“属性”对话框 (**文件 | 属性**), 并在“环境”标签中插入参数。

菜单配置工具

菜单内容可通过菜单配置工具来配置，该工具从“管理工具”画面的“Citect 配置”面板中启动。

菜单配置工具有两个面板。左侧面板表示为目录结构中的当前工程配置的菜单。右侧面板显示有关左侧面板中当前选定项的信息。

左侧面板中的目录是 DBF 查找表的图形表示，该表是运行时所显示菜单的基础。分级结构由此表中的下列字段确定：

画面	该画面字段定义为“通用”或工程内某个画面的名称。通用表示菜单与所有画面相关，特定画面名称表示菜单只出现特定画面上。
菜单名	包括在指定画面上的菜单的名称。
菜单项	出现在各菜单中的菜单项。
子菜单	出现在菜单中的子菜单（可选）。添加子菜单会自动删除为分支菜单项定义的动作，因为该父项变成了孩子菜单列表的占位符。

显示的菜单名称有五个缺省值：“页面”、“趋势”、“报警”、“文件”和“工具”。

预期可以看见与某个项关联的字段（在 DBF 表中定义）如下：

动作	要显示画面的名称或 Cicode 函数。如果指定 Cicode 函数，则一定带有问号前缀 ("?")。要通过菜单系统调用具有参数的 Cicode 函数，请使用下面的格式：函数名称，空格，以逗号分隔的字符串参数列表。例如，?LoadPav dP45.pav。
权限	触发相关动作所需的登录权限。
禁止	指示该项当前是否处于禁用状态（浮雕）。例如，如果当前登录不满足为该菜单项设置的必要权限（见上），则此字段将设置为 True 且该菜单项将显示为浮雕，表示其当前未激活。
选中	指示该菜单项当前是否处于选中状态（带钩选标记），True 表示选中。
按钮宽度	指定按钮的宽度（可选）。

另请参见
[构建自定义菜单](#)

构建自定义菜单

创建自定义菜单需要将所需的画面、菜单和项添加到“配置菜单”目录结构中，然后将所需的动作和权限分配给每个项。所有这些操作均通过右键单击菜单实现。

右键单击目录中的分支可调出上下文菜单，菜单中会显示可对当前选定项采取的动作。右键单击菜单中包括以下动作：

编辑项	选定一个项时提供。选择此选项会显示“编辑项”对话框，可在该对话框中定义与当前项关联的字段。特别是，可以在此指定与该项关联的动作。（参见 编辑项 ）
新建画面	将新画面添加到菜单配置中。使用此选项可新建专门为工程中特定画面设计的自定义菜单。注意，在菜单配置中为新画面指定的名称必须与要将菜单应用到的工程中的画面的名称完全相同。
新建按钮	将新菜单按钮添加到当前画面中。
新建项	将新项添加到当前选定的菜单中。
新建子项	将子项添加到当前选定的项中。子项会覆盖为其父项配置的动作。父项会成为标识分配给它的子项列表的标签。
删除画面	从菜单配置中删除当前选定的画面。
删除按钮	从菜单配置中删除当前选定的菜单按钮。
删除项	从菜单配置中删除当前选定的项。
删除子项	从菜单配置中删除当前选定的子项。
复制画面	复制当前选定的画面并将它粘贴到菜单配置的末尾。如果要让画面包括在通用画面菜单配置中，但在显示特定画面时只显示增加的菜单，此选项非常有用。为此，请复制该通用画面，重命名它以便与要为其自定义菜单的画面相符，然后添加所需的菜单和项。
保存	保存当前菜单配置。

编辑项目

您可以使用**编辑项目菜单**对话框来配置菜单项的功能。

使用此对话框，可以识别某项将要触发的动作、将要导航到的某个特定画面或将要执行的某个函数。要启用此对话框，请右键单击要配置的项目，然后从菜单中选择**编辑项目**。

您要填充的字段如下：

动作	表示要显示的画面的名称或 Cicode 函数。如果指定 Cicode 函数，则一定带有问号前缀 ("?")。
权限	需要登录权限才可以触发某动作。只有拥有相应访问权限的用户才可以使用此项。
禁止	此设置自动由当前用户的访问权限确定，除非您希望缺省禁止该项，否则将此设置保留为 假 。
检查	除非您希望缺省检查该项，否则将此设置保留为 假 。
按钮宽度	将此设置保留为零 (0)，它可以将菜单按钮设置为适当的宽度。要设置一个具体的按钮宽度，请输入一个宽度值（以像素度量）。

另请参见
[通用工具栏](#)

创建报警组

CSV_Include 工程允许使用“报警组”来显示特定的系列标签，这些标签是由在运行期 Vijeo Citect 工程范围内配置的报警分类和区域设置定义的。

例如，用户可以创建由一个分类的所有报警定义的组。然后，就可以将该组作为一个过滤器，用于创建当前显示在“激活报警”画面上的一个分类的所有报警的列表。

在管理工具画面左边的 Citect 配置面板中点击相应链接，即可对报警组进行配置。也可以将功能添加到自定义菜单中以便访问。

配置报警组：

1. 单击**工具**按钮，转到运行期间工程的“管理工具”画面。
2. 从“Citect 配置面板”中选择**配置报警组**。随即会显示“配置报警组”对话框。
3. 在**报警组描述**框中，键入要用来标识该组的名称。
4. 在**类别**框中，列出在要用于定义组的 Vijeo Citect 运行期工程中配置的类别。
5. 在**区域**框中，键入在要用于定义组的 Vijeo Citect 工程中定义的区域。
6. 点击**添加**。

该对话框右侧的信息表示已配置的报警组个数以及用户当前在列表中的位置。可以使用上下箭头滚动已配置组的列表。

要更改组，请在列表中找到它，根据需要进行更改，然后点击**替换**。

另请参见
[创建趋势组](#)

创建趋势组

CSV_Include 工程允许使用“趋势组”来显示特定的趋势标签集。一个趋势组最多可包括 8 个标签变量，它们可以自动加载到趋势显示中，而不必逐个选择每个标签。

在管理工具画面左边的 Citect 配置面板中点击相应链接，即可对趋势组进行配置。也可以将功能添加到自定义菜单中以方便访问。

配置趋势组：

1. 单击**工具**。
2. 从“Citect 配置面板”中选择**配置趋势组**。随即会显示“配置趋势组”对话框。
3. 在**描述**文本框中，输入要用来标识该组的名称。
4. 在**趋势笔**文本框中，选择要让该组趋向的 8 个标签变量。单击每个字段右边的按钮，以查看 Vijeo Citect 运行工程中的可用标签。
5. 在**区域**文本框中，键入在要用于定义组的 Vijeo Citect 工程中定义的区域。
6. 点击**添加**。

该对话框右侧的信息表示已配置的趋势组个数以及用户当前在列表中的位置。可以使用上下箭头滚动已配置组的列表。

要更改组，先在组列表中找到它，或者使用**描述**文本框右边的选择按钮浏览该组，进行必要的更改后，点击**替换**。

另请参见
[创建报警组](#)

使用环境变量

参数将指定规则应用到所有模板页面时，环境变量也会将规则应用到**特定**页面。通过此方式，可以使用环境变量指定特定页面的功能。

例如，要显示特定页面，应将页面名称指定为环境变量。要调用特定 **Cicode** 函数，应指定函数名称（带有“?”前缀）、空格，然后是逗号分隔参数的类别，如下所示：`?WinPrint LPT1,0,0,Trend.pal`。（“?”表示变量被看作函数调用，而不是要显示的页面。）

我们来看看更详细的范例：通常，标准**打印**按钮用于打印 WYSIWYG 格式的图形页面。如果页面中显示文本/html 文件（或者可能是趋势），则可以创建各自的打印函数（例如，`PrintTrend`）并在环境变量中将其指定为 `?PrintTrend`。要减少墨粉的使用量，请通过用户定义的调色板指定 `WinPrint` 函数。在这种情况下，应将其指定为参数，而不是指定为环境变量，以便应用到所有页面。

注意：在标准 `CSV_Include` 趋势页面中，可以使用标准**打印**按钮打印屏幕，或使用**趋势绘图**按钮绘制趋势。由于两者用途不同，应让用户自行选择使用。

词汇表

1

10base2

以太网以细同轴电缆来实现连接。通常用 BNC 连接。

10base5

以太网以粗同轴电缆来实现连接。

10baseT

以太网以非屏蔽双绞线来实现连接。通常用 RJ45 连接。

B

BCD 变量 (I/O 设备)

BCD (二进制码的十进制数) 是 2 字节 (16 位) 的数据类型, 值的允许范围为 0 到 9,999。2 个字节被分为 4 个 4 位, 每个 4 位代表一个十进制数。例如, 二进制数 0010 代表十进制数 2。因此, BCD 0010 0010 0010 0010 代表 2,222。

C

Cicode

用于工厂监视和控制应用的编程语言。类似于 Pascal 语言。

Cicode 阻塞函数

有一些 Cicode 函数在返回前停滞, 或等待一个异步的事件的结束。

CiNet

CiNet 现在不被支持。CiNet 是为低速广域网而设计的（远程监控应用）。如果您的工程分布很广，各个计算机之间相距遥远，使用局域网来连接控制客户机代价会很高。在这种情况下连接控制客户机，您应该使用微软的远程接入服务器 (RAS) 或微软认证的解决方案 – 如 Shiva LanRover。

citect.ini 文件

一个文本文件，存储关于每台计算机（服务器和控制客户机）如何在配置和运行时环境中运行的信息。Citect.ini 文件存储有每台计算机特定的参数，因此不能作为工程的一部分来配置。

D

DB-15

因为外形类似 D 形状，所以经常被称为“D”型连接器。内有两排共 15 针，一排 8 针，另一排 7 针。虽然不如 DB-9 或 DB-25 那么常见，在某些计算机和数据通讯设备上仍可能找到它们。通常是一边公头(针突出的一头)，另一边是母头(针孔一头)。

DB-25

因为外形类似 D 形状，所以经常被称为“D”型连接器。内有两排共 25 针，一排 13 针，另一排 12 针。此种类型的连接是 RS-232-D 标准的一部分，并且在许多计算机、调制解调器、和其它数据通讯设备上都可以找到它们。通常是一边公头(针突出的一头)，另一边是母头(针孔一头)。

DB-9

因为外形类似 D 形状，所以经常被称为“D”型连接器。内有两排共 9 针，一排 5 针，另一排 4 针。此种类型的连接很普通并经常被用在计算机串行 (com) 口上。经常用在调制解调器、和其它数据通讯设备上。通常是一边公头(针突出的一头)，另一边是母头(针孔一头)。

Digiboard

一个由 Digiboard Corporation 生产的高速串行通讯板卡。

I/O 服务器

专用的通讯服务器，用来在 I/O 设备和控制客户机之间交换数据。I/O 服务器不进行数据处理（除非为了本地显示）。数据被采集并传递给控制客户机以用来显示，或其它服务器以作进一步处理。从任意一台计算机发送给一个 I/O 设备的数据也通过 I/O 服务器。如果数据传送负荷过大时，你可以用几台 I/O 服务器来平衡负荷。

I/O 设备

与现场控制或监视设备（传感器、控制器等）通讯的设备名称。最常见的 I/O 设备是 PLC（可编程逻辑控制器）；不过，它支持很多 I/O 设备，包括回路控制器、条码识别器、科学分析仪、远程终端设备 (RTU) 和分布式控制系统 (DCS)。可以与具有标准通信通道或数据高速通道的任何 I/O 设备进行通信。

I/O 设备变量

中使用的信息单元。变量存储在 I/O 设备的内存寄存器中。通过读取和写入变量与 I/O 设备交换信息。依靠其寄存器地址指向 I/O 设备变量。I/O 设备通常支持几种类型的变量，尽管如此，最普通的还是数字量变量和整型变量。

I/O 设备地址

系统中 I/O 设备的（逻辑）位置。在系统中，每个 I/O 设备都必须具有唯一的地址，除非该 I/O 设备是在其他服务器上定义。如果配置了冗余 I/O 设备，则互为冗余的 I/O 设备在每一个 I/O 服务器中必须拥有相同的 I/O 设备命名、编号和地址。

Internet 显示客户机

允许您从远程位置通过互联网运行工程。它基本上是“仅限于运行期”版本的：您可以从该计算机运行工程，就像从任何正常客户机运行一样。

IP 地址

一个在互联网协议 (IP) 上使用的独一无二的逻辑地址。它包含有网络和主机的 ID。该格式被称作句点十进制符号，写为下列格式：w.x.y.z。

P

PLC 接口板

有时可以在服务器中安装 PLC 接口板。一个专用的接口板通常由你的 PLC 厂家提供，然后你就可以将其连接到 PLC 或 PLC 网络上。你只能将专用接口板用在相同品牌的 PLC 上。

PSTN

公共交换电话网是包含世界各地公共交换电话网的网络。现在以数字交换为主，同时还包括移动和固定电话。

R

RJ11

一种通常用于数据通讯的 IDC 物理接口。又是在电话线和电话听筒连接器里使用的数据物理接口。RJ11 是一个 6/4 物理接口，有 6 个接线点但只有 4 个被使用。

RJ12

一种通常用于数据通讯的 IDC 物理接口。又是在电话线和电话听筒连接器里使用的数据物理接口。RJ12 是一个 6/6 物理接口，而且有 6 个接线点。

RJ45

一种通常用于数据通讯的 IDC 物理接口。又是在电话线和电话听筒连接器里使用的数据物理接口。RJ45 通常与 10baseT 一起使用，并且是一个 6/8 物理接口，而且有 8 个接线点。

RS-232

一种串行通讯的工业标准。此标准说明了用于控制设备之间串行传输数据的接线和信号特征的标准。

RS-422

一种串行通讯的工业标准。此标准说明了用于控制设备之间串行传输数据的接线和信号特征的标准。RS-422 使用平衡电压接口电路。

RS-485

一种串行通讯的工业标准。此标准说明了用于控制设备之间串行传输数据的接线和信号特征的标准。RS-422 使用多节点系统里的平衡电压接口电路。

U

UAC

用户帐户控制。引入 Windows Vista 的安全技术，能够让用户更容易地按标准用户权利运行。 .

一

一般协议

磁盘 I/O 设备支持的伪协议，可以为表示磁盘数据提供一种合适的方法。一般协议不是真实协议（与非物理 I/O 设备通讯）。

中

中断

一个外部的事件表示 CPU 应该暂停它当前的任务而处理另外一个分配的活动。

串

串行通讯

串行通讯使用你的计算机上的通讯端口或一个或多个安装在你的计算机上的高速串行板卡。

主

主报表服务器

通常处理报表的服务器。

主报警服务器

通常处理报警的服务器。

主趋势服务器

通常处理趋势的服务器。

事

事件趋势/SPC

为了构建事件趋势/SPC，会在工厂中的特定事件被触发时进行采样。这个采样被显示在窗口内。在下一次采样前，此事件必须被还原并重新触发。事件是以事件号来识别的。

仅

仅查看客户端

配置为只允许经理访问运行时系统的计算机。不允许控制系统，但是允许完全访问监视数据。

代

代理服务器

缓存通过存储请求和接收信息来为当一个相同的请求来临时的重复使用服务，从而减少了平均交易时间以提高性能。当一个互连网显示客户机 (IDC) 连接到一个代理服务器时，代理服务器提供为访问报表服务器会议层信息而必须的 TCP/IP 地址。

以

以太网

广泛应用的局域网类型，以 CSMA/CD 总线接触 (IEEE 802.3) 为基础。

任

任务

包括 I/O 处理，报警处理，显示管理和 Cicode 执行等操作。任何一个 Cicode 的进程也是一个“任务”。

低

低限和低低限报警

通过指定触发每个报警的变量值定义。因为低限报警一定在低低限报警前发生，所以低低限报警触发后低限报警就不存在了。注意：变量值必须回升到死区以上报警才会消除。

保

保留词

不能被用作任何数据库记录或 Cicode 函数名称的词。

偏

偏差报警

偏差报警是由变量数值偏离设定值达到一定量而触发的。报警一直保持激活状态直到变量值下降（或回升）到死区值内。

偏斜度

是一个频率分布的不对称度的指标（通常与正态分布有关）。当分布向左偏斜时（例如），则尾部向该边延伸，图形左边的数据比正态分布要多。正的偏斜度标志着分布的平均值（和总数）是偏向右边。负的偏斜度标志着分布的平均值（和总数）是偏向左边。

停

停止位

在异步传输时标识字符结束的位的个数。该数字通常为 1 或 2。在异步传输时需要停止位是因为传输字符间不规则的时间间隔使得服务器或者 I/O 设备无法确定下一个字符何时应该到达。

全

全双工

双边（两个方向）同时独立地传输（4 线制）。

全局 Cicode 变量

全局 Cicode 变量可以被系统中的全部 Cicode 文件所共享（以及包含工程）。

全局客户机

用来监视来自几个系统或部门（使用群集）的信息的控制客户机。

冗

冗余

在系统中使用硬件的一种方法，如果系统中有一个组件出现故障的话，仍然能够保持对系统的控制并且不丢失数据。

分

分布式处理

在非常大的应用里面，因为数据量很大，你可以分散数据处理以减少单个计算机的负荷。

分布式服务器

如果您的工厂包含几个部门或系统，您可以为每个部门指定一个群集，然后使用一个控制客户机监视所有部门。注意：分散的服务器不应用来将一个单个部门或过程分成离散的区域。在这种情况下，更适用的是采用单个具有分布式处理机制（**distributed processing**）的集群系统。这是因为这样的系统不会象分布式服务器系统那样受到昂贵的维护开销的限制（例如额外的工程编译，等等）。

动

动态数据交换 (DDE)

Microsoft Windows 标准的信息协议集和指导方针，它们使同一台 Windows 计算机上的 Windows 应用程序间能够通讯。

动态数据交换 (DDE) 服务器

支持的 Windows 标准通信协议。I/O 服务器与 DDE 服务器用 Windows 标准 DDE 协议进行通讯。当数据通讯不是很关键的时候，DDE 服务器就很合适。(DDE 服务器不是为高速数据传送而设计的)。

动点

对象显示的图形画面上的点。当您在页面上添加对象时，会自动为动点（即对象的位置）分配一个号 (AN)。

动点文件 (.ANT)

动点文件是包含有动点 (ANs) 列表和每点坐标位置（按像素计）的 ASCII 码文本文件。

包

包含文件 (.CI)

可以在命令或表达式字段中键入的字符的最大数目（通常为 128 个）。如果需要在属性字段包含许多命令（或表达式），可以定义一个单独的包含命令或表达式的包含文件。包含文件是独立的单个 ASCII 文本文件，它只包含一个命令或表达式序列，该序列以其他方式键入到内的命令或表达式字段中会显得太长或过于复杂。输入包含文件的名称，调用时将激活整个文件。

区

区域

一个大型的工程可以被看成由一系列离散的部分或区域所组成。您可以从地理位置角度（当工厂被遥远距离分开时）或从逻辑角度（作为不连续的过程或单个任务）定义这些区域。

半

半双工

单向传送，而不是同时。

协

协议

消息传输格式，由用于服务器与 I/O 设备之间通信的一系列消息和准则组成。通信协议决定与 I/O 设备之间如何通信；要交换数据的类型；控制通信开始和终止的规则；以及错误检测。

单

单向传输

数据只在单向传输。

参

参数

运行时（如操作员输入）通过键盘输入至键盘命令而传递进来的数值(或变量)。参数也可以是当 Cicode 函数执行时传递给它的值（或变量）。

双

双工

在同一条通讯线上发送和接受数据的能力。

变

变化率报警

当变量值的变化快于指定的速率时被触发。报警保持激活状态直到其变化速率回落到设定值以下。死区不适用于变化速率报警。

变量类型 (Cicode)

变量的类型 - INT（32 位）， REAL（32 位）， STRING（256 字节）， OBJECT（32 位）。

句

句点符号

用于互联网地址。句点符号包含四个字段（称为八位字节），每个包含一个由句号 (.) 分隔的取值范围在 0 到 255 之间的十进制数。

可

可伸缩架构

一种不需要调整现有系统硬件或软件就可以重新设置大小的系统架构。随着更多 计算机增加进来以及分散处理负荷的需要，允许您重新分配任务。

向

向导

通过提供由一系列简单步骤组成的过程来简化复杂过程的 工具。

周

周期趋势

按一定的周期进行采样的趋势。你也可以定义一个触发条件（一个事件）来停止和启动趋势（当工厂内一个特定条件发生时）。

命

命令

命令在运行时系统中执行一个特定的任务或一系列任务。命令在 Cicode 中创建，可以只包含一个函数或语句。

回

回拨调制解调器

仅从远程 I/O 设备返回拨号。

回调函数

一个作为另一个函数的参数来传递的函数。回调函数必须是一个用户自定义的函数。

图

图形画面

在工作站上显示的图形（或映像），以便为操作员提供对工厂的控制，并可视化显示工厂内部的情况。

图形边界框

一个变灰的(灰色)点划线的用来定义图形对象外部边界的长方形边框。仅在图形对象被选取和被改变大小时才看得出。在其每个角处有大小变化控制器，而且(如果对象大小显示得足够大时)每一条边的中间也有一个。

域

域名服务器 (DNS)

一个将 URL 名翻译成 IP 地址的数据库服务器。

备

备用报表服务器

在主报表服务器不可用时处理报表的服务器。

备用报警服务器

在主报警服务器不可用时处理报警的服务器。

备用趋势服务器

在主趋势服务器不可用时处理趋势的服务器。

多

多-数字量报警

使用来自三个数字变量的值定义八个状态。对于每个状态，指定一个描述（例如，正常或停止），以及该状态是否触发报警。

奇

奇偶校验

一个通讯上的错误检查机制。每组无误传送位的 1 的个数必须保持一致（不是偶数就是奇数）。

字

字节变量 (I/O 设备)

BYTE 是一个单一字节数据类型，允许的数值范围从 0 到 255。一个字节包含 8 位。每个 ASCII 码字符通常由一个字节来表示。

实

实型变量 (Cicode)

实型 (浮点) 是一个 4 字节 (32 位) 数据类型，允许的数值范围从 3.4E38 到 3.4E38。可用一个实型变量来存储含有小数位的数。

实型变量 (I/O 设备)

实型 (浮点) 是一个 4 字节 (32 位) 数据类型，允许的数值范围从 3.4E38 到 3.4E38。可用一个实型变量来存储含有小数位的数。

客

客户机

一台通过由另外一台被叫作服务器的计算机提供的网络共享资源来获取信息的计算机。 的客户机-服务器架构用于分布式处理任务以优化性能。

对

对象

图形画面的基本构建块。 大多数对象具有在用户可自定义的条件下可以动态变化的属性，这使它们能够为工厂里的情况提供动画显示。

对象 ID (OID)

工程中与每个标签关联的对象 ID，唯一标识供基于标签的驱动程序使用的标签，编译时自动生成。它是代替实际的寄存器地址（是其它大多数驱动程序用来与 I/O 设备读写数据时用到的）来用的。

对象变量 (Cicode)

ActiveX 控件只能在局部、模块或全局范围内声明。

局

局域网 (LAN)

一个连接计算机的系统，允许这些计算机共享信息和硬件资源。有了实时 LAN 通讯，你可以在计算机间传送数据、消息、命令、状态信息、和文件。

局部 Cicode 变量

只能够在声明它的函数中被识别，并且只能在这个函数中使用。局部变量在使用之前必须先声明。任何在函数中（在函数名后）定义的变量都是局部变量，因此无须前缀。局部变量在函数退出时被销毁，其优先级高于全局和模块变量。

属

属性, 对象

描述对象的外观（大小、位置和颜色等）和函数（对象执行的命令或表达式，获取对象访问权限所需的优先级等）。

峰

峰度

是一个频率分布的峰度的指标(通常与正态分布有关)。Kurtosis < 3 - 表示窄的分布和相对高的峰度。Kurtosis > 3 - 表示宽的分布和相对平的峰度。

工

工程

监视和控制系统中的元素，例如图像画面、对象等等。这些元素存储在不同类型的文件中，例如，图像文件存储图形画面，数据库存储配置记录，等等。您使用 Citect 编译器将工程编译成运行时系统。

带

带时间标签的报警

由数字变量的状态变化触发的报警。带时间标签的报警在 I/O 设备中有相应的寄存器，以记录报警变为活动时的精确时间。您可以在您需要知道报警发生的确切时间的情况下使用含时间标签的报警。

带时间标签的数字量报警

带时间标签的数字量报警与数字量报警的工作方式相同，除了带有时间标签（报警开和报警关时间），使用现场设备（即 RTU 或 PLC）提供的毫秒级精度的时间。带有时间标签的数字报警的配置和数字报警配置完全相同。

带有时间戳的模拟量报警

带时间标签的模拟量报警与模拟量报警的工作方式相同，除了带有时间标签（报警开和报警关时间），使用现场设备（即 RTU 或 PLC）提供的毫秒级精度的时间。带有时间标签的模拟报警的配置和模拟报警配置完全相同。

开

开放数据库连接 (ODBC)

允许应用程序使用结构化查询语言 (SQL) 访问数据库管理系统中的数据。

当

当前报警

当前报警是指以下几种状态下的报警：ON 并且没被确认过；ON 并且被确认过；OFF 并且没被确认过。

打

打包

打包整理数据库重新对数据库中的记录进行索引，并删除标有删除标志的记录。如果将数据库从外部编辑到，应该随后打包整理该数据库。

报

报表

工厂现场条件的声明或报告。报表可以周期性的或由事件发生触发来获取。

报表服务器

控制报表处理。你可以在任何时候或特定事件发生是生成报表。

报表格式文件

控制报表的布局和内容。格式文件可用文本编辑器来编辑，而且可以是 ASCII 或 RTF 格式。

报警分类

可以为每个报警指定一个分类，然后把每个分类作为一个组来对待。例如，对于每个分类，可以指定显示特征、触发该分类中的报警时要采取的动作，以及记录关于该报警的多少数据。还可以为每个报警分类指定一个优先级，用来为报警显示、过滤确认等排序。

报警显示画面

报警显示画面以下列格式显示报警信息：报警时间，标签名称，报警名称，报警描述。

报警服务器

报警服务器不断地监视所有的报警。一旦某个报警条件满足，报警服务器将立即在相应的控制客户端上显示这个报警。

报警汇总页面

以下列格式显示报警汇总信息：报警名称，开始时间，结束时间，增量时间，注释。

拨

拨入调制解调器

仅接收来自远程 I/O 设备的拨号，识别呼叫者，然后立即挂断，因此可以接收其它拨号。随后使用回拨调制解调器返回拨号。

拨出调制解调器

对响应请求（例如，调度请求、基于事件的请求或操作员请求）的远程 I/O 设备进行拨号。远程 I/O 设备也会返回拨号。

持

持久缓存

保存到计算机硬盘上的缓存数据，它允许关闭并重启 I/O 服务器时不必重新拨号到每个 I/O 设备以获取当前值。此缓存由所有的 I/O 设备的标签值组成。

按

按时间显示的事件数据

作为按事件号查看事件趋势数据的替代，它可以通过时间线查看数据趋势。当事件趋势按时间显示时，趋势图形包含开始和结束时间，并允许操作员查看触发事件的时间和事件间的流逝周期。此数据还可以在同一个图形中显示为周期趋势。

控

控制客户机

运行时系统和操作员之间的接口。如果是在网络上使用，则所有计算机（在网络上）都是控制客户机。

操

操作员提示

当操作员将鼠标停留一个对象上面时，一条在弹出窗口里显示的帮助信息。

数

数字量变量 (I/O 设备)

通常与 I/O 设备中的离散 I/O 相关联，数字量变量只能以下列两个状态之一的形式存在：**on (1)** 或 **off (0)**。因此数字量数据类型的取值范围是 0 或 1。离散输入（例如限位开关、光电池和紧急停止按钮）和离散输出被存储为数字量变量。

数字量报警

由数字变量的状态变化触发。当过程量仅有一个或两个状态时使用数字量报警。（数字量变量的）**ON (1)** 状态或 **OFF (0)** 状态都可以用来触发报警。

数据传送

信息从一个地方传送到另外一个地方。数据传送的速度是按位/每秒 (**bps**) 来计算的。

数据位

一组用来代表在异步传输里的一个单个字符的数据的二进制数据位(位)。

数据类型 (I/O 设备)

I/O 设备变量的类型。I/O 设备支持用于与 交换数据的多种数据类型。只要在 系统中定义或引用 I/O 设备变量，就必须指定正确的 数据类型。

数据终端设备 (DTE)

完成数据源、数据槽或两者兼有的功能的设备。

数据通讯设备 (DCE)

建立、维护并终止数据传送连接的设备。通常指调制解调器。

数据采集板

数据采集板卡直接与现场设备通讯(传感器、控制器等)。您可以在 服务器上安装一个数据采集板，以便直接访问现场设备。

整

整型变量 (Cicode)

一个 4 个字节 (32 位) 的数据类型，允许的数值范围从 2,147,483,648 到 2,147,483,647。

整型变量 (I/O 设备)

一个 2 个字节的 数据类型，允许的数值范围从 -32,768 到 32,767，用来存储数字（例如温度或压力）。某些 I/O 设备也支持其它数字变量，如实型（浮点）数据、字节和二进制码的十进制数。

文

文件服务器

用来存储文件的具有大量数据存储能力的计算机，它可以被其它客户机计算机通过网络访问。在较大的网络上，文件服务器使用特殊的网络操作系统。在较小的安装中，文件服务器可以使用增补了点对点网络软件的 PC 操作系统。

文本框

当在图形画面上添加文字时，是添加到文本框内的。文本框有一些句柄，可以用来处理文字对象。

无

无效值

表示变量不包含有效值。无效变量，返回的 **VarType** 为 1。无效值不同于空值，空值表示变量尚未初始化。它也不同于长度为零的字符串 ("")，长度为零的字符串有时被看成一个无效的字符串。无效不等同于零或空。一个无效的值不被认为大于、小于、或等于任何其它值，包括另外一个无效的值。使用无效值做逻辑比较则返回 FALSE。

无符号整型变量 (I/O 设备)

一个 2 字节（16 位）数据类型，表示的取值范围是 0 到 65,535。支持全部可以使用 INT 类型的 I/O 设备。这也就意味着你可以将任何整数定义成无符号整数以增加其正数值的范围。

显

显示周期

显示周期定义了趋势数据在趋势画面上的显示频率。

最

最大请求长度

在一个单一请求中最大的能从 I/O 设备读取的数据位请求长度。例如，如果最大请求长度是 2048 位，最多能被读取的整型数是： $2048/16 = 128$ 。

服

服务器

连接到（一个或多个）I/O 设备的计算机。当 Citect 系统运行时，服务器与 I/O 设备交换数据，同时将信息按需求分发至其它控制客户机上。局域网 (LAN) 计算机执行处理过程，或者为其它客户机计算机提供可用的资源。在客户机-服务器架构中，分布处理任务可优化性能。

本

本地语言

最终使用者的语言。运行时显示的事物如报警描述、按钮文本、键盘/报警记录、图形文本、Cicode 字符串等等可以被显示成本地语言，哪怕是配置时使用的是开发者的语言(本国语言)。

柱

柱状图

一个发生频率对应数值的柱状图。通常数据形成的分布符合一个像正态分布这样的分布。 .

模

模块 Cicode 变量

专用于变量在其中定义的文件中。这就意味着它只能被这个文件里的函数使用，而不能被其它文件里的函数使用。Cicode 变量被缺省定义为模块，因此无须前缀（不过最好加上前缀 MODULE）。模块变量应在文件起始处声明。

模拟

使用组织好的图形画面的集合对产品系统的可视化表示。 .

模拟量报警

当一个模拟变量达到指定值时被触发。支持四种类型的模拟量报警：高和高高报警；低和低低报警；偏差报警和速率变化报警。

模板

用来构成图形画面的基本图形或省时的模式。每个模板包含画面的基本信息，例如边界和常用的控制按钮。为所有常用画面类型提供了模板。

模糊扩散 (导入位图)

在导入或粘贴的位图中模拟颜色的方法，包含从颜色调色板上组合不同的像素或颜色。

正

正态分布

又被称为“贝尔”曲线，正态分布是最知名和常见的分布。此分布是对称的并通常用于表示机会法则。68.27%的区域位于 -1 sigma 和 +1 sigma 之间，95.45% 位于 -2 sigma 和 +2 sigma 之间，99.73% 位于 -3 sigma 和 +3 sigma 之间。偏斜度的值和峰度的值被用来提供正态的质量测量尺度。假设至少 20 个采样被用来构筑一条分布曲线，一个好的法则就是当 $-1.0 = \text{skewness} = 1.0$ $2 = \text{kurtosis} = 4$ 条件满足时数据被认为是符合正态分布的。

毫

毫秒趋势

允许你使用小于一秒的趋势采样周期。

波

波特率

在一个通讯通道内一个信号每秒钟改变的次数。由于波特率直接影响数据的传输速率，因而此术语经常被错误地用来描述数据传输率。数据传输率的正确描述方法是比特/秒 (bps)。

浏

浏览次序

一系列由浏览次序连接的图形画面，浏览次序是运行时系统中使用“上一页”和“下一页”命令的线性导航次序。

滑

滑钮控制

滑钮控制允许操作员通过拖动一个图形画面上的对象（或对象集合）来改变模拟量变量的值。滑动钮也会随着变量标签的值的自动地移动。

点

点击 (运行时)

当操作员单击对象时，该对象认为被点击的。

点击命令

可以被指定给图形画面上的对象。点击命令允许您通过单击对象把命令发送到运行时系统。

点数限制

从 I/O 设备读取的单独的数字（或模拟）变量。只计算物理点（无论被使用多少次都只计算一次）。点数限制是可以读取的最大数量的 I/O 设备地址，而且是在许可证中指定的。当你运行 Citect 时，你的工程的点数就会根据你的硬件狗的点数限制进行检查。

瓶

瓶颈

当过多的请求被发送至 PLC 通讯连接/数据高速公路上时会导致瓶颈效应。瓶颈效应可能发生在所有的协议上，并且与几个因素有关。这些因素包括请求频率、请求重复的次数(也就是无用的请求)、通讯协议或网络是否支持请求队列。

画

画面环境变量

当进行关联时与特定的画面相关联的只读变量，您要命名该变量并为它指定一个值。当页面在运行时被打开，就会创建该变量。然后它的值就可以被读取。当页面被关闭时，环境变量内存就被释放。

知

知识库

提供超出标准 技术文档范围的高级别技术信息，它定期更新并通过 <http://www.citect.com> 提供。

磁

磁盘 I/O 设备

存在于计算机硬盘或模拟真实 I/O 设备上的磁盘文件。磁盘 I/O 设备中的每一个变量的值存放于计算机硬盘中。磁盘 I/O 设备不连接到工厂的现场设备上。

空

空值

表示变量尚未初始化（指定一个值）。空变量返回 0 变量类型。长度为零的字符串变量 (" ") 不是空，数值为 0 的数据变量也不是。

端

端口

提供连接到 I/O 设备的通讯高速路。

符

符号

存储在库中以备以后检索或使用的对象（或对象组）。通过在库里存储通用的对象，你就减少了存储你的工程所需的磁盘空间大小，而且减少了运行系统所需的内存大小。

精

精灵

如果你有很多相同类型的设备（如 100 个离心泵），而它们的图形显示方式大致相同。那么使用精灵，你就只用对相同的部分进行一次组态就可以了。这时的图形就可以被存为精灵然后为每个设备粘贴一次。

精灵控制器

引用超级精灵的精灵（使用超级精灵函数）。使用精灵控制器，同一个超级精灵可以被不同的应用程序多次使用。

累

累积器

一个可用来跟踪不断增长的运行时数据（例如，马达运行时间、电量消耗和停运时间等）的工具。

线

线程

线程是用来在多任务操作系统中管理任务的同时执行的，使得操作系统能够决定优先权和安排使用 CPU。

组

组合（对象的）

允许你将多个对象组合在一起。每个组合有其独特的属性，而这些属性决定了组合作为一个整体在运行时的行为特征。

缓

缓存 (I/O 设备数据缓存)

启用缓存时，所有从 I/O 设备读取的数据都被临时存储在 I/O 服务器的内存中。如果在缓存时间内从同一台或另一台控制客户端对同样的数据提出了另一个请求，则 I/O 服务器将返回内存中的参数值，而不是再次读取 I/O 设备。

网

网络

一组计算机和外围设备通过一个通讯链接连接在一起。数据和服务(如打印机、文件服务器、和调制解调器)可以在网络上共享。一个 PC 机局域网被称为 LAN。

网络动态数据交换 (NetDDE)

使相互连接的不同计算机上的 Windows 应用程序能够通过普通网络进行通讯。

网络计算机

通过网络适配卡和网络软件连接到一个局域网并运行 的计算机。 .

自

自动组件 (ActiveX 对象)

ActiveX 对象通常包括一个视觉组件(即你在屏幕上所看见的)和一个自动组件。自动组件允许容器对象与 ActiveX 对象之间的相互作用。

自定义报警过滤器

自定义报警过滤器提供了一种过滤和显示有效报警的方法。最多可以为一个报警配置指定八个自定义过滤字符串。与用户自定义的查询函数相结合，操作人员使用自定义过滤器可以辨别并显示需要的有效报警。

节

节点

图形对象的结构锚定点，通常以添加到图形上的小方框形式出现。节点将散布于图形对象的起始、结尾、和任何方向上的变化点。

虚

虚拟

是一个功能上的标识而不是一个物理上的标识。例如，Windows 95 是一个虚拟的桌面系统。

表

表达式

一个有返回值的表达式(或一组表达式)。一个表达式可以是一个单个变量、一个数学公式、或一个函数。

计

计算机

计算机运行 其它工业上常用的称呼可以是节点、机器或工作站。

记

记录名

通常是数据库记录的主属性，在系统中通过其名称引用。数据库记录名必须在任何一种数据库记录里是独一无二的。有时不同类型的记录可以使用同样的名称。尽管如此，为了避免混淆，你应该为你的应用中的每个数据库记录使用一个独一无二名字。为数据库记录指定名称时，名称必须以字母 (A-Z, a-z) 开头，而且只能包

含字母数字 (A-Z, a-z, 0-9) 和下划线 (_)。例如, “Pressure”、 “Motor_10”、和 “SV12_Open” 都是有效的数据库记录名。每个数据库记录名字可最多含有 16 个字符。数据库记录名称不区分大小写, 因此 “MOTOR_1”、 “Motor_1” 和 “motor_1” 是相同的数据库记录名称。因此, 应使用有意义的名字并遵循必要的命名惯例为数据库记录命名。

语

语言数据库

在对工程进行编译时, 会创建一个语言数据库 (dBASE III 格式), 由两个字段组成: **native** 和 **local**。被语言变更指示标记过的文字将会被自动地写进 **native** 字段。接着用户就可以打开数据库并在 **local** 字段中键入翻译好的文字。

质

质量认证 – 等级 1

按照 CiTDriversQA96 质量和认证体系开发的驱动程序, 该体系确保驱动程序的设计、编码和测试符合可能的最高标准。

质量认证 – 等级 2

使用 CiTDriversQA92 驱动程序质量和认证体系开发的驱动程序。

超

超时

一个任务必须在此时间周期内完成。如果在任务完成前已经到了超时时间, 那么任务将被终止。

趋

趋势

一种用图形方式来表示工厂现场一个或多个变量 (或表达式) 的变化的数值。 .

趋势图

趋势图是由一个或多个趋势、名称、注释、比例、时间等来组成的。

趋势服务器

控制趋势信息累积和记录的服务器。此信息提供了工厂当前和历史的景象，而且经过处理可以用来在图形画面上显示或在报表里打印。

趋势线

趋势曲线是趋势画面上用来表示工厂现场变量（或表达式）的变化的数值的实际的曲线。

软

软 PLC

一个纯粹的由软件创造的（虚拟）并仅存于计算机内存中的 PLC。通常为与软 PLC 的通讯（读取和写）操作而提供一个软件接口。也被称为一个“虚拟现场单元”或“虚拟 I/O 设备”。

软件保护

使用插入到计算机打印端口的保护狗来防止许可证侵权。硬件保护狗内包含有你的用户版权的详细信息。当您运行 Vijeo Citect 时，将根据硬件密钥中指定的点数限制检查你工程中的点数。

运

运行系统

监控应用、过程或工厂的系统。运行系统有时被称为人机界面 (MMI)，而且是从一个工程中进行编译的。

远

远程终端

一个与控制它的计算机相隔遥远的终端。计算机和远程终端通过一个调制解调器和电话线来通讯。

远程通讯

两个计算机之间通过一个调制解调器和电话线来相互作用。

通

通信链接

计算机与外围设备之间的连接，允许数据传输。一个通讯的链接可以是网络、调制解调器、或仅是一根电缆。

通讯端口

用来发送和接收串行数据的 PC 端口（也被称为串行或 COM 端口）。

链

链接

库项的副本，处理库原始值的属性。因为是链接的，所以当原始值变化时副本就会更新。

键

键盘命令

键盘命令包括一个操作员在键盘上键入的键序列，以及键入键序列时执行的一个指令（或者一系列指令）。键盘命令可以定义给一个对象或画面，或对整个工程有效。

长

长整型 BCD 变量 (I/O 设备)

一个 4 字节（32 位）的数据类型，值的允许范围为 0 到 99,999,999。4 个字节被分为 8 个 4 位，每个 4 位代表一个十进制数。例如二进制数 0011 代表十进制数 3。因此 BCD 0011 0011 0011 0011 0011 0011 0011 0011 代表 33,333,333。

长整型变量 (I/O 设备)

一个 4 个字节（32 位）的数据类型，允许的数值范围从 2,147,483,648 到 2,147,483,647。

附

附件单元接口 (AUI)

通常被用来通过被称为引入电缆的电缆连接接收器。

集

集群

报警服务器、趋势服务器、报表服务器和 I/O 服务器的离散组。它通常处理本地的 控制客户机。对于拥有多个单独的部门或系统的工厂，可能要使用多个群集，每个群集对应一个部门。

预

预定周期

确定 I/O 服务器连接预定的 I/O 设备以从其读取数据的频率。

驱

驱动程序

用来以不依赖于设备的方式（与监控设备）进行通信，从而使运行时系统能够直接与不同类型设备进行交互。与 I/O 设备通信需要实现通信协议的设备驱动程序。

高

高级报警

当 Cicode 表达式的结果变为 True 时触发。最好在其他报警类型不能满足功能需要时才使用高级报警。如果配置太多高级报警的话，可能会影响系统性能。

索引

A

- ActiveX 对象
 - 标识, 375
- ActiveX 对象类型, 372
- ActiveX.zip, 691
- ActiveX 对象
 - 管理关联数据源, 372
- ANSI 字符集, 620
- ASCII 字符集, 620
- ASCII 设备, 544
 - 格式, 552

C

- cicode 内核命令, 663
- Cicode 对象, 368
- Citect 支持, 208
- citect.ini, 位置, 7
- CiUSAFE 对话框, 654
- cls 内核命令, 664
- COMx 驱动程序, 167
 - 调试, 201
- Cp 指数, 511
- cpk 指数, 511

D

- dBASE 数据库, 576
- dBASE 数据库设备, 544
- dBASE 设备
 - 格式, 553
- dbf 编辑, 593
- DDE 中的文件名, 569
- DDE 会话, 563
 - 配置, 564
- DDE 共享, 572

- DDE 受信任共享, 573
- DDE 服务, 570
- DevAppend() 函数, 555
- DevDelete() 函数, 557
- DevFind() 函数, 556
- DevGetField() 函数, 556
- DevOpen Cicode 函数, 555
- DevSetField() 函数, 555
- DevWrite() 函数, 555
- DspGetEnv() 函数, 306

E

- Excel, 593
- Excel 宏, 575

F

- FAQ
 - Web 客户端, 704
- FAT32 文件系统, 101
- ftp 服务器, 冗余, 156

I

- I/O 服务器冗余, 228
- I/O 设备
 - 写入, 231
 - 属性, 181
 - 磁盘设备, 189
 - 读取, 232
 - 调试, 200
 - 远程多路, 225
- I/O 设备属性, 181
- I/O 设备数据, 错误, 536
- I/O 设备限制, 224
- IFDEF 宏, 隐藏图形对象, 614
- ini 内核命令, 669

L

log 内核命令, 669
LogRead 字段, 181
LogWrite 字段, 181

M

Misc.zip 文件, 691

N

NTFS 文件系统, 101

O

ODBC 服务器, 将 Citect 用作, 587
ODBC 驱动程序, 582, 583
OEM 字符集, 621
OFS 导入标签, 261
OFS 标签导入, 260
OID 验证, 217
OPC 数据访问服务器标签浏览器, 259

P

Pareto 图, 511
projectDBF, 593

R

Runtime 信息, 684

S

SetLanguage() 函数, 616, 619
shell 内核命令, 683
SQL 中的日期, 581
SQL 数据库, 578
SQL 数据库设备, 544
 格式, 553
SQLExec() 函数, 579
syslog 内核命令, 683
syslog.dat 文件, 649

T

TagGen 导入模板, 262

TCP/IP 驱动程序, 调试, 204
TCP/IP: 特殊选项, 169
TrnGetTable() 函数, 535
TrnPlot() 函数, 535
t 标签浏览器: OPC 数据访问服务器, 259

U

Unity 标签导入, 258

V

Vista 兼容性, 643

W

Web Client
 入门, 688
 设置系统, 688
Web 客户机
 Web 部署准备工具, 692
 使用不同于当前系统区域设置的语言, 701
 准备用于部署的工程, 689
 多语言支持, 700
 缺省语言, 700
 设置非缺省语言, 702
Web 客户机的多语言支持, 700
 非缺省语言, 702
Web 客户机的缺省语言, 700
Web 客户端, 687
 ActiveX.zip 文件, 691
 Misc.zip 文件, 691
 准备要交付的用户文件, 691
 创建部署, 694
 删除部署, 699
 功能限制, 690
 常见问题, 704
 显示部署, 698
 更新部署, 699
 简介, 687
 系统架构, 687
 编辑部署, 698
 部署工程, 696
 配置部署, 693

Web 客户端的部署, 696

 创建, 694

 配置, 693

Web 部署准备工具, 692

Windows, 122

Windows Vista 兼容性, 643

Windows 用户安全性, 120

X

XML 模板, TagGen, 262

XP 风格按钮, 358

XRS 控制图表, 509

三

三维 (3D) 效果, 377

不

不兼容的类型错误, 640

不支持的功能, 690

专

专用板卡, 170

专用板卡驱动程序, 调试, 206

中

中心限, 510

串

串口回环测试, 207

串行口回环电缆, 208

串行板卡, 167

串行通讯, 166, 169

事

事件, 483

 属性, 483

 时间和周期, 485

 触发器, 486

 运行, 484

 页面, 305

事件设置画面, CSV_Include 工程, 725

事务, 数据库, 580

互

互联网显示客户端, 645

互联网服务器, 645

交

交换数据, 562, 566

交换颜色对话框, 314

从

从 I/O 设备进行读取, 232

会

会话, DDE, 563

位

位图, 319, 324

使

使用

 串行板卡, 167

 外部数据库, 576

 符号, 318

 精灵, 600

 超级精灵, 602

使用 DDE 公布数据, 567

使用 ODBC 附加数据, 590

保

保存精灵, 599

保护

 包含工程, 99

 顶级工程, 98

保护, 软件, 654

值

值

参数, 541

元

元素, 阵列, 250

共

共享, DDE, 572

内

内插, 趋势, 499

内核, 656

内核命令, 660, 661, 666

cicode, 663

cls, 664

drivertrace, 666

ini, 669

log, 669

shell, 683

syslog, 683

单元页, 681

帮助, 669

暂停, 682

画面内存, 675

画面驱动程序, 673

统计数据, 683

综合页, 670

缓存, 662

表格页, 680

表格页 cicode, 680

表格页 csatopsi.subs, 680

表格页传输, 680

表格页统计, 680

调试, 664

退出, 668

队列页, 675

页面 rdb, 677

内核窗口, 657, 658

冗

冗余

磁盘 (虚拟) I/O 设备, 191

冗余, ftp 服务器, 156

冗余, 报表服务器, 155

冗余, 报警服务器, 154

冗余, 文件服务器, 156

冗余, 服务器, 151

冗余, 网络, 151

冗余, 趋势服务器, 155

写

写

报警属性, 476

写入

I/O 设备, 231

准

准备用于 Web 客户机部署的工程, 689

分

分类, 报警, 431

分组

寄存器, 210

对象, 330

分配变量标签, 234

列

列表, 在报警中显示, 467

创

创建

Web 客户端部署, 694

图形画面, 291

工程, 724

报警画面, 478

报警组, 732

模板, 297

浏览顺序, 297

- 画面, 727
- 精灵, 598
- 自定义菜单, 729, 731

删

- 删除
 - 对象, 329
 - 访问表中的行, 591

刻

- 刻度标记, 716
- 刻度缺省值, 718

前

- 前移一层命令, 331
- 前进按钮, 721

功

- 功能限制, 690

动

- 动作查询, 调用, 591
- 动态数据交换 (DDE), 562
 - 公布数据, 567
 - 写参数值到, 568
 - 和 Office 应用程序, 569
 - 数据类型, 565
 - 读取参数值, 569
- 动态数据交换 (DDE)
 - 连接到标签数据库, 566
- 动点 (AN), 310, 334

包

- 包含工程, 84
 - 保护, 99
- 包含工程对话框, 85
- 包含文件, 84
- 包含的工程, 84

区

- 区域, 326
 - 和安全性, 114
 - 定义, 112
 - 权限, 117, 119
 - 查看, 119
 - 标签用于, 115
 - 组, 115, 116
 - 超级精灵和, 611

十

- 十进制计数法, 248

协

- 协议, 调试, 200
- 协议驱动程序
 - 调试, 205

单

- 单元页内核命令, 681

即

- 即时趋势, 726

历

- 历史
 - 设备, 558
 - 趋势, 501
- 历史文件, 504
- 历史模式, 717

压

- 压缩报表, 537

参

- 参数
 - 值, 541
 - 标签中, 540
- 参数值

从 DDE 应用程序读取, 569
写入 DDE 应用程序, 568
参数查询, 592

双

双重 NIC, 151

发

发送至底层命令, 331

变

变差, 过程, 507

变量

报表中, 535

环境, 306

重映射, 211

变量数据, 在报警中显示, 468

变量标签

分配, 234

属性, 242

插入到字段, 88

数据类型, 242

配置, 241

变量标签名称, 234

只

只读工程, 101

工程效果, 102

可

可见性, 380, 381

后

后移一层命令, 331

后退按钮, 721

启

启动画面, 298

启动网络 DDE 服务, 570

启动配置, 643, 646

周

周期, 指定事件, 485

呼

呼叫, 报警, 9

命

命令

SQL, 579

内核, 660

触击, 406

键盘, 408

命令字段, 560

回

回环测试, 207

回调函数, 654

固

固定文字

报警显示, 467

固定文本

报表, 535

图

图形

导入, 324

图形对象

隐藏, 614

图形画面

位图, 319

创建, 291

定义颜色, 311

屏幕分辨率, 299

属性, 300

颜色定义, 311

图形画面上的颜色, 311

图形页面

为其准备的键盘命令, 303
事件, 305
保存, 292
外观, 302
库, 317
打开, 292
查找, 292
滑钮, 410
环境, 306
符号, 318
缩放, 316
缺省, 306
调整大小, 298

图表

性能: 性能图表, 511
控制, 510

图表, 趋势, 497

圆

圆

绘制, 341

在

在工程中进行故障排除搜索, 96
在报表中换页, 535

地

地址转发, 650

垂

垂直滑钮属性, 412
垂直移动, 383

域

域, 121
域转换, 279

基

基于标签的驱动程序数据验证, 217

填

填充属性 (对象), 325
填充颜色, 393, 394, 395, 399
填充高度, 403

增

增量编译, 636

声

声音报警, 实现, 726

备

备份, 工程, 79
备份工具, 81
备用 .ini 文件, 648

复

复制

对象, 330

工程, 74

复制工程对话框属性, 74

外

外观

图形页面, 302

对象, 324

外部数据库, 使用, 576

外部数据源, 274

多

多

工程, 619

语言, 619

多国语言工程, 616

多数字量报警, 440

多监视器, 使用, 726

多路, 224

多边形对象, 346

多边形属性, 347

多边连线, 绘制, 346
多进程系统
 运行单独的服务器, 644

大

大小, 计算数组, 250

字

字体

 属性, 480
 报表, 535
 自定义, 482

字段

 命令, 560
 报警显示, 468
 报警汇总, 471

字符串数组, 250

字符串替换, 542

字符集, 620

存

存储, 计算磁盘, 502

存储方式, 趋势数据, 502

安

安全, 122

 只读工程, 101
 指定要求, 119

安全性, 6, 107, 120

 MS Office, 570

 包含工程, 99

 将区域用于, 114

 运行时, 105

 顶级工程, 98

宏

宏, 575

定

定义

 区域, 112

 图形画面上的颜色, 311

 报警分类, 431

 标签, 543

 用户权限, 110

 画面属性, 300

 精灵的替换, 599

 访问对象, 414

 超级精灵的替换, 605

 趋势画面, 498

 路径, 505

定位对象, 334

定位对象对话框, 334

客

客户端语法和 DDE, 563

寄

寄存器, 分组, 210

对

对象, 322

 3D 效果, 377

 分组, 330

 删除, 329

 可见性, 380, 381

 垂直移动, 383

 填充颜色, 393, 394, 395, 399

 填充高度, 403

 复制/粘贴, 330

 外观, 324

 定位, 334

 对齐, 332

 应用效果, 377

 操作, 327

 改变大小, 328

 旋转, 333

 旋转属性, 413

- 水平移动属性, 382
- 滑钮, 326, 411, 412
- 移动, 328, 381
- 移动属性, 325
- 管道, 350
- 线条, 323
- 组, 323
- 缩放, 325, 388, 391
- 触击命令, 406
- 访问, 414, 415, 416
- 访问属性, 326
- 输入属性, 326
- 重塑, 323
- 锁定/解锁, 329
- 键盘命令, 409
- 镜像, 334
- 长方形, 338
- 鼠标属性, 407
- 对象, 库, 317
- 对象对齐, 332
- 对象对齐方式, 308, 309
- 对象属性, 324
 - 禁止访问, 417
- 对象类型, 335
 - ActiveX, 372
 - Cicode, 368
 - 多边形, 346
 - 手画线, 335
 - 按钮, 358
 - 数字, 357
 - 文本, 351
 - 椭圆, 341
 - 直线, 337
 - 符号集, 360
 - 管道, 350
 - 粘贴的符号, 370
 - 趋势, 365
 - 长方形, 338
- 对齐对话框, 332
- 对齐对象, 308
- 对齐排列对象, 309, 332
- 对齐网格命令, 308

导

导入

- 图形, 324
- 标签变量, 255
- 导入标签, 同时导入名称和字段, 253

导出

- 标签, 272
- 结果, 92
- 趋势数据, 500
- 导出到文件按钮, 718
- 导出标签变量对话框, 273

屏

- 屏幕分辨率和图形画面, 299

属

属性

- 3D 效果, 377
- 写入报警, 476
- 变量标签, 242
- 垂直移动, 383
- 填充属性, 325
- 填充高度, 403
- 多边形, 347
- 字体, 480
- 对象, 324
- 对象可见性, 381
- 手画线, 336
- 文本, 352, 353, 354, 355, 356
- 旋转, 413
- 椭圆, 342
- 滑钮, 411
- 画面, 300
- 直线, 337
- 移动, 381
- 管道, 351
- 精灵, 602
- 缩放, 388, 391
- 访问, 415
- 长方形, 338
- 鼠标, 407

嵌

嵌套超级精灵, 610

工

工具条

漫游, 721

工具栏

报警, 722

工具栏, 通用, 720

工程

包含, 84

只读, 101

备份, 79

复制, 74

多, 619

多国语言, 616

恢复, 83

打印详细信息, 75

混合, 97

编译, 635

编辑属性, 72

工程, 创建, 724

工程升级, 只读, 103

工程单位, 248

工程恢复和安全性, 107

布

布局, 格式文件, 276

带

带时间标签的报警, 446

带时间标签的报警属性, 446

带时间标签的数字量报警, 457

带时间标签的数字量报警属性, 458

带时间标签的模拟量报警, 461

帮

帮助内核命令, 669

常

常见问题

Web 客户端, 704

常量和超级精灵, 607, 608

库

库, 317

延

延迟报警, 429

开

开始括号预期错误, 640

引

引导线 (对齐方式), 309

引导线设置对话框, 310

归

归档

工程, 79

数据, 560

快

快速通信向导

I/O 设备地址, 174

I/O 设备类型, 173

I/O 设备连接调度, 174

I/O 设备通讯选择, 173

TCP/IP 地址, 174

串行设备, 177

摘要, 178

服务器选择, 173

设备选择, 173

链接到外部数据库, 176

快速通讯向导, 172

呼叫者 ID, 175

性

性能, 209, 684
性能, 过程, 509

恢

恢复工程, 83

手

手动配置, 178
手画线属性, 336
手画线工具, 335

打

打印
 工程详细信息, 75
 趋势数据, 499
打印机设备, 544
 格式, 552
打印管理, 561
打开
 精灵, 599
打开/另存为对话框, 294

扫

扫描时间, 页面, 300

技

技术支持, 1

报

报表, 528
 Cicode, 535
 I/O 设备数据错误, 536
 中的 OLE 对象, 535
 中的注释, 535
 压缩, 537
 固定文本, 535
 字体, 535
 报表格式文件, 535

 换页, 535
 绘图, 535
 表达式和变量, 535
 触发器, 532
 趋势图, 535
 趋势数据, 535
 运行, 530
 通信错误, 536
 配置, 528
报表中的 Cicode 块, 535
报表中的 OLE 对象, 535
报表中的注释, 535
报表中的表达式 (Cicode), 535
报表服务器冗余, 155
报表格式文件, 535
报警, 428
 写入属性所属于的, 476
 分类, 431
 创建画面, 478
 多数字量, 440
 多数字量报警属性, 441
 将属性用作标签, 473
 已配置, 428
 带时间标签, 446
 带时间标签的报警属性, 446
 带时间标签的数字量, 457
 带时间标签的数字量报警属性, 458
 带时间标签的模拟量报警属性, 461
 延迟, 429
 报警分类属性, 432
 数字量, 436
 数字量报警属性, 437
 显示变量数据, 468
 显示顺序, 473
 模拟, 450
 模拟量报警属性, 450
 画面, 显示, 478
 自定义过滤器, 429
 设置, 478
 设置显示格式, 466
 调试硬件, 648
 过滤器, 429

运行期, 478
高级, 455
高级报警属性, 455

报警, 声音, 726
报警: 作为标签支持的属性, 473
报警呼叫属性, 9
报警显示字段, 468
报警服务器冗余, 154
报警标签名称, 234
报警汇总字段, 471
报警画面模板, 714
报警组, 732
报警过滤器, 429

指

指定
 搜索范围, 91
 时间表, 230
指数计数法, 248

按

按钮 (对象), 358
按钮属性对话框, 358

排

排列, 管道, 350

控

控制, 统计, 508
控制上限, 510
控制下限, 510
控制图表, 510
控制菜单页面, CSV_Include, 725

提

提升至最上一层命令, 331

插

插入函数对话框, 89

插入变量标签, 88
插入标签对话框, 88
插入趋势对话框, 368

搜

搜索, 故障排除, 96
搜索文本, 90
搜索范围, 指定, 91

摘

摘要显示, 报警, 473

操

操作对象, 327

支

支持, 1

改

改变
 报警摘要显示, 473
 语言, 616
改变对象大小, 328

故

故障排除通讯, 229

效

效果, 应用三维, 377

数

数值
 字符串转换, 541
数字 (对象), 357
数字变量, 246
数字量报警, 436
数据
 交换, 562, 566
 使用设备读取, 546

- 在报警中显示, 468
- 导出趋势, 500
- 归档, 560
- 打印趋势, 499
- 用不同语言记录, 620
- 缓存, 209
- 编辑, 使用 ODBC, 590
- 设置设备格式, 551
- 读取来自访问表的数据, 589
- 趋势, 490
- 附加, 590
- 数据库
 - 外部, 576
 - 语言, 618
- 数据库事务, 580
- 数据库设备, 555
- 数据文件夹, 位置, 7
- 数据源, 外部, 274
- 数据类型
 - DDE, 565
 - 变量标签, 242
- 数组, 249
 - 超级精灵和, 607
- 数组大小, 250

文

- 文件
 - syslog.dat, 649
 - 包含, 84
 - 备用 .ini, 648
 - 报表格式, 535
 - 更新服务器-客户端, 646
 - 设备历史, 558
 - 趋势历史, 501
 - 重新配置历史记录, 504
- 文件服务器, 冗余, 156
- 文件画面模板, 719
- 文字
 - 报警显示中, 467
 - 标记要改变语言, 617
- 文本

- 查找和替换, 89
- 文本对象, 351
- 文本属性, 352, 353, 354, 355, 356

断

- 断开与数据源的链接, 254

新

- 新建工程对话框, 70
- 新时间同步服务, 10
- 新风格对话框, 297

方

- 方向, 梯度, 338, 342, 347
- 方式, 存储, 502

旋

- 旋转对话框, 333
- 旋转对象, 333
- 旋转属性, 413

日

- 日志文件, 位置, 7
- 日期转换, 593

时

- 时间
 - 在 SQL 中, 581
 - 指定事件, 485
- 时间同步, 77, 78
- 时间表, 指定, 230
- 时间转换, 593

显

- 显示
 - 报警中的列表, 467
 - 报警中的表, 467
 - 报警画面, 478
- 显示, 设置报警格式, 466

普

普通画面模板, 714

暂

暂停内核命令, 682

更

更新

服务器-客户端文件, 646

替

替换

为超级精灵定义, 605

字符串, 542

结果, 93

路径, 504

替换文本, 89, 90

服

服务, 启动网络 DDE, 570

服务器冗余概述, 153

服务器-客户端文件更新, 646

服务器重定向, 650

未

未使用标签, 86

未链接的符号, 317

本

本国语言, 616

本地语言, 616

本机, 6

本机安全性, 120

权

权限, 使用区域, 117

权限/区域结合, 119

权限等级, 112

权限输入对话框, 101

板

板卡

专用, 170

串行, 167

查

查找和替换对话框, 89

查找对象, 334

查找文本, 89

查看

包含工程, 85

工厂区域, 119

查询, 参数, 592

标

标签, 538

参数, 540

名称, 235, 237

名称的语法, 235

命名, 234, 236

定义, 543

导入, 255

导出, 272

将报警属性用作, 473

未使用, 86

用于区域, 115

链接, 253

标签: 支持的报警属性, 473

标签变量

导入, 255

标签名中的前缀, 235

标签名中的区域前缀, 235

标签名称, 236

标签名称中的事件段, 236

标签名称中的特征段, 236

标签名称属性, 237

标签名称语法, 235

标签导入

OFS, 260

OFS, 限制, 261

Unity, 258

标签数据库, 连接, 566

标记符, 234

根

根据不同的 Variable.DBF 重新编译的画面, 217

格

格式文件, 276, 284

格式文件 (导入, 导出, 链接), 275

梯

梯度方向, 338, 342, 347

梯度颜色, 338, 342, 347

检

检测到工程标签不匹配, 217

椭

椭圆对象, 341

模

模式, 演示, 655

模拟量报警, 450

模板, 70

 CSV_Include, 713

 创建, 296, 297

 打开, 296

 报警画面, 714

 文件画面, 719

 普通画面, 714

 用于图形页面, 295

 管理工具页面, 719

 精灵替换, 602

 趋势画面, 715

 链接, 296

正

正方形, 绘制, 338

水

水平移动, 382

浏

浏览顺序

 画面, 297

混

混合工程, 97

添

添加, 用户记录, 108

滑

滑钮, 410, 413

滑钮属性, 326

演

演示模式, 655

漫

漫游工具条, 721

父

父画面按钮, 721

特

特权用户, 724

环

环境

 图形页面, 306

 绘图, 308

环境变量, 306

 超级精灵, 611

用

用户, 121
用户, 特权, 724
用户对话框, 108
用户文件夹, 位置, 7
用户权限, 108, 110
用户记录, 添加, 107, 108

画

画面, 727
 启动, 298
 报警, 478
 显示报警, 478
画面内存内核命令, 675
画面属性对话框, 300
画面风格, 图形, 295
画面驱动程序内核命令, 673

登

登录记录, 107

直

直线对象, 337
直线属性, 337

短

短格式符号, 249

硬

硬件报警
 调试, 648

磁

磁盘 I/O 设备, 189
 设定, 190
磁盘 (虚拟) I/O 设备
 冗余, 191
磁盘存储, 计算, 502

禁

禁止对象访问, 417

移

移动, 381
 垂直, 383
 对象, 328
 属性, 325
 水平, 382

符

符号, 318
符号集对象, 360

等

等级, 权限, 112

管

管理
 打印, 561
管理工具页面模板, 719
管道对象, 350
管道排列, 350

粘

粘贴到剪贴板按钮, 718
粘贴对象, 330
粘贴的符号对象, 370

精

精灵, 597
 使用, 600
 保存, 599
 创建, 598
 和替换, 602
 和超级精灵, 607
 属性, 602
 打开, 599
 替换, 599

标签名称, 611

系

系统

运行, 643

系统调整, 684

累

累积器, 487

配置, 487

线

线条对象, 323

组

组, 120, 121, 122

组 (区域), 115, 116

组 (设备), 546

组, 对象, 323

结

结构化查询语言 (SQL), 578

命令, 579

结构化标签名称, 235, 611

结构查询语言 (SQL)

时间和日期, 581

结果

导出, 92

替换, 93

结果列表 (查找和替换), 91

绘

绘制

圆, 341

多边形, 346

多边连线, 346

按钮, 358

椭圆, 341

正方形, 338

管道, 350

线条, 337

长方形, 338

绘制趋势按钮, 718

绘图

线条, 335

绘图, 在报表中, 535

绘图环境, 308

选项, 310

统

统计控制, 508

统计数据内核命令, 683

统计过程控制 (SPC), 507

综

综合页内核命令, 670

缓

缓存

数据, 209

调整, 684

缓存内核命令, 662

编

编译

增量, 636

工程, 635

调试, 636

编辑

使用 ODBC, 590

工程属性, 72

编辑收藏颜色, 312

编辑菜单项, 731

编辑项目菜单对话框, 731

缩

缩放, 316

缩放对象, 325, 388, 391

缩放按钮, 718

缺

- 缺省
 - 图形页面, 306
- 缺省集群上下文, 65

网

- 网格, 对齐, 308
- 网格设置对话框, 308
- 网络 DDE, 570, 573
- 网络 DDE 共享, 572
- 网络 DDE 共享应用程序, 574
- 网络上的系统, 重新启动, 652
- 网络冗余, 151
- 网络冗余, 配置, 152

自

- 自动缩放按钮, 718
- 自定义字体, 482
- 自定义报警过滤器, 429

范

- 范围标记, 716

菜

- 菜单, 731
 - 创建, 729
- 菜单配置工具, 730

行

- 行, 删除, 591

表

- 表, 显示, 在报警中, 467
- 表格页 `cicode` 内核命令, 680
- 表格页 `csatopsi.subs` 命令, 680
- 表格页传输内核命令, 680
- 表格页内核命令, 680
- 表格页统计内核命令, 680

规

- 规格上限, 511
- 规格下限, 511

角

- 角, 长方形, 338
- 角色, 121, 122

解

- 解锁对象, 329

触

- 触击命令, 406
- 触发器
 - 事件, 486
 - 和报表, 532

计

- 计算
 - 数组大小, 250
 - 磁盘存储 (趋势), 502
- 计算机设置向导, 126
 - CPU 配置, 130
 - 事件, 131
 - 互联网服务器, 128
 - 其它安全, 134
 - 常规选项设置, 134
 - 报表, 129
 - 报警, 129
 - 网络模式, 127
 - 趋势, 130
 - 键盘安全性, 134
 - 集群连接, 132
- 计算机设置向导, 关于, 125
- 计算机设置向导, 为 `CSV_Include` 运行, 725
- 计算机设置向导: 启动函数设置, 132
- 计算机设置向导: 工程, 126
- 计算机设置向导: 计算机角色, 127
- 计算机设置向导菜单安全, 133

设

- 设备, 544
 - ASCII 设备格式, 552
 - dBASE 设备格式, 553
 - SQL 设备格式, 553
 - 使用数据库设备, 555
 - 打印机设备格式, 552
 - 组, 546
 - 设置数据格式, 551
 - 读取数据, 546
 - 配置, 551
- 设备历史文件, 558
- 设置格式
 - 报警显示, 466
 - 数字变量, 246
 - 设备数据, 551
- 设置跨度按钮, 717

访

- 访问
 - 对象, 414
 - 禁止, 416
- 访问属性, 415
- 访问属性, 对象, 326
- 访问表, 读取数据, 589

语

- 语法, ODBC, 584
- 语言
 - Web 客户机, 700
 - 不同记录, 620
 - 在运行期间改变, 619
 - 多, 619
 - 改变, 616
- 语言数据库, 618

调

- 调制解调器, 219, 220
- 调度
 - 通讯, 230

调整

- 系统, 684
- 缓存, 684
- 调整大小
 - 图形页面, 298
 - 缩放的对象, 388
- 调用动作查询, 591
- 调试
 - COMx 驱动程序, 201
 - I/O 设备, 200
 - TCP/IP 驱动程序, 204
 - 专用板卡驱动程序, 206
 - 协议, 200
 - 协议驱动程序, 205
 - 服务器重定向, 650
 - 编译, 636
 - 运行期系统, 648
- 调试内核命令, 664

超

- 超级精灵, 597
 - 使用, 602
 - 区域, 611
 - 和精灵, 607
 - 嵌套, 610
 - 常量和, 607, 608
 - 数组和, 607, 608
 - 替换, 605
 - 标签名称, 611
 - 环境变量, 611

趋

- 趋势
 - 内插, 499
 - 创建画面, 498
 - 图表, 497
 - 存储方式, 502
 - 导出数据, 500
 - 打印数据, 499
- 趋势, 即时, 726
- 趋势, 插入, 368

趋势光标, 717
趋势历史文件, 501
趋势图, 535
趋势对象, 365
趋势数据, 490
趋势数据, 报表, 535
趋势服务器, 140
趋势服务器冗余, 155
趋势标签
 属性, 491
 配置, 491
趋势标签名称, 234
趋势画面模板, 715
趋势组, 733
趋势组文件夹, 718

跨

跨度标记, 717

路

路径定义, 505
路径替换, 504

转

转换
 日期和时间, 593
 由数值到字符串, 541

软

软件保护, 654

输

输入 (键盘命令), 409
输入属性(对象), 326

迁

迁移工具, 34

过

过滤器, 自定义报警, 429
过程变差, 507
过程性能, 509

运

运行
 事件, 484
 报表, 530
 系统, 643
运行单独的服务器, 644
运行时安全性, 105
运行时配置, 643, 646
 运行各个进程, 644
运行期
 处理报警, 478
 显示报警画面, 478
运行期系统
 调试, 648
运行期间
 改变语言, 619

连

连接
 SQL 数据库, 578
 数据库, 566
 网络 DDE 共享应用程序, 574

退

退出内核命令, 668

选

选项, 310

通

通信
 报表错误, 536
通信测试工程, 171
通用工具栏, 720
通讯

串行, 166, 169
故障排除, 229
调度, 230

部

部署 Web 客户端
删除, 699
显示, 698
更新, 699
编辑, 698

配

配置
DDE 会话, 564
ODBC 驱动程序, 583
Web 客户端部署, 693
事件, 483
历史文件, 504
变量标签, 241
报表, 528
报警, 428, 478
累积器, 487
网络 DDE 共享, 572
设备, 551
趋势标签, 491
通信, 178
配置, 启动/运行时, 643, 646
配置多路远程 I/O 设备, 225
配置报警组对话框, 732
配置文件, 位置, 7
配置的报警类型, 428
配置趋势组对话框, 733

采

采样周期, 491

重

重塑对象, 323
重新启动系统, 651
重映射

变量, 211

链

链接
标签, 253
模板, 296
链接, 断开, 254
链接的符号, 317

锁

锁定对象, 329

错

错误消息
查找与替换, 94

键

键盘命令, 303, 408

镜

镜像对话框, 334
镜像对象, 334

长

长文件名 (用于 DDE), 569
长方形对象, 338

队

队列页内核命令, 675

限

限制
Cicode 函数, 690
功能, 690

隐

隐藏
图形对象, 614

对象, 380

集

集群

 集群属性, 138

集群环境, 65

页

页面 rdb 内核命令, 677

页面, 预先配置的, 713

页面事件, 305

页面模板, 295

顶

顶级工程, 98

预

预先配置的页面, 713

颜

颜色, 填充, 393, 394, 395, 399

颜色, 梯度, 338, 342, 347

颜色字体, 定义, 482

驱

驱动程序

 ODBC, 583

 ODBS, 582

驱动程序信息, 216

验

验证基于标签的驱动程序数据, 217

高

高度, 填充, 403

高级安全设置对话框, 101

高级报警, 455

鼠

鼠标属性, 407