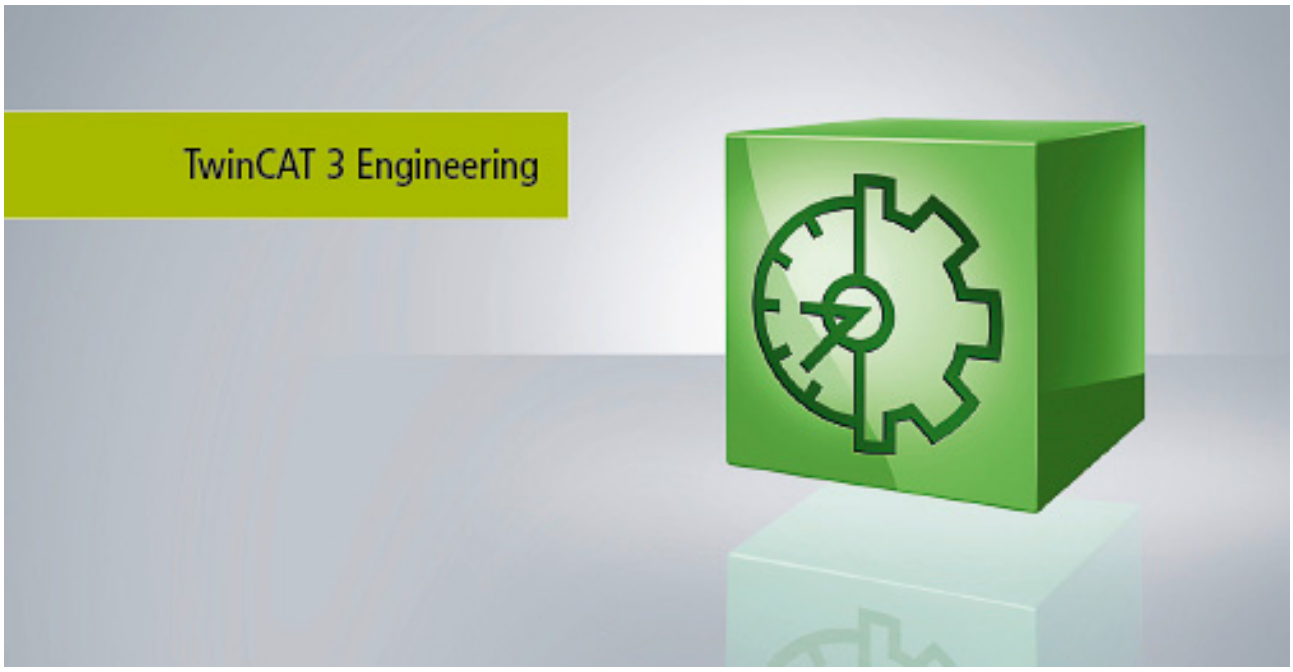


手册

EAP

TwinCAT 3

版本： 1.6
日期： 2020-06-16

BECKHOFF

目录

1 前言 5
1.1 文档说明 5
1.2 安全说明 5
2 产品描述 7
2.1 基本原则 7
2.1.1 通讯方式 9
2.1.2 通过 ARP 进行远程站监控 11
2.1.3 EAP 发送机制 11
2.1.4 EAP 性能 13
2.1.5 EAP 状态机 14
2.2 技术概念 15
2.2.1 EAP 报文结构 15
3 EAP 连接诊断 19
3.1 订阅服务器 19
3.2 发布服务器 21
4 创建 EAP 配置 23
4.1 添加 EAP 设备 23
4.2 添加发布服务器变量 24
4.3 添加订阅服务器变量 28
4.4 使用自定义数据类型 33
5 配置 EAP 设备 37
5.1 TwinCAT EAP 设备 37
5.2 发布服务器框 39
5.3 发布服务器变量 42
5.4 订阅服务器框 43
5.5 订阅服务器变量 43
5.6 TwinCAT 2 和 3 之间的 EAP 45
6 CANopen 对象字典 46
6.1 EAP 对象字典 (子配置文件 1000) 46
6.2 EAP 设备的 TwinCAT ADS 接口 57
6.3 ADS over EtherCAT (AoE) 58
6.4 TwinCAT EAP 设备的在线配置 60
6.5 轮询数据交换的配置 63
6.6 还原在线配置 63
7 EAP 设备配置 (EDC) 文件 65

1 前言

1.1 文档说明

本说明仅供熟悉适用国家标准的控制和自动化工程专家使用。
在安装和调试元器件时，必须遵循本文档及以下注意事项和说明。
技术人员应负责在每次安装和调试时使用已发布的文档。

负责人员必须确保所述产品的应用或使用符合所有安全要求，包括所有相关法律、法规、准则和标准。

免责声明

本文档经过精心准备。然而，所述产品正在不断开发中。
我们保留随时修改和更改本文档的权利，恕不另行通知。
不得依据本文档中的数据、图表和说明对已供货产品的修改提出赔偿。

商标

Beckhoff®、TwinCAT®、EtherCAT®、EtherCAT G®、EtherCAT G10®、EtherCAT P®、Safety over EtherCAT®、TwinSAFE®、XFC®、XTS® 和 XPlanar® 均为倍福自动化有限公司的注册商标并由公司授权使用。
本出版物中使用的其他名称可能是商标，第三方出于自身目的使用它们可能侵犯商标所有者的权利。

正在申请的专利

涵盖 EtherCAT 技术，包括但不限于以下专利申请和专利：:
EP1590927、EP1789857、EP1456722、EP2137893、DE102015105702
包括在其他各国家的相应专利申请或注册。

EtherCAT® 是注册商标和专利技术，由德国倍福自动化有限公司授权使用

版权所有

© 德国倍福自动化有限公司
未经明确授权，禁止复制、分发、使用本文档及擅自将内容与他人交流。
违者将承担赔偿损失的责任。在专利授权、工具型号或设计方面保留所有权利。

1.2 安全说明

安全规范

请注意以下安全说明和阐述！
可在以下页面或安装、接线、调试等区域找到产品相关的安全说明。

责任免除

所有元器件在供货时都配有适合应用的特定硬件和软件配置。禁止未按文档所述修改硬件或软件配置，德国倍福自动化有限公司不对此承担责任。

人员资格

本说明仅供熟悉适用国家标准的控制、自动化和驱动工程专家使用。

符号说明

在本文档中，下列符号随安全指示或说明一起使用。必须仔细阅读并严格遵守安全说明！

⚠ 危险

严重受伤的风险！

未遵守带有此符号的安全说明将直接危及人员生命和健康。

⚠ 警告

受伤的风险！

未遵守带有此符号的安全说明将危及人员生命和健康。

⚠ 谨慎

人身伤害！

未遵守带有此符号的安全说明可能导致人员受伤。

注意

危害环境或损坏设备

未遵守带有此符号的安全说明可能危害环境或损坏设备。

提示或指示

此符号表示该信息有助于更好地理解。

2 产品描述

EtherCAT 自动化协议 (EAP) 设备允许在通过以太网连接的 PC 之间周期性地高度准确交换任何所需变量。EAP 设备之间的通讯根据发布服务器/订阅服务器原则进行，并由 EtherCAT 技术组 (ETG) 指定 (ETG 1005 – 参见网页 www.ethercat.org)。

为进行高度准确的通讯，必须为 TwinCAT EAP 设备安装 TwinCAT 实时以太网驱动程序。

与 TwinCAT 2 网络变量进行比较

TwinCAT EAP 设备基于 TwinCAT 2 中熟悉的网络变量 (NWV)，并包含一些扩展。除其他改变外，EAP 报文还稍稍扩展了 NWV 报文。但是，这一扩展只涉及报文内容。EAP 报文的结构仍与 NWV 报文相同。因此，网络变量与 EtherCAT 自动化协议兼容，反之亦然。有关 TwinCAT 2 和 3 之间 EAP 通讯的更多信息，参见章节 TwinCAT 2 和 3 之间的 EAP [▶ 45]。

前提条件

TwinCAT 3.1 版 (版本 4018.13) 或更高版本中提供了本文档所述的 TwinCAT EAP 设备全部功能范围。

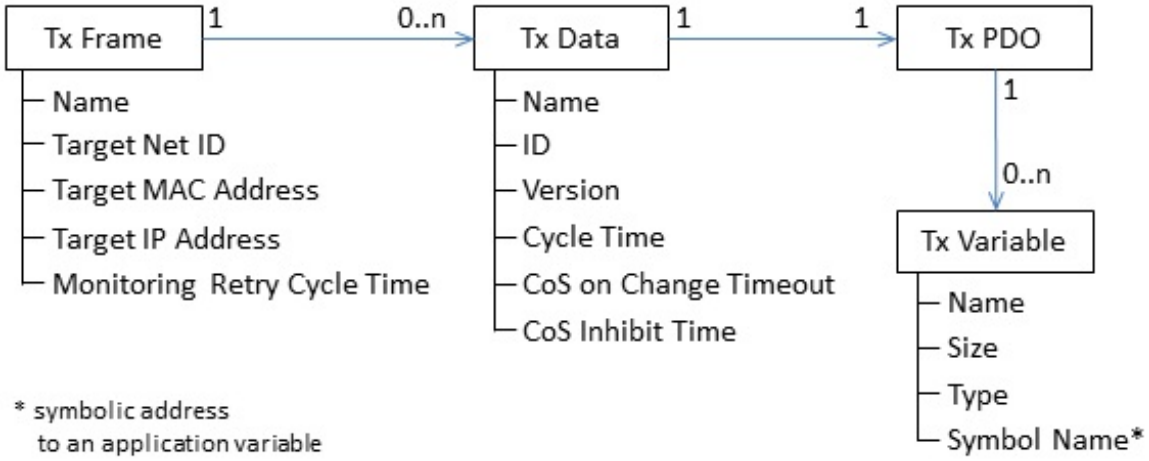
2.1 基本原则

TwinCAT EAP 设备支持通过网络将来自 TwinCAT 控制器 A 的任何变量数据传输到 TwinCAT 控制器 B。这些变量通常用于控制机器内部的处理操作。因此，它们也被称为过程变量 (*PV*)。对于 TwinCAT EAP 设备，可以通过一个由 TwinCAT 实时以太网驱动器支持的标准网络适配器进行发送和接收。

EAP 设备之间的通讯基于发布服务器/订阅服务器原则进行。发送器 (称为发布服务器) 将信息发送到全部或多个网络设备；通常，发布服务器不知道是哪个接收器，或不知道接收器是否存在。另一方面，接收器 (称为订阅服务器) 对某些信息感兴趣并接收这些信息，却不知道它们源自哪个发布服务器，或不知道这个发布服务器是否存在。

EAP 发布服务器的结构

EAP 发布服务器由许多嵌套元素组成，如下图所示。最低级别的基本元素是 *Tx 变量*。它定义了一个输出变量，该变量链接到一个过程变量，并具有多个其他属性，例如数据类型。可以自由选择数据类型；它可能是一个大小为几百字节的复杂数据类型。唯一条件是不得超过 EAP 帧的最大尺寸 (EAP 帧的大小相当于标准以太网帧的大小)。操作过程中，过程变量提供的值由发布服务器发送。



在下一个更高的级别，*TxVariables* 被 *TxPDO* 元素 (TxPDO = TxProcessDataObjects) 引用。一个 *TxPDO* 可以引用多个 *TxVariables*，因此可以将他们合并到一个对象中。然后，*TxPDO* 定义一组有序的 *TxVariables*。不得超过 EAP 帧的最大尺寸这一条件也适用于 *TxPDO*。

TxData 元素 (*TxProcessData = TxPD*) 位于下一个更高级别。它代表*发布服务器变量*，并被理解为 EAP 中发布服务器的通讯单元。*TxData* 元素引用一个特定的 *TxPDO*，并定义许多属性，例如*发布服务器变量*的 *ID*、其*版本*和时钟周期，首先根据这些属性发送*发布服务器变量*。根据这些属性，*发布服务器变量*在发送器上定义一个对象，因此，必须在接收器上定义一个合适的订阅服务器变量，以便可以成功进行数据交换。

数据传送基于网络，通过以太网协议或 UDP/IP 进行。类似地，随后为 *TxFrame* 分配一个 TxData 列表，这些数据将被发送到相同的目标地址。*TxFrame* 受限于每个数据包的最大数据长度。要发送*发布服务器变量*，至少必须定义以下属性：

目标地址：
目标地址通常为多播地址，以便将*发布服务器变量*自动发送到一组接收器。也可以输入单个接收器的地址。

ID：
每个*发布服务器变量*都定义了一个数字，该数字在网络中必须是独一无二的。基于该数字，*发布服务器变量*可以被*订阅服务器*识别。

时钟周期：
时钟周期定义了发送*发布服务器变量*的时间间隔。EAP 周期范围通常在几毫秒到几百毫秒之间。

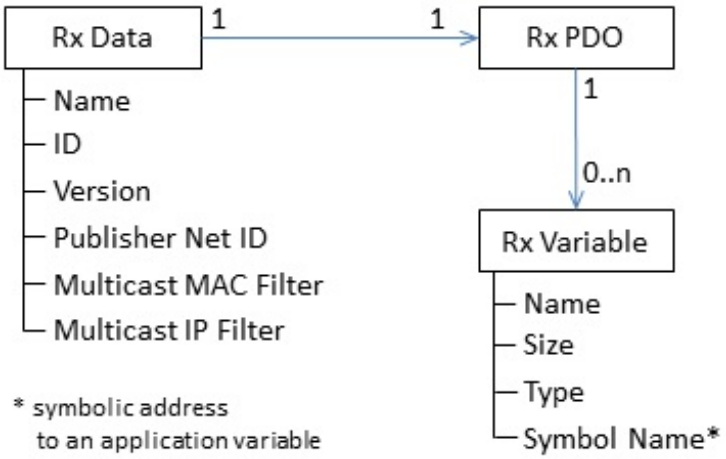
过程变量链接：
最后但同样重要的一点是，需要在*发布服务器变量*与*过程变量*之间建立链接，以确保过程数据借助*发布服务器变量*被实际发送。否则，*发布服务器变量*的值始终保持为零。

EAP 订阅服务器的结构

EAP 订阅服务器的结构与发布服务器的结构相似，如下图所示。订阅服务器最低级别的基本元素称为 *RxVariable*。RxVariable 定义了一个输入变量，该变量也链接到一个过程变量，并包含多个属性，例如数据类型。在操作期间，过程变量获得订阅服务器接收的值。

相应的，上述级别的元素称为 *RxPDOs* (RxProcessDataObjects)。每个元素定义一组有序的 *RxVariables*。

RxData 元素 (*RxProcessData = RxPD*) 位于下一个更高级别。它代表一个*订阅服务器变量*，并被理解为 EAP 中订阅服务器的通讯单元。*RxData* 元素引用一个特定的 *RxPDO*，并定义所需属性 (ID 和版本)，该属性必须与要接收的*发布服务器变量*匹配。为了成功进行数据交换，所引用 *RxVariable* 的数据类型及其在 *RxPDO* 中的顺序必须与*发布服务器变量*的 *TxPDO* 相同。因此，*发布服务器变量*在接收器上定义一个对象，为此必须在发送器定义一个匹配的*发布服务器变量*，以便进行数据交换。



由于 EtherCAT 自动化协议的设计，对于订阅服务器而言，所接收的数据与发送器并不相关。特别是，在一个帧内发送哪些*发布服务器变量*不相关。因此，订阅服务器没有将特定*订阅服务器变量*合并为一个单元的帧元素或类似元素，因此只能被整体接收。但是，如果订阅服务器只应接收特定发送器的*发布服务器变量*，则 *RxData* 提供将 *AMS NetID* 定义为筛选器地址的选项。在这种情况下，至少必须为*订阅服务器变量*定义以下属性：

ID：
*订阅服务器变量*的 *ID* 定义哪个*发布服务器变量*应该接收它。*ID* 是一个数字，对于整个网络中的每个*发布服务器变量*而言，它都应该是独一无二的，其用于在接收端识别*发布服务器变量*。

过程变量链接：
最后，只有链接到过程变量，使用*发布服务器变量*才有意义。只有这样，接收到的数据才会被过程变量实际应用，并在机器控制中被考虑在内。

此外，*订阅服务器变量*的数据长度必须与*发布服务器变量*的数据长度相同。否则，接收的*发布服务器变量*将被弃用。

2.1.1 通讯方式

TwinCAT EAP 设备支持两种通讯类型：周期性过程数据通讯（EtherCAT 类型 4）和非周期性 EtherAT 邮箱通讯（EtherCAT 类型 5）。对于邮箱通讯，TwinCAT EAP 设备仅支持 AoE 协议（AoE - ADS over EtherCAT）。AoE 协议规范在 EtherCAT 协议增强版（ETG 1020）中进行描述。对于过程数据通讯，有两种不同的通讯模式：

推送数据交换模式，在这种模式下，EAP 发送器周期性地或在检测到状态变化时将其过程信息发送到网络，并且 EAP 接收器预期到此过程信息并相应地将其接收。此模式与 TwinCAT 2 网络变量（NWV）的发布服务器/订阅服务器原理相同。
轮询数据交换模式，在这种模式下，EAP 客户端将带有其过程信息的请求报文发送到 EAP 服务器，然后服务器以响应报文的形式将其过程信息返回到 EAP 客户端。

此外，TwinCAT EAP 设备在处理数据通讯过程中，支持不同的连接类型和不同寻址模式。支持的连接类型包括：

- **单播**：EAP 信息从一个终点发送到另一个终点，即该信息被精确发送到一台 PC。
- **多播**：EAP 信息从一个终点发送到多个终点，即该信息被发送到一组 PC。
- **广播**：EAP 信息从一个终点发送到所有可访问的终点，即该信息被发送到所有设备。

可以使用 MAC 地址、AMS NetID 或 IP 地址。根据连接类型和寻址方式的配置，为 EAP 过程数据通讯启动特定的网络协议。具体分配如下表所示。

表 1: 网络协议

寻址模式 连接类型	MAC 地址	AMS NetID	IP 地址
单播	以太网协议	以太网协议	UDP/IP
多播	以太网协议	*不支持*	UDP/IP
广播	以太网协议	*不支持*	UDP/IP

根据不同寻址模式（MAC、AMS NetID 和 IP），支持以下连接类型的单播、多播和广播：

MAC 寻址：
EAP 信息基于以太网协议传输。将接收消息的网络适配器 MAC 地址配置为目标地址。在该寻址模式下，EAP 信息无法从一个路由器转发到另外一个子网络，因为它基于 IP 地址运行。因此，该消息只能通过交换机在子网络内发送。

广播和多播

为广播或多播信息保留专用的 MAC 地址：

广播 MAC：FF-FF-FF-FF-FF-FF

多播 MAC：多播 MAC 地址必须符合以下条件。

- 第一个字节的最低顺序位（第 1 位）的值为 1（组位）。
- 如果 MAC 地址是全局唯一的，则第 2 位的值为 0；
  如果该地址仅为本地唯一，则值为 1。
- 前 24 位（第 3 至 24 位）对应于制造商 ID，称为组织唯一标识符（OUI）。Beckhoff 的 OUI 为“00-01-05”。
- 剩余 24 位（第 25 至 48 位）可以为每个接口单独指定。为 EtherCAT 自动化协议定义了序列“04-00-00”。

⇨ 为 TwinCAT EAP 设备生成的标准多播 MAC 地址为 01:01:05:04:00:00。

AMS NetID 寻址：
EAP 信息基于类型 4 EtherCAT 协议（EAP）传输。根据地址解析协议（ARP）和配置的 AMS NetID 确定所需的目标 MAC 地址。与 MAC 寻址一样，EAP 信息只能在子网络内发送。

通过 AMS NetID 进行通讯

使用 *AMS NetID* 作为目标地址的优点在于它是逻辑地址。使用已配置的 *AMS NetID*，借助特定的 ARP 请求来确定目标设备的 *MAC* 地址。

即使更换了控制计算机或计算机网络适配器，例如导致 *MAC* 地址改变，也不必更改 EAP 连接配置。唯一条件是为新的控制计算机分配原始 *AMS NetID*。

如果配置了连接类型*单播*，则默认情况下还将配置*订阅服务器监控*机制（参见通过 ARP 进行远程站监控 [▶ 11]）。

IP 寻址：
对于 EAP 信息，互联网协议（IP）连同用户数据报文协议（UDP）一起用于收件人的转发和寻址。根据地址解析协议（ARP）和配置的 IP 地址确定所需的目标 MAC 地址。通过 UDP/IP 寻址，路由器可以将 EAP 信息转发到其他子网络（例如，包括互联网）。

为广播或多播信息保留专用的 IP 地址：

广播 IP：255.255.255.255 被指定为广播 IP 地址。广播 MAC 地址 FF-FF-FF-FF-FF-FF 直接派生于此 IP 地址。

多播 IP：多播 IP 地址必须在 224.0.0.0 到 239.255.255.255（IPv4）之间。在 EAP 设备中，TwinCAT 为每个已配置的*多播 IP 地址*生成一个兼容的*多播 MAC 地址*，该地址在 TwinCAT 启动时（即在运行模式启动时）使用。

推送数据交换（n:m 连接）

推送数据交换模式基于与 NWV 传输相同的模型（发布服务器/订阅服务器原则）。它提供了网络中的 n:m 连接选项。每个 EAP 设备可发送一个或多个 EAP 报文，及其输出过程数据（*TxData*）。每个 EAP 设备可以“侦听”，以确定接收的 EAP 报文中包含的过程数据是否与其输入过程数据（RxData）相匹配，以便在适用的情况下对其进行处理。因此，同一个 EAP 设备可以发送和接收过程数据。这样就可以建立双向通讯。

通过推送数据交换，可以根据需要自由选择每个已配置 EAP 报文的寻址模式（单播、多播或广播）。

轮询数据交换（1:1 连接）

轮询数据交换模式遵循客户端/服务器结构原则。借助该结构，它可以实现“软”同步。EAP 设备可以同时充当客户端和服务器。

轮询模式连接类型

对于轮询数据交换模式，只有单播连接类型是唯一定义的。

单播（1:1 连接）
客户端将 EAP 报文及其输出过程数据一同发送到服务器，然后服务器通过单独的 EAP 报文将其输入过程数据返回至客户端。

网络协议

以太网协议
以太网协议负责交换网络中的数据包。它处理 OSI 层 1 和 2（物理层和数据链路层）的任务。以太网协议标题应包含发送器地址、接收器地址和以太网类型，该类型指定了下一个更高的 OSI 层使用哪一个协议。发送器和接收器地址以 MAC 地址的形式输入。MAC 表示*介质访问控制*，这里指在生产过程中分配给每个以太网设备的唯一硬件地址。例如，可以为 Beckhoff PC 的以太网端口分配 MAC 地址 00:01:05:34:05:84；“00:01:05”是 Beckhoff ID，第二部分在生产过程中指定。发送器和接收器 MAC 地址确定了网络中两台 PC 之间每个以太网报文的路由。以太网报文可以通过交换机进行进一步处理，但通常不能通过路由器进行处理。

用户数据报文协议/互联网协议（UDP/IP）
接收器通过以太网报文中的附加 IP 标题识别，以便可以由路由器进行进一步处理。该报文具有以太类型 0x0800，该类型指定报文为 IP 报文。在后续 UDP 标题中，端口编号 0x88A4 用于源端口和目标端口。根据此端口编号，TwinCAT 系统检测到该报文为实时用户数据报文。

TwinCAT 基于以太类型 0x88A4（如果使用以太网协议）或目标端口 0x88A4（适用于 UDP/IP）识别 EAP 报文。相应的，TwinCAT 以太网驱动器使接收到的 EAP 报文绕过操作系统的 NDIS 堆栈，因此，TwinCAT 优先将其视为 Beckhoff 实时以太网报文。发送 EAP 报文时，也会绕过操作系统的 NDIS 堆栈。

一旦 TwinCAT PC 接收到 EAP 报文并将其识别出来，在进一步处理报文时，报文中传输的*过程数据*（*PD*）将被分配给 EAP 设备中配置的 RxData。此分配基于 *PD* ID。如果接收器未配置带匹配 *PD* ID 的 RxData，则丢弃接收的 *PD*。

最后，只有当接收到的 *PD* 数据长度和版本编号与预期的数据长度和版本编号匹配时，才会应用 *PD* 各个*过程变量*（*PV*）的值。

2.1.2 通过 ARP 进行远程站监控

EAP 基于无连接协议（以太网协议和 UDP/IP）。这些协议不返回消息确认。TwinCAT EAP 设备使用地址解析协议（ARP）进行远程终端监控，以使 EAP 报文发送器能够检测到接收器不再可用。EAP 发布服务器中可使用 *ARP 重试间隔*来配置时帧，用于检查接收器是否仍可访问。只有配置单播连接时，才可启用远程终端监控（*订阅服务器监控*）。

如果启用了*订阅服务器监控*，发布服务器将根据配置的时间间隔向配置的目标设备发送 *ARP 请求*报文。如果接收器仍按预期运行，则以 *ARP 答复*报文响应。否则，将无响应。在诊断变量 FrameState（参见发布服务器）中，如果发生错误，则设置第 3 位（0x0004）。

ARP 处理

将 MAC 地址分配给网络地址（IP 地址）的 ARP 处理由操作系统（Windows）进行。将 MAC 地址分配给 AMS NetID 的 ARP 处理由 TwinCAT 系统进行。

2.1.3 EAP 发送机制

EAP 报文的发送基于触发机制触发。EAP 设备的配置用于确定此*触发*机制的工作方式。为每个 *TxData* 定义一个*触发*条件。如果满足此*触发*条件，将通过 EAP 报文发送 *TxData*。换句话说：在每个 EAP 设备中，*触发*条件用于每个 *TxData*，以配置*触发*机制的操作。

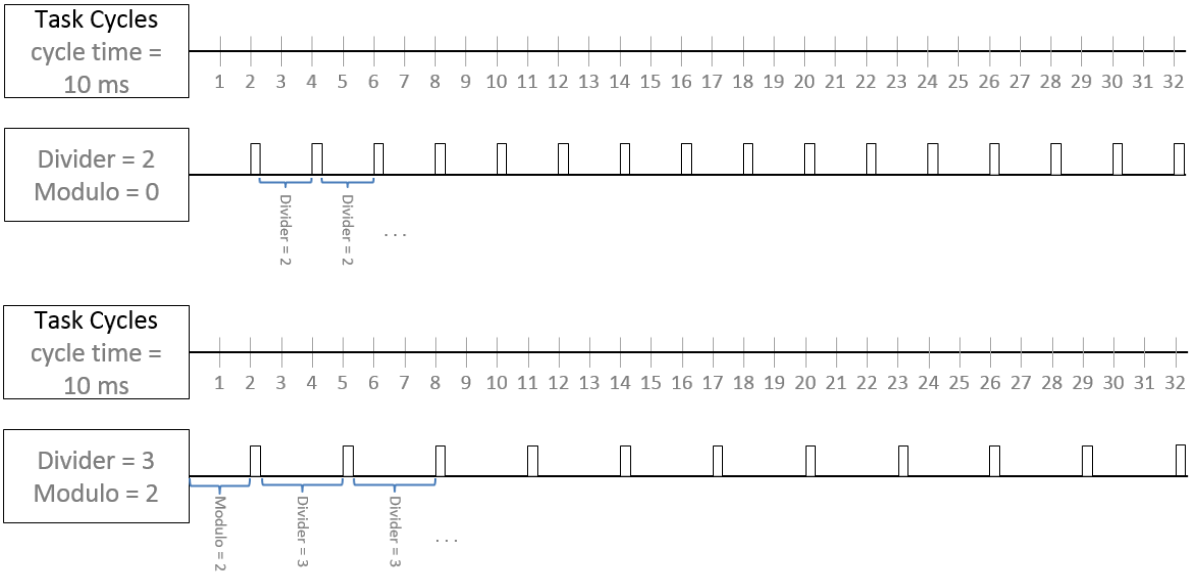
有 5 种不同类型的触发条件，如此处所述。

触发条件的叠加

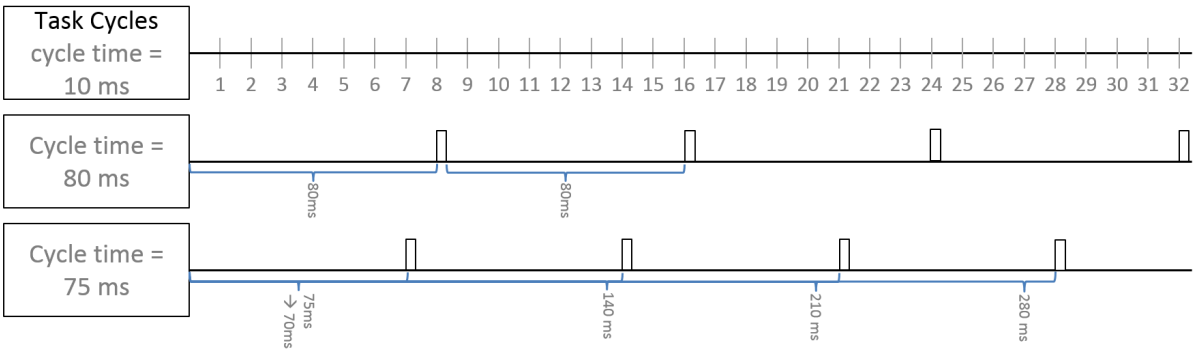
各个触发条件的说明指示需要停用哪些其他触发条件。换句话说，哪些条件不允许组合在一起。以下示例进一步显示了多个触发条件相互重叠的情况。未明确定义它们如何重叠。因此，建议禁用所有不允许的触发条件。

1. **轮询请求 Rx PD**
   触发条件*轮询请求 Rx PD* 以轮询数据交换模式控制响应报文的发送（参见章节通讯方式 [▶ 9]）。一旦 *TxData* 具有触发条件*轮询请求 Rx PD* 的有效条目，相应的 *TxData* 将在此模式下运行。如果与 EAP 设备中配置的 *RxData* 对象索引相匹配，则存在有效条目。然后，此 *RxData* 定义将返回 *TxData* 作为响应的预期请求。如果 EAP 设备接收到包含预期*过程数据*的 EAP 报文，在下一个周期中，*TxData* 将以新 EAP 报文的形式返回至请求报文的发送器。因此，当触发条件*轮询请求 RxData* 启用时，EAP 设备充当此 *TxData* 的*轮询数据交换*服务器。
   如果启用了*轮询请求 Rx PD* 条件，则必须禁用所有其他条件（2 到 5）。
2. **分隔符/模数**
   *分隔符/模数*条件用于指定发送 *TxFrame* 或 *TxData* 的频率（参见下图）。该频率始终是驱动 EAP 设备任务周期的倍数。*分隔符*值定义该倍数。*模数*值指定首次发送 *TxFrame* 或 *TxData* 的起始周期。如果*分隔符*的值为 0，则禁用此条件。

如果启用了*分隔符/模数*条件，则条件 3、4 和 5 不相关；而应该被禁用。必须禁用条件 1。



3. **周期**
   TxData 按*周期*值（单位：µs）定义的特定间隔发送（参见下图）。*周期*应该是任务周期的整数倍。如果配置的值不是任务周期的整数倍，则将自动设置下一个较小的倍数，可降至 0（如有必要）。如果值为 0 µs，则禁用此条件。
   如果启用了*周期*触发条件，则应禁用触发条件 1、2、4 和 5。

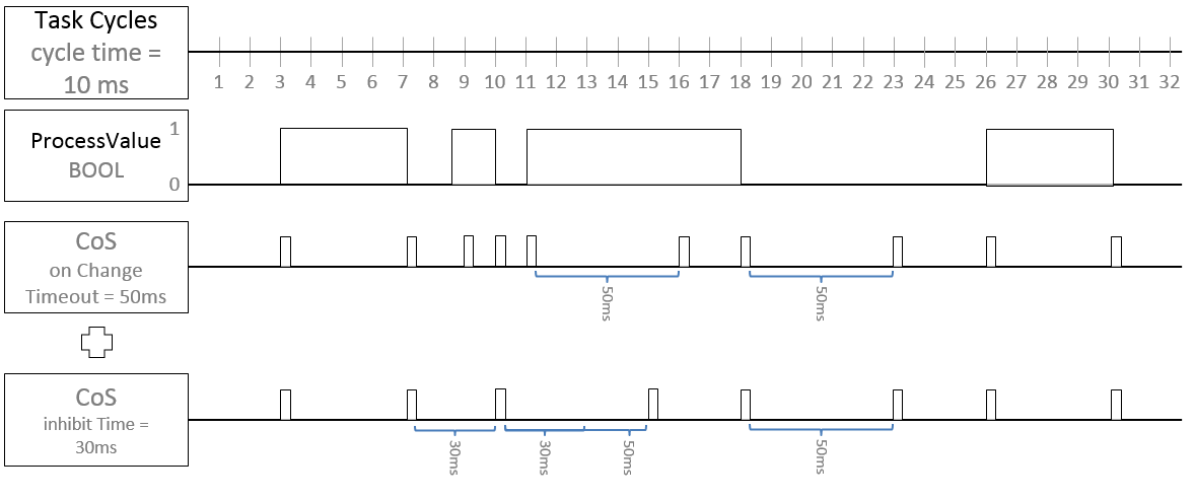


周期与任务周期之间的关系

假设任务周期为 5 ms（5000 µs），则*过程数据* A 的周期为 10000 µs，*过程数据* B 的周期为 20000 µs。现在，任务周期从 5 ms 降至 15 ms（15000 µs）。*过程数据* A 和*过程数据* B 的周期都不是任务周期的倍数；因此周期无法被任务周期整除。
所以，*过程数据* A 仅每 15 ms（15000 µs）发送一次，*过程数据* B 仅每 30 ms（30000 µs）发送一次。

4. **状态改变（CoS）：改变超时**
   TxData 仅在其中一个变量的值与前一个变量的值相比发生变化时才发送。最大时间间隔定义为超时时间（单位：µs）。如果变量值在此时间间隔内没有变化，则在时间间隔结束后，无论如何都将发送

TxData（参见下图）。时间间隔值必须是任务周期的整数倍。如果时间间隔设置为 0 µs，则会禁用触发条件 *CoS 改变超时*。如果启用了触发条件 *CoS 改变超时*，则必须禁用触发条件 1、2 和 3。



5. **状态改变（CoS）：禁止时间**
   *禁止时间*指定最小时间间隔，这样，在上次发送后，*TxData* 不会在该时间间隔结束前再发送。
   因此，*禁止时间*指定最小时间间隔（以 µs 为单位），即使所含 *Tx 变量*的一个值改变，也会在该时间间隔后发送 *TxData*（参见下图）。此时间间隔的值只能是任务周期的整数倍，并且必须小于 *CoS改变超时*的值。如果时间间隔设置为 0 µs，则会禁用触发条件*禁止时间*。
   如果启用了禁止条件*禁止时间*，则应禁用触发条件 1、2 和 3。

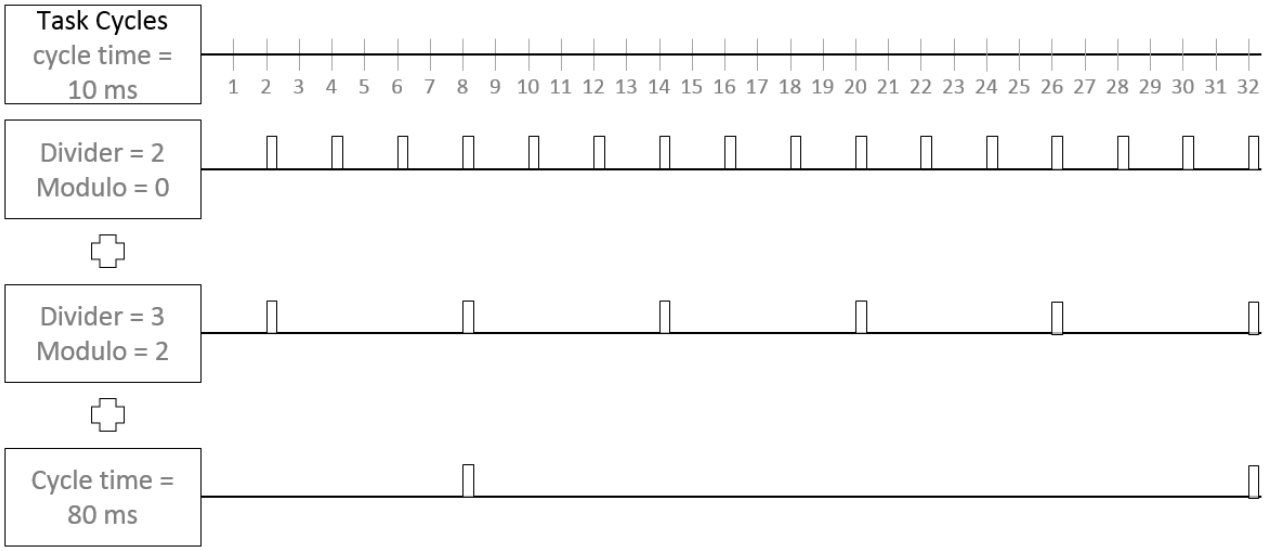
触发条件配置选项

触发条件 1、3 和 5（轮询请求 RxData、周期和禁止时间）可通过 EAP 对象字典（参见 TwinCAT EAP 设备文档中的章节 CANopen 对象字典 [▶ 46]）进行配置。

触发条件的特殊性

对于定义时间间隔的所有触发条件，此间隔不能小于 EAP 设备驱动任务的任务周期时间。

不建议将这些条件组合使用，因为这些条件没有被明确定义。以下提供了很好的复杂性示例：



最后一行清楚表明，160 ms 和 240 ms 的传输不会发生，因为它被附加的分隔符/模数条件阻止。

2.1.4 EAP 性能

如果使用 TwinCAT EAP 设备，则必须考虑所用网络结构的时间边界条件：

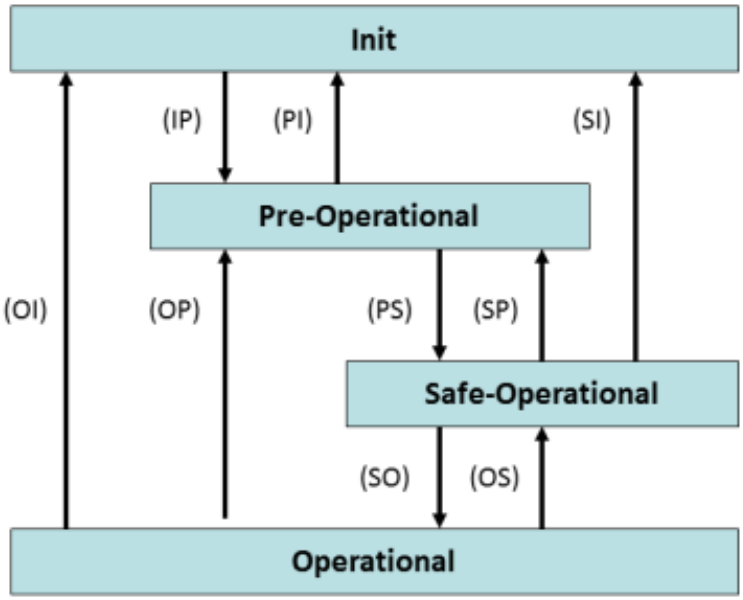
- 在仅通过交换机（例如，按照以太网协议）发送报文的网络结构中，可以实现约 10 ms 或以下的通讯周期。
- 在也可经由路由器（即通过 UDP/IP）发送报文的网络结构中，可以实现约 100 ms 的通讯周期。

在并行进行其他通讯的网络中，EAP 通讯性能可能受到影响。

2.1.5 EAP 状态机

EAP 状态机（EAP SM）控制 EAP 设备的状态。根据状态，EAP 设备可以访问或执行不同的功能。以下状态之间有所区别：

- 初始化
- 操作前
- 安全操作和
- 操作



每个 EAP 设备启动后的常规状态为 *OP* 状态。在达到 *OP* 状态前，EAP 设备将依次切换到每个状态。在每个状态跳转期间，EAP 设备都会执行特定操作。如果在跳转期间发生错误，则无法将设备切换至相应的后续状态，因此将保持最后到达的状态。可读错误代码用于诊断错误原因。

初始化

通常，EAP 设备的*初始化*状态是临时状态。即，EAP 设备无法明确设置为*初始化*状态。但是，在某些情况下，SM 会将 EAP 设备重置为*初始化*状态。在这种状态下，邮箱通讯和 EAP 设备的过程数据通讯均无法进行。

操作前（Pre-Op）

从*初始化*转变到 *Pre-Op* 的过程中，EAP 设备检查邮箱是否已正确初始化。在 *Pre-Op* 状态下，可以进行邮箱通讯，但不可进行过程数据通讯。

安全操作（Safe-Op）

从 *Pre-Op* 转变到 *Safe-Op* 的过程中，EAP 设备检查内部对象引用及更新：

- 基于周期的配置参数，
- 过程映像输入和输出变量的引用指针，以及
- 每个发布服务器/订阅服务器变量从 PLC 到其过程变量的映射。

在 *Safe-Op* 状态下，将进行邮箱通讯并发送发布服务器变量。尚未收到 EAP 报文。

操作 (Op)

从 *Safe-Op* 转变到 *Op* 的过程中，EAP 设备再次检查启动期间是否发生错误。

在 *Op* 状态下，如果需要，EAP 设备将接收传入的 EAP 报文，并将接收到的过程数据复制到输入变量中。进行邮箱通讯，发送发布服务器变量，并接收订阅服务器变量。

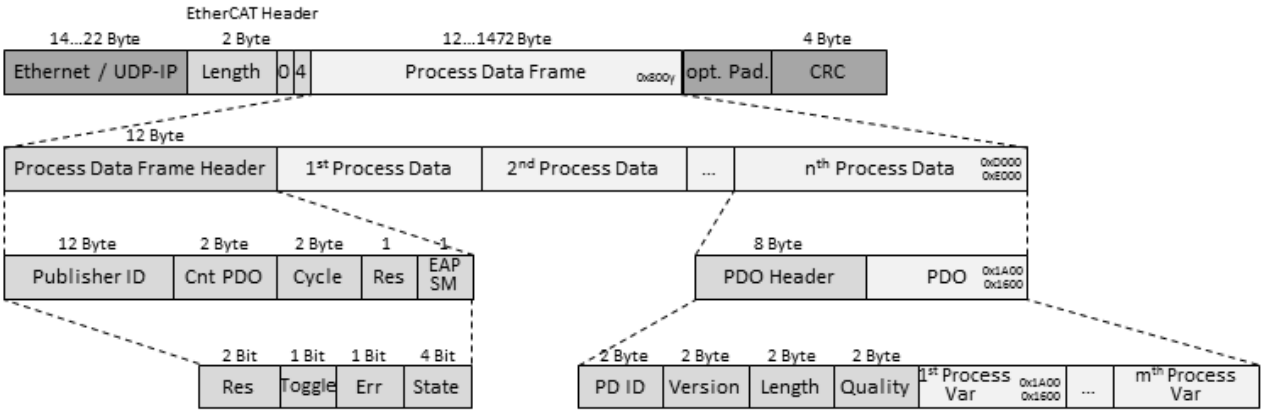
如果在上述状态转变过程中发生错误，则 EAP 设备保持在最后达到的状态，或重置为 Safe-Op 状态。同时，将设置错误位和相应的错误代码 (参见 TwinCAT EAP 设备文档中的章节 CANopen 对象字典)。例如，由于 CANopen 对象词典中的不一致而导致发生典型错误，从而使配置无效。

2.2 技术概念

2.2.1 EAP 报文结构

EAP 报文可以通过以太网 (Ethertype = 0x88A4，符合 EtherCAT 协议) 或 UDP/IP (UDP 端口 0x88A4) 进行传输。如果 EAP 基于 EtherCAT 协议，那么 EAP 专用报文部分将嵌入到 EtherCAT 协议的用户数据中。

EAP 报文由*过程数据帧标题*和一个或多个*过程数据 (PD)* 组成。*PD* 是通过 EAP 交换数据的主要单元。*PD* 由所谓的 *PDO 标题*和至少一个*过程变量 (PV) 组成*。总体而言，*PD* 的*过程变量*构成*过程数据对象 (PDO)*。参见下图。



过程数据帧标题

*过程数据帧标题*包括五个字段 (参见上图中的第 2 行和第 3 行)。后一个字段 (EAP SM) 扩展了 twincat 2 的 NWV 报文内容。该字段除其他信息外还包含有关 EAP 设备当前状态的信息。

发布服务器 ID
根据发布服务器 ID 识别报文发送器。它包含发送器的 AMS NetID。如果接收器配置为仅处理来自特定发送器的报文，则根据发布服务器 ID 字段检查 EAP 报文是否源自此发送器。

Cnt PDO
该字段包含报文中的*过程数据*编号，以便接收器可以完全处理报文。

周期
该字段的值随发送器的每个任务周期而递增。在接收端，该字段的内容可作为序列号，例如，用来检查报文是否丢失。应在轮询数据交换模式下，在服务器端监控该字段。

Res
为以后的扩展保留。

EAP SM
该字段包括 4 个子条目。

- **Res**：为以后的扩展保留。
- **切换**：为每个新 EAP 报文切换此位。
- **Err**：该字段指示 EAP 状态机是否发生错误。

- **状态：**该字段包含 EAP 状态机当前状态值。
  - 1：初始化
  - 2：操作前
  - 4：安全操作
  - 8：操作

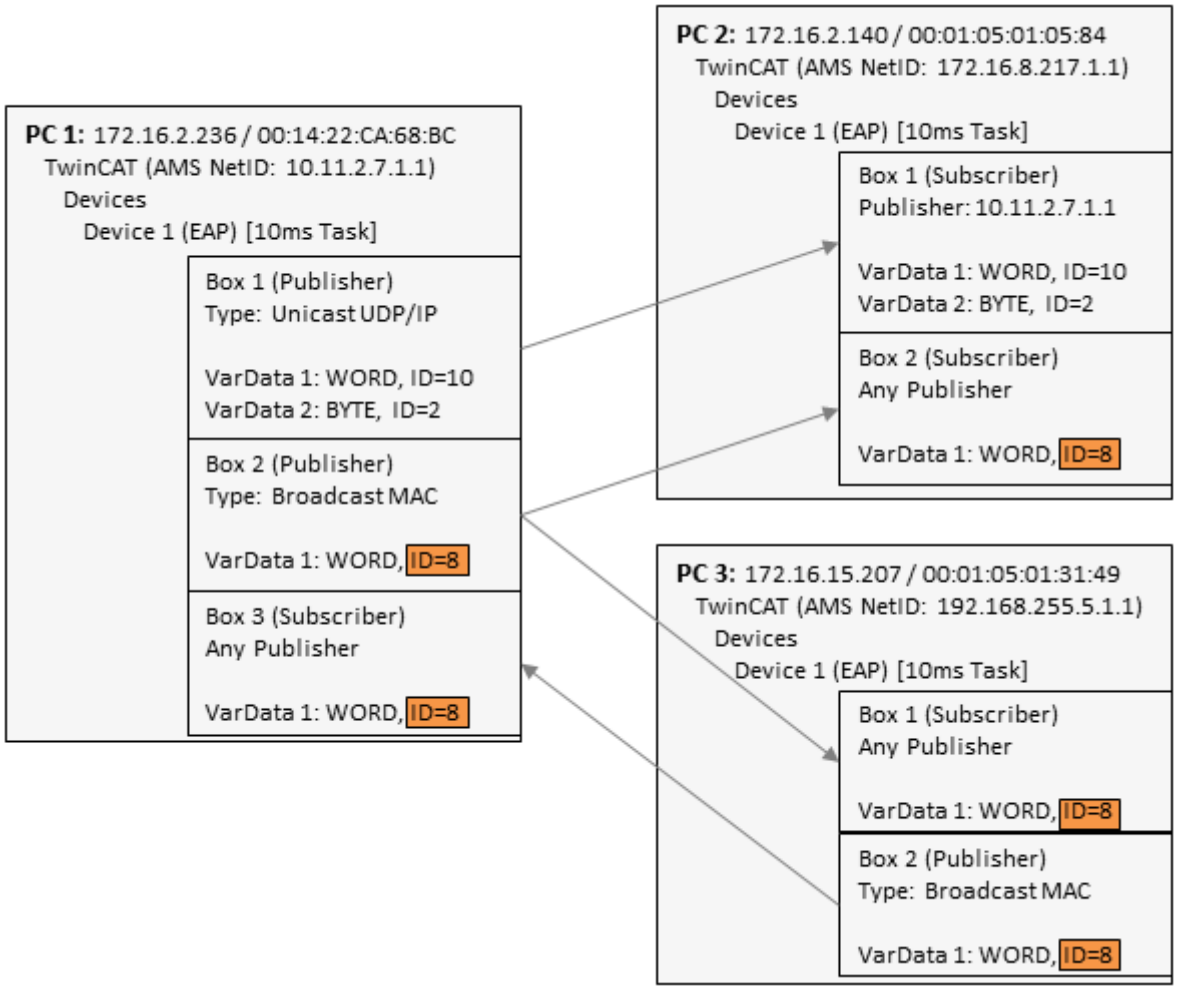
不允许使用其他值

PDO 标题

*PDO 标题*包含 4 个字段，每个字段的数据长度为 2 个字节（参见上图中的第 3 行和第 4 行）。

PD ID
PD ID（16 位）可作为 *EAP 变量*的全局标识，且在网络中具有唯一性。

> **选择过程数据标识符（PD ID）**
> 为了实现明确的分配，我们建议为连接的 PC 之间的每个数据通讯使用不同的 ID。
> **原因：**下图中，*框 2（订阅服务器）*中的 *PC 2* 不仅从 *PC 1* 接收 ID 8 作为*框 2（发布服务器）*的预期变量，而且由于它们是作为广播（！）发送，因此来自*框 2（发布服务器）*中的变量也来自 *PC 3*。之后将无法在 *PC 2* 中进行区分！



版本
可在此字段中为 *PD* 输入版本编号。*版本*对 EAP 通讯的影响在章节通讯方式 [▶ 9]的*网络协议*部分中进行了说明。如果对*过程数据*进行了至少一项更改，则应该一致增加版本编号。

- *过程数据*的数据长度已改变。
- *过程数据*变量的数据类型已改变。
- PDO 中的变量顺序已改变。

长度
该字段包含以字节为单位的*过程数据*长度。不包括*过程数据标题*本身的长度。

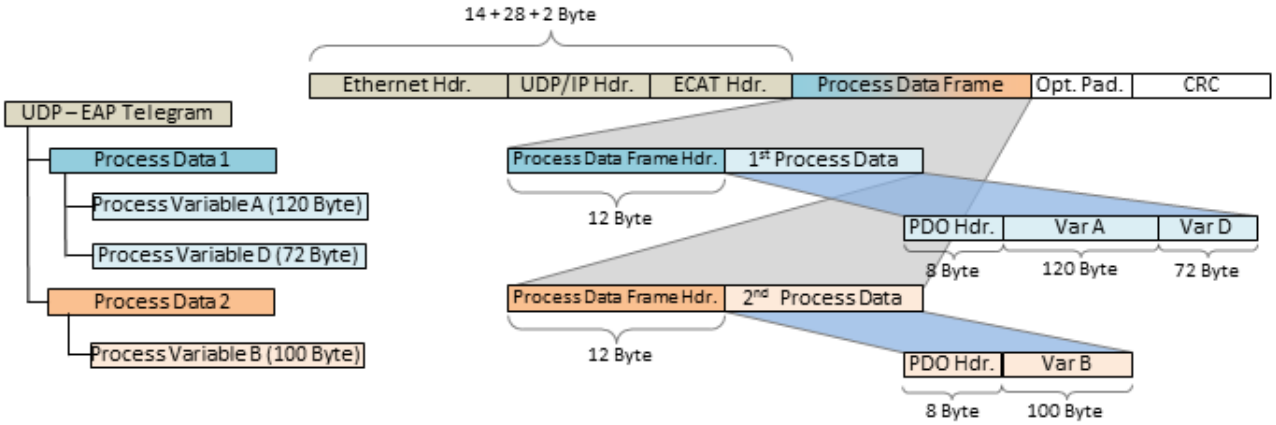
质量
如果值介于 0x0 和 0xEFFF 之间，则该字段的值以单位 [100 µs] 显示*过程数据*的时间。大于或等于 0xF000 的值表示*过程数据*无效。

过程变量 (PV)

*过程变量*包含要传输的实际数据 (参见上面第一个图中第 4 行的*过程 Var*)。只有 *PV* 的数据长度非常大时，整个 EAP 报文才不大于 1514 字节。

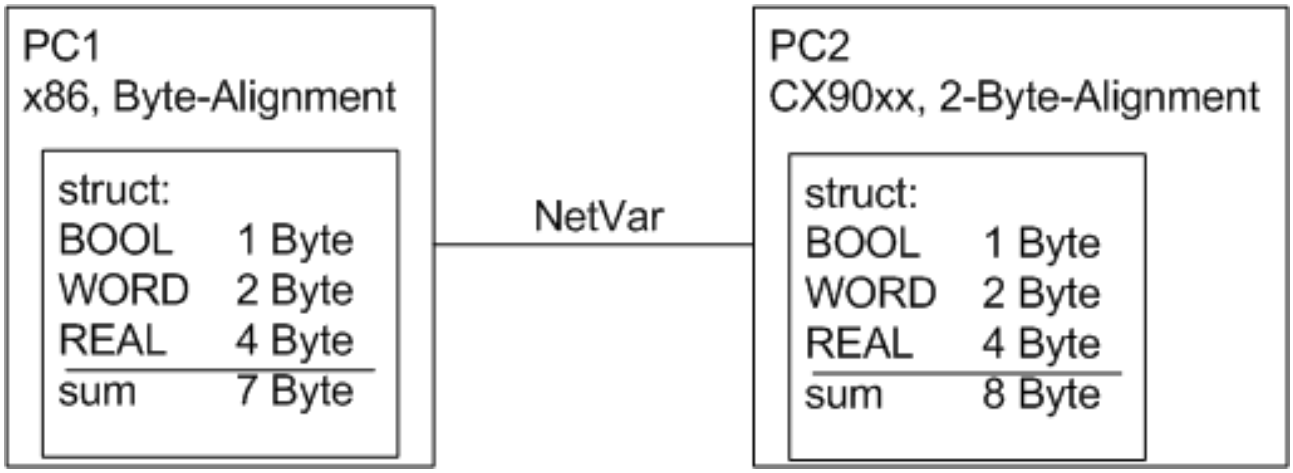
计算 EAP 报文大小

所有标题和所有发布服务器变量的长度总和不得超过 1514 字节的限值。所示 EAP 报文 (参见下图) 的总大小为
14 + 28 + 2 + 12 + 8 +120 + 72 + 12 + 8 + 100 = 376 字节。



不同平台上的数据表示形式

请注意，简单数据和复杂数据 (字、数组、实数、字符串和自定义结构) 在不同平台上均以不同方式在内部表示。x86 平台使用字节对齐，其他平台 (ARM) 使用 2 字节或 4 字节对齐。
这意味着，如果在 x86/PC PLC 项目和 ARM PLC 项目中都创建了复杂结构，它们可能有不同的有效尺寸和不同的内部结构。



在示例 (参见上图) 中，PC2 中的结构大于 PC1 中的结构。不仅如此，字和实数变量不匹配，因为变量可在 PC1 中的任意字节位置开始，但只能在 PC2 中的任意偶数字节位置开始。

建议在汇编结构时遵循以下规则，以避免对齐产生的尺寸差异：

- 首先，所有 4 字节变量 (必须位于可被 4 整除的地址)
- 然后，所有 2 字节变量 (必须位于可被 2 整除的地址)
- 然后是所有 1 字节变量

进一步建议：

- 如果在 PLC 中使用了数据类型字符串 (x)，则字符串的空终止也属于字符串本身，因此 x+1 必须被 4 整除。否则，将没有 4 字节对齐。
- 以上规则也适用于子结构。

其他信息参见 Beckhoff 信息系统，位于“TwinCAT 3/TExxxx | TC3 Engineering/PLC/PLC/编程参考/数据类型/自定义数据类型/结构”或“TwinCAT 3/TExxxx | TC3 Engineering/PLC/PLC/编程参考/声明/对齐”。

总线端子控制器 (BCxxxx、BXxxxx) 的使用

浮点数 (数据类型实数) 无法传输到总线端子控制器 (BCxxxx、BXxxxx)，因为总线端子控制器上的浮点数表示形式与 x86 平台不同。
例如，数据类型 SINT 可用于带符号的值。

3 EAP 连接诊断

TwinCAT EAP 设备提供不同的变量，借助这些变量可以诊断 EAP 通讯的质量或故障。可在发布服务器和订阅服务器中找到此类诊断变量。可以通过编程评估这些变量，以便应对可能的故障。

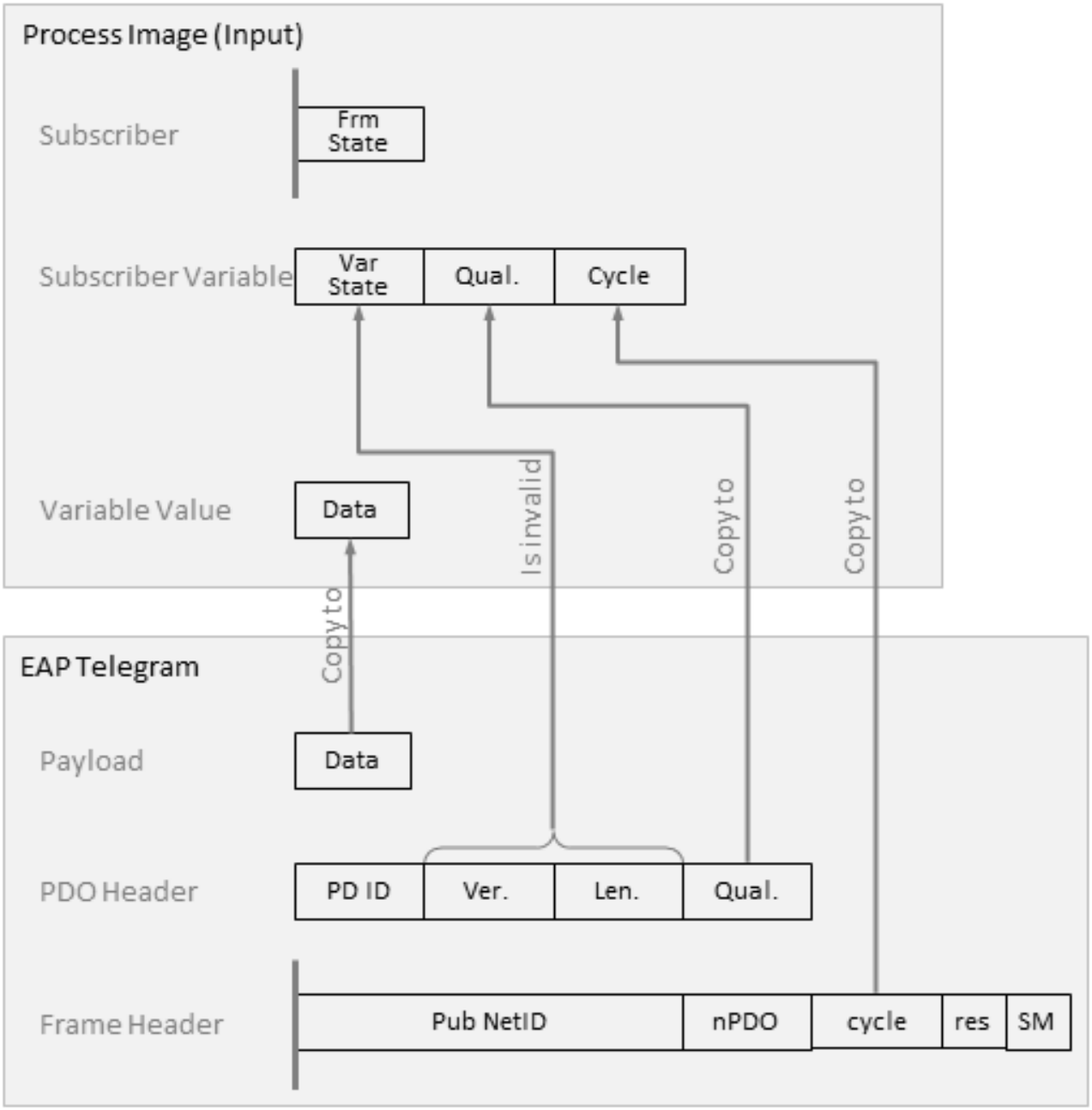
3.1 订阅服务器

对于订阅服务器，存在诊断变量*质量*、*CycleIndex* 和 *VarState*。

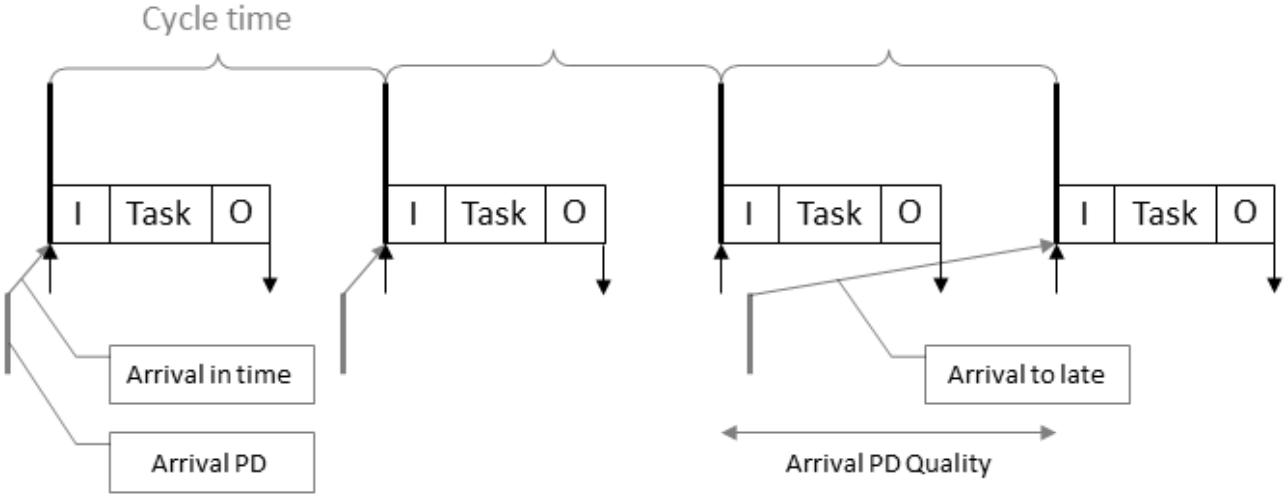
- *质量*:
  *质量*表示已接收数据的时间。
- *CycleIndex*:
  可借助 *CycleIndex* 检测传输故障。
- *VarState*:
  VarState 变量表示已配置的 *RxData* 和阻止接收数据的传入*过程数据*之间是否存在差异。

质量

*质量*变量包含以 [100 µs] 表示的时间长度，*过程数据*在该时间前到达接收器的时间过迟。更新 TwinCAT EAP 设备的输入过程映像前，这一时间长度在每个周期中即以周期递增。在处理已接收的 EAP 报文的过程中，更新输入过程映像。在此处理过程中，将每个报文的多个值分配给输入过程映像的相应变量。



如上图所示，例如，将从报文中接收的*质量*值分配给输入过程映像的相应*质量*变量。
如果 EAP 报文中*质量*值为 0，则在接收到报文后立即将输入过程映像像的*质量*重置为 0。



上图显示了 EAP 报文延迟到达如何影响*质量*值：开始时，在第一个周期开始前及时收到 EAP 报文。第二个周期也是一样。之后，直到第三个周期开始后，报文才到达接收器。其结果是，直到第四个周期才可以处理报文。因此，*质量*变量值会在第三个周期时增加一个*周期*，但不会重置为 0。*质量*变量只在第四个周期时通过分配延迟报文的*质量*值重置为 0。

诊断变量质量
假设订阅服务器的任务周期是发布服务器的十倍，则仅每十个周期接收一次 EAP 报文。因此，在九个周期内没有报文到达订阅服务器，这也意味着输入过程映像的*质量*变量无法在九个周期过程中重置。因此，*质量*变量将以九个周期的周期时间递增。最多增加九倍周期时间。

其结果是：“缓慢”发送器（例如：发布服务器中的 100 ms 传输时间）与“快速”接收器（例如订阅服务器中的 10 ms 接收时间）一起导致诊断变量*质量*的相应增加值。

因此，必须为 EAP 报文的传输和接收考虑不同的周期时间。在这方面，特别重要的是注意发送器中已配置的*触发*条件（参见 EAP 发送机制 [▶ 11]）。

EL6601/EL6614
使用 EL66xx 时，*过程数据*的到达时间恰好是该数据出现在 EAP 设备的输入过程映像中时，而不是数据到达 EL66xx 或 EtherCAT 设备的输入映像中时。

CycleIndex
CycleIndex（尺寸：16 位）是一个计数器，由发布服务器通过*过程数据*传输。在发送 EAP 报文之前，通常在发送器端随每个新周期而递增，从而可以推断出传输中断。它可以在接收器端（订阅服务器中）读取为 *CycleIndex*（参见本章最上方的插图）。

VarState
VarState（尺寸：16 位）提供有关 *RxData* 当前状态的信息。
VarState 可能有以下值：

简短描述	位	描述
无效散列/版本	VS.0	如果由于接收到的过程数据版本与该 RxProcessData 的配置版本不相符而无法接收到*过程数据*，则该位设置为 1。否则，该位设置为 0。
收到无效的变量长度	VS.1	如果由于接收到的变量数据长度与配置的 RxVariable 不相符而无法收到*过程数据*，则该位设置为 1。否则，该位设置为 0。

EL6601/EL6614
使用 EL66xx 终端时不会创建此 VarState。

3.2 发布服务器

诊断变量 *VarState* 和 *FrameState* 在发布服务器中可用。通过这些诊断变量，可以阐明 EAP 报文传输的详细属性。

VarState
VarState（尺寸：16 位）提供有关 *TxData* 当前状态的信息。
VarState 可能有以下值：

简短描述	位	描述
未发送（跳过变量）	VS.0	只要该值为 0，就会传输 TxProcessData。否则将暂停传输 TxProcessData。

EL6601/EL6614
使用 EL66xx 终端时不会创建此变量。

FrameState
FrameState（尺寸：16 位）提供有关 *TxFrame* 当前状态的信息。*FrameState* 可能有以下值：

简短描述	位	描述
未发送（跳过帧）	FS.0	只要该值为 0，就会传输以太网帧。否则将暂停传输帧。

简短描述	位	描述
错误（帧过大）	FS.1	如果该值为 1，则已超出以太网帧的最大尺寸。链接的变量应较小。
订阅服务器缺失（仅单播）	FS.2	仅在订阅服务器监控启用时设置此位（参见章节发布服务器框 [▶ 39]）。一旦订阅服务器监控机制检测到无法再到达以太网帧发送到的 EAP 设备，就会设置该值。恢复访问后，立即将该值重置为 0。

EL6601/EL6614

使用 EL66xx 终端时不会创建此变量。

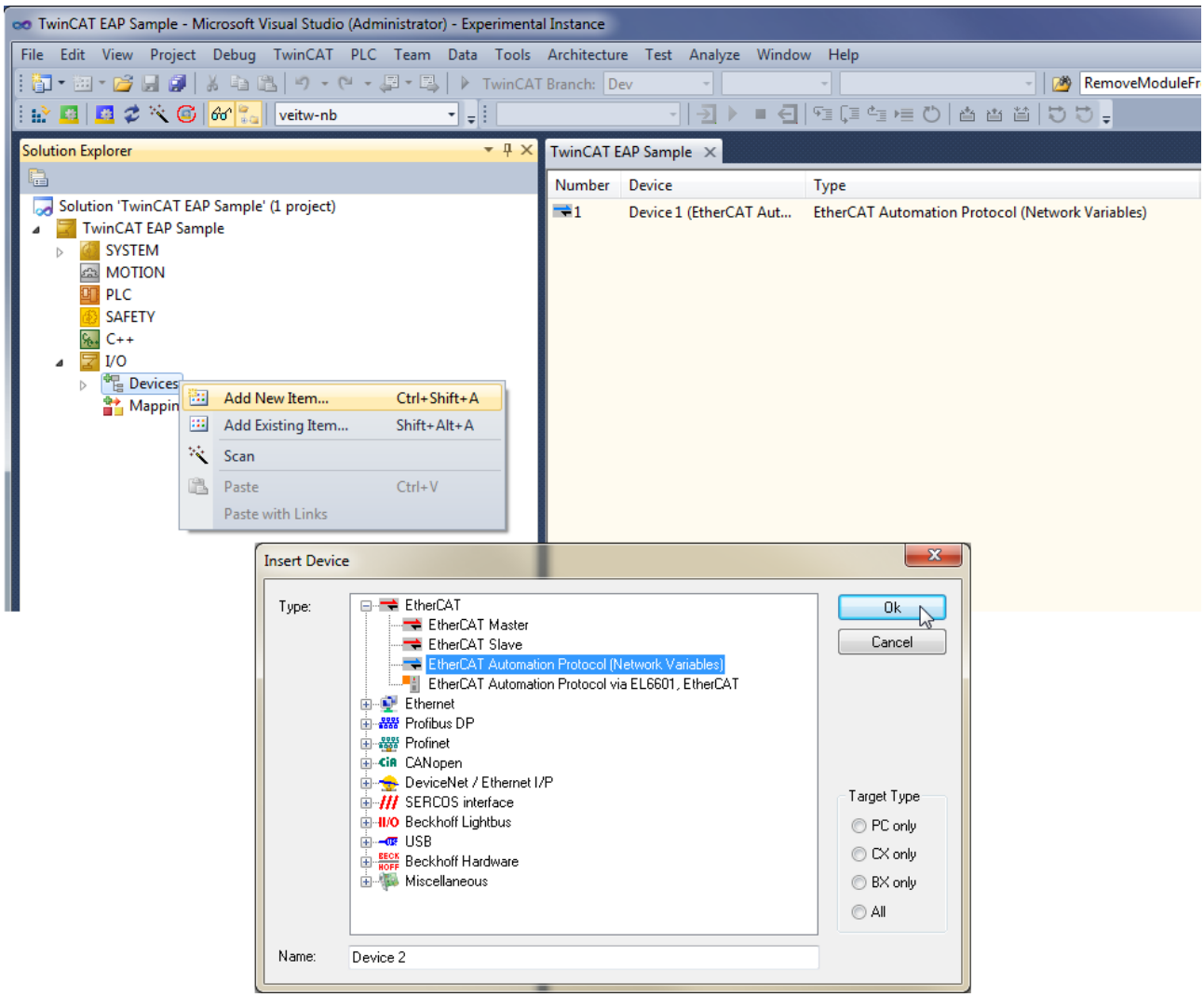
4 创建 EAP 配置

借助 TwinCAT 3，通过 EAP 在 TwinCAT 控制器之间创建通讯连接。

4.1 添加 EAP 设备

通过路径 [*I/O*] → [*Devices (设备)*] 将 EtherCAT 自动化协议设备添加到 TwinCAT 3 中（参见下图）。

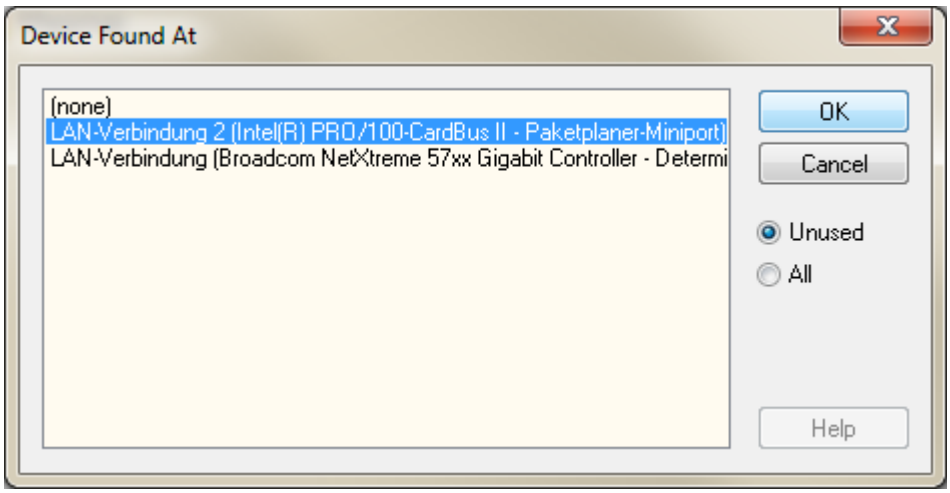
1. 在 [*Devices (设备)*] 节点的上下文菜单中，单击命令 [*Add New Item...(添加新项目...)*]
   ⇨ *Insert Device (插入设备)* 对话框打开。
2. 选择 EtherCAT 节点下的 *EtherCAT Automation Protocol (Network Variables)(EtherCAT 自动化协议(网络变量))*，并单击 [*OK (确定)*] 确认您的选择。



如果计算机有多个未使用的、支持实时处理的网络适配器，则会出现一个对话框，可以在其中选择网络适配器（参见下图）。

3. 选择所需适配器，然后单击 [*OK (确定)*] 确认。

⇨ 随后，可以在 *Devices (设备)* 节点下，通过名称 *Device 1 (EtherCAT Automation Protocol)(设备 1 (EtherCAT 自动化协议))* 看到 EAP 设备。



在下一步，可配置*发布服务器*或*订阅服务器变量*。

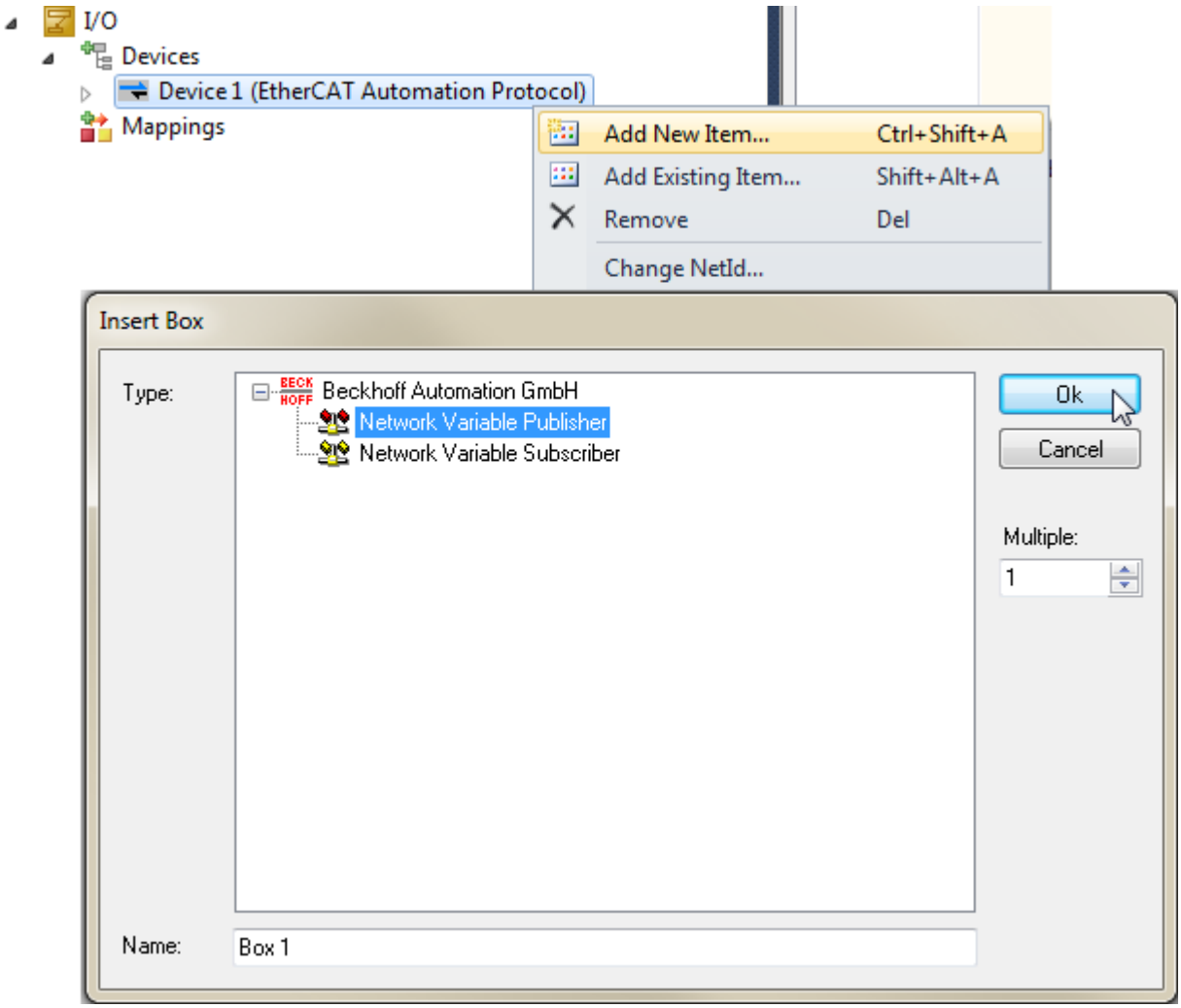
4.2 添加发布服务器变量

如果 EAP 设备要发送变量，则必须添加*发布服务器变量*，以完成其配置（参见下图）。

1. 在 EAP 设备节点（*Device 1（设备 1）*）的上下文菜单中，单击条目［*Add New Item...*(添加新项目...)］。
   ⇨ *Insert Box（插入框）*对话框打开。
2. 选择 *Network Variable Publisher（网络变量发布服务器）*并单击［*OK（确定）*］。

添加多个发布服务器

如果要创建多个*发布服务器*，可以使用 *Multiple（多个）*输入字段设置数量。



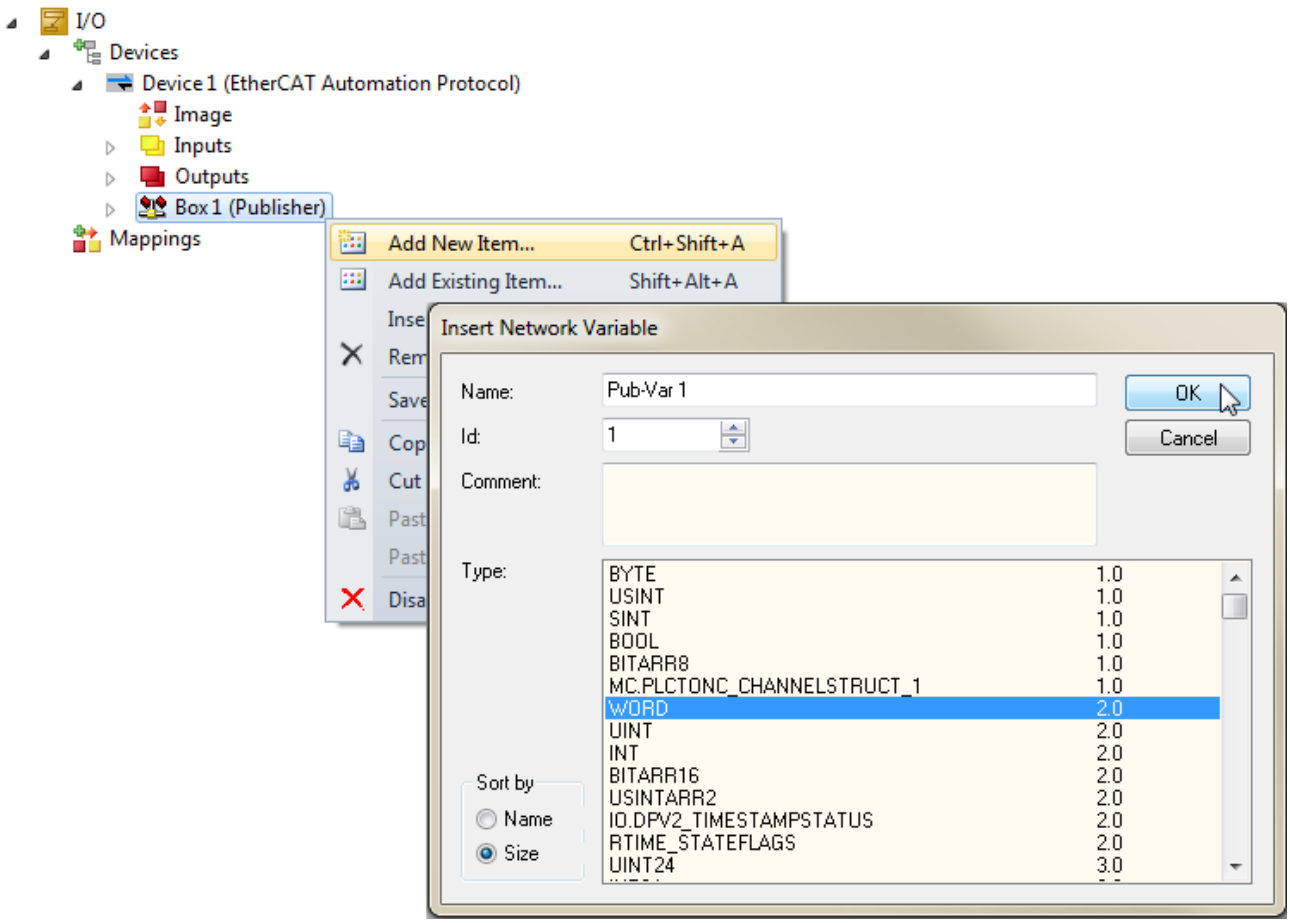
随后，将*发布服务器变量*（*TxData*）附加到创建的*发布服务器*中（参见下图）。

3. 在*发布服务器*的上下文菜单中，单击菜单项［*Add new Item...*（*添加新项目...*）］。
   ⇨ 出现对话框 *Insert Network Variable*（*插入网络变量*）。
4. 选择*发布服务器变量*的数据类型，在 *Name*（*名称*）中输入名称，定义 *ID*，然后单击［*OK*（*确定*）］。

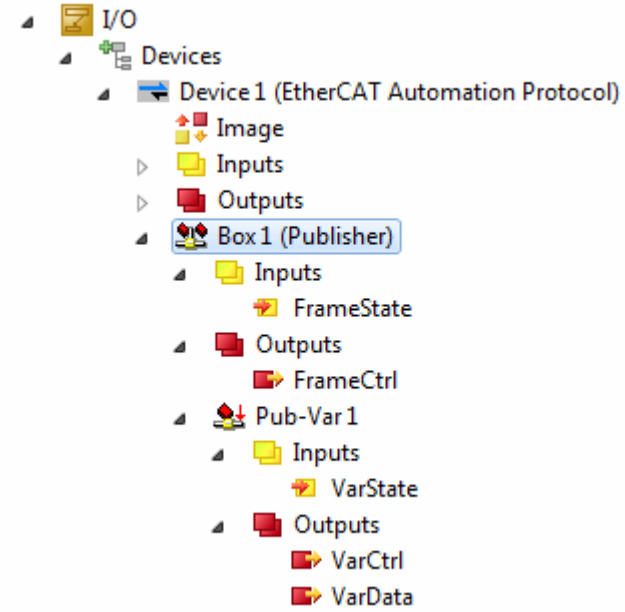
⇨ 此*发布服务器变量*的节点出现在*发布服务器*下（参见下图）。

分配 ID

为*发布服务器变量*选择的 *ID* 在网络中应该是唯一的。



在配置发布服务器变量的过程中，生成输入和输出变量，它们属于 EAP 设备的各个节点和子节点（参见下图）。除了无条件输入/输出变量外，还有一些仅在进行特定配置设置后才会生成的条件变量。



所有输入/输出变量最终共同构成 EAP 设备的过程映像。如果它们与 TwinCAT 中的应用程序（例如 PLC 程序）变量链接在一起，则后者可以从过程映像中读取这些内容或将值写入过程映像。输出变量的作用是写入例如控制 EAP 设备行为的值。输入变量的作用是读取它们的值，以便例如评估 PLC 程序的状态信息并对此作出反应。通过这种方式，EAP 设备的行为会直接受到 PLC 的影响。

输入和输出变量的外观

过程映像的结构会有所不同，具体取决于 EAP 设备用于通讯的端口（PLC 的网络适配器或 EL66xx 交换机端口终端）。

无条件输入/输出变量的描述

FrameState
*发布服务器*节点下的 *FrameState* 输入变量指示 *TxFrame* 的当前状态。详细描述请参见发布服务器 [▶ 21] 部分。

FrameCtrl
*发布服务器*下的 FrameCtrl 输出变量可用于控制 EAP 报文传输。*FrameCtrl* 可能有以下值：

简短描述	位	描述
禁用发送	FC.0	如果该位设置为 1，将中断帧的传输。仅当该值返回 0 时，帧的传输才重新开始。
从 ARP 缓存中删除目标 MAC	FC.1	仅当 EAP 报文通过 IP 或 *AMSNetID* 发送时，此控制框才有效。如果设置了该位，则删除 ARP 缓存中输入的目标 *MAC* 地址。如果 EAP 设备可以通过 ARP 确定目标 *MAC* 地址，则只有在该位重置为 0 时才能创建新条目。(另请参见通过 ARP 进行远程站监控 [▶ 11])

输出变量 FrameCtrl 的外观

通过 EL66xx 终端进行 EAP 通讯时，此变量不存在！相反，输出变量 *CycleIdx* (尺寸：16 bits) 被创建，该变量用于诊断通讯连接 (参见EAP 连接诊断 [▶ 19])。

VarState
*发布服务器变量*下的输入变量 *VarState* 指示 *TxData* 的当前状态。详细描述请参见发布服务器 [▶ 21]部分。

VarCtrl
*发布服务器变量*下的输出变量 *VarCtrl* 可用于控制*发布服务器变量*的传输。*VarCtrl* 可能有以下值：

简短描述	位	描述
禁用发布	VC.0	*发布服务器变量*传输中断。仅当该位的值返回到 0 时，才重新启动*发布服务器变量*的传输。

输出变量 VarCtrl 的外观

通过 EL66xx 终端进行 EAP 通讯时，此变量不存在！

CycleIndex
TxProcessData 下的输出变量 *CycleIndex* 应与周期性递增的应用变量链接，因为可以评估接收端的 *CycleIndex* 以用于诊断目的 (参见EAP 连接诊断 [▶ 19])。

输出变量 CycleIndex 的外观

只有通过 EL66xx 终端进行 EAP 通讯，输出变量 CycleIndex 才会存在！
如果通过 PC 网络适配器进行 EAP 通讯，则此变量不存在。然后，EAP 报文中的 *CycleIndex* 随每个任务周期自动递增。

VarData
*发布服务器变量*的输出变量 *VarData* 可与适合数据类型的任何变量链接 (例如与 PLC 程序的变量链接)。

条件输出变量的描述

MAC / NetID / IP
只有激活 *Target Address Online Changeable (可在线更改目标地址)* 选项时，*发布服务器框*节点下的输出变量 *MAC / NetID / IP* 才存在。在这种情况下，可以动态更改 EAP 报文的目标地址 (例如借助 PLC 程序)。
根据为 *TxFrame* 配置的寻址类型 (MAC 地址、AMS NetID 或 IP)，变量的数据类型为 MAC、NetID 或 IP。

VarId
仅当激活变量 *ID* 的 *Online Changeable (可在线更改)* 选项时，*发布服务器变量*下的输出变量 *VarId* 才存在。在这种情况下，可以动态更改 TxData 的 *ID* (例如借助 PLC 程序)。

4.3 添加订阅服务器变量

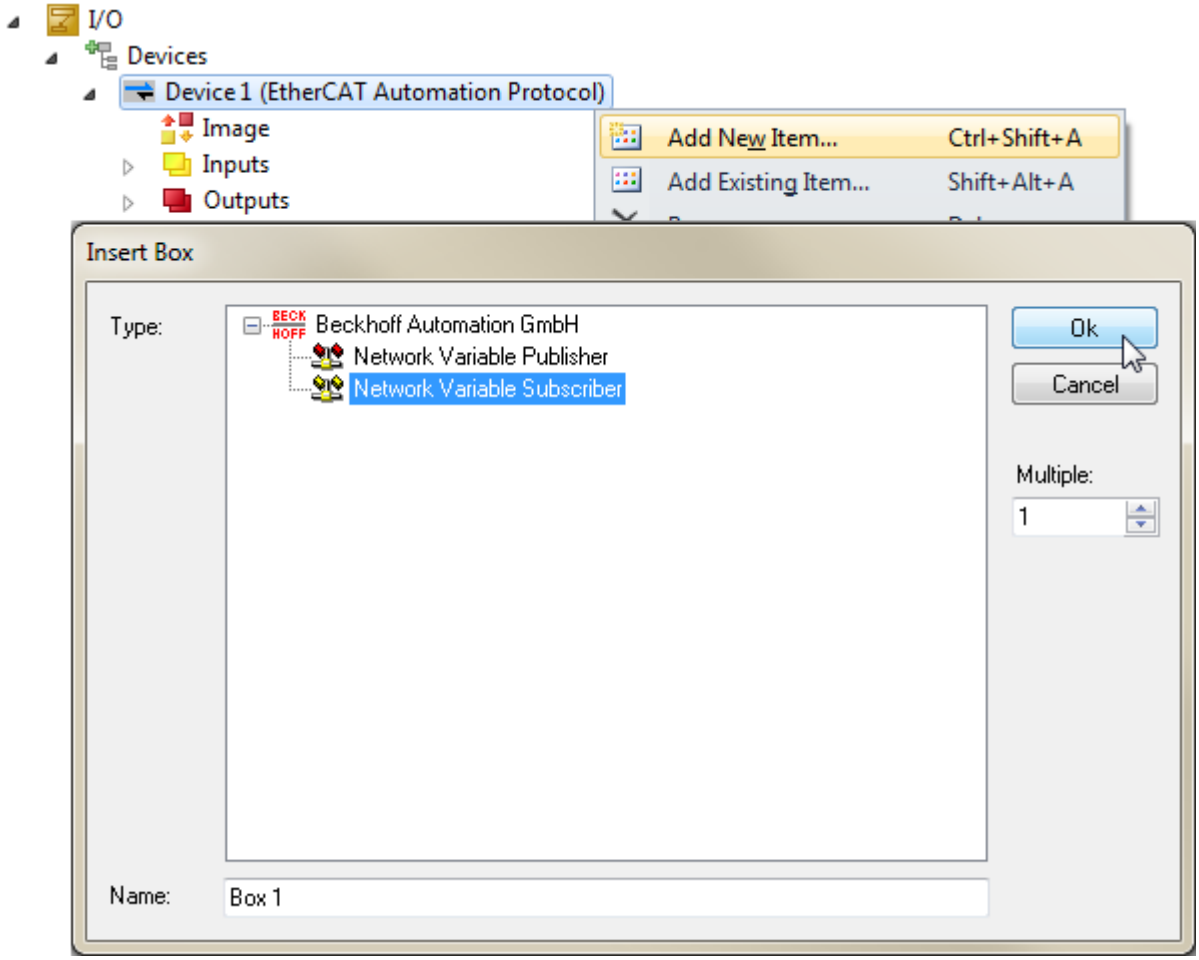
如果 EAP 设备要接收变量，则需要添加订阅服务器变量，以完成其配置（参见下图）。

1. 在 EAP 设备节点（*Device 1（设备 1）*）的上下文菜单中，
   单击条目［*Add New Item...*（添加新项目...）］。
   ⇨ *Insert Box（插入框）*对话框打开。
2. 选择 *Network Variable Subscriber（网络变量订阅服务器）*并单击［*OK（确定）*］。

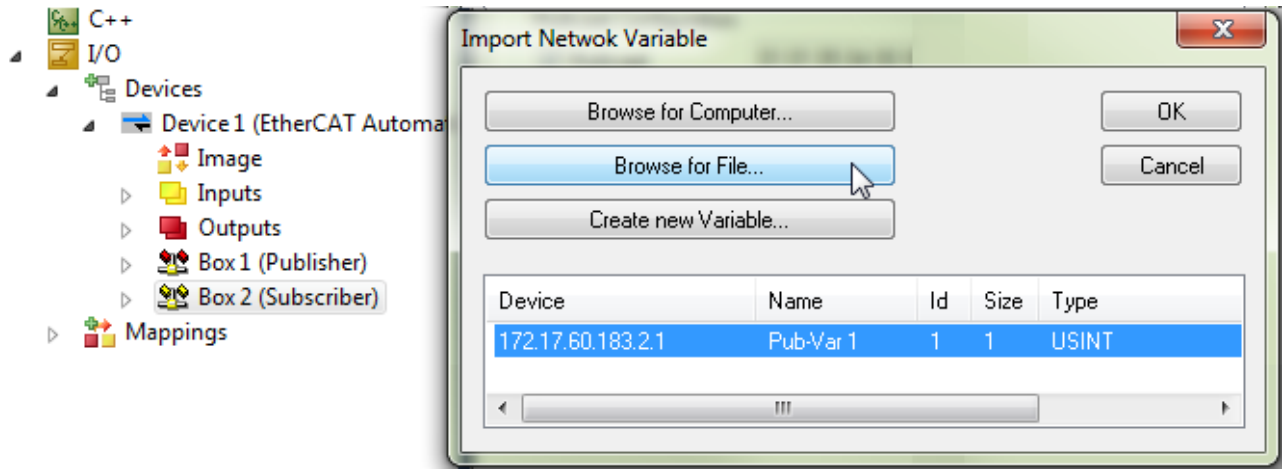
添加多个订阅服务器

如果要创建多个*订阅服务器*，可以使用 *Multiple（多个）*输入字段设置数量。

最后，将*订阅服务器变量*（*RxData*）附加到生成的*订阅服务器*（参见下图）。



3. 在*订阅服务器*的上下文菜单中，单击菜单项［*Add new Item...（添加新项目...）*］。
   ⇨ 将打开 *Import Network Variable（导入网络变量）对话框*，可借助其导入或定义*订阅服务器变量*。
   有三种选择。



Browse for Computer（浏览计算机）
可以自动建立与网络中另一个控制器的*发布服务器变量*的连接。

✓ 另一个控制器必须位于已知目标系统的列表中。

4. 单击［*Browse for Computer…(浏览计算机...)*］。
   ⇨ *Choose Target System（选择目标系统）*对话框打开。
5. 从列表中选择所需的控制计算机，然后单击［*OK（确定）*］。
   ⇨ 将列出此计算机提供的所有*发布服务器变量*。
6. 选择所需的*发布服务器变量*，并点击［*OK（确定）*］确认。

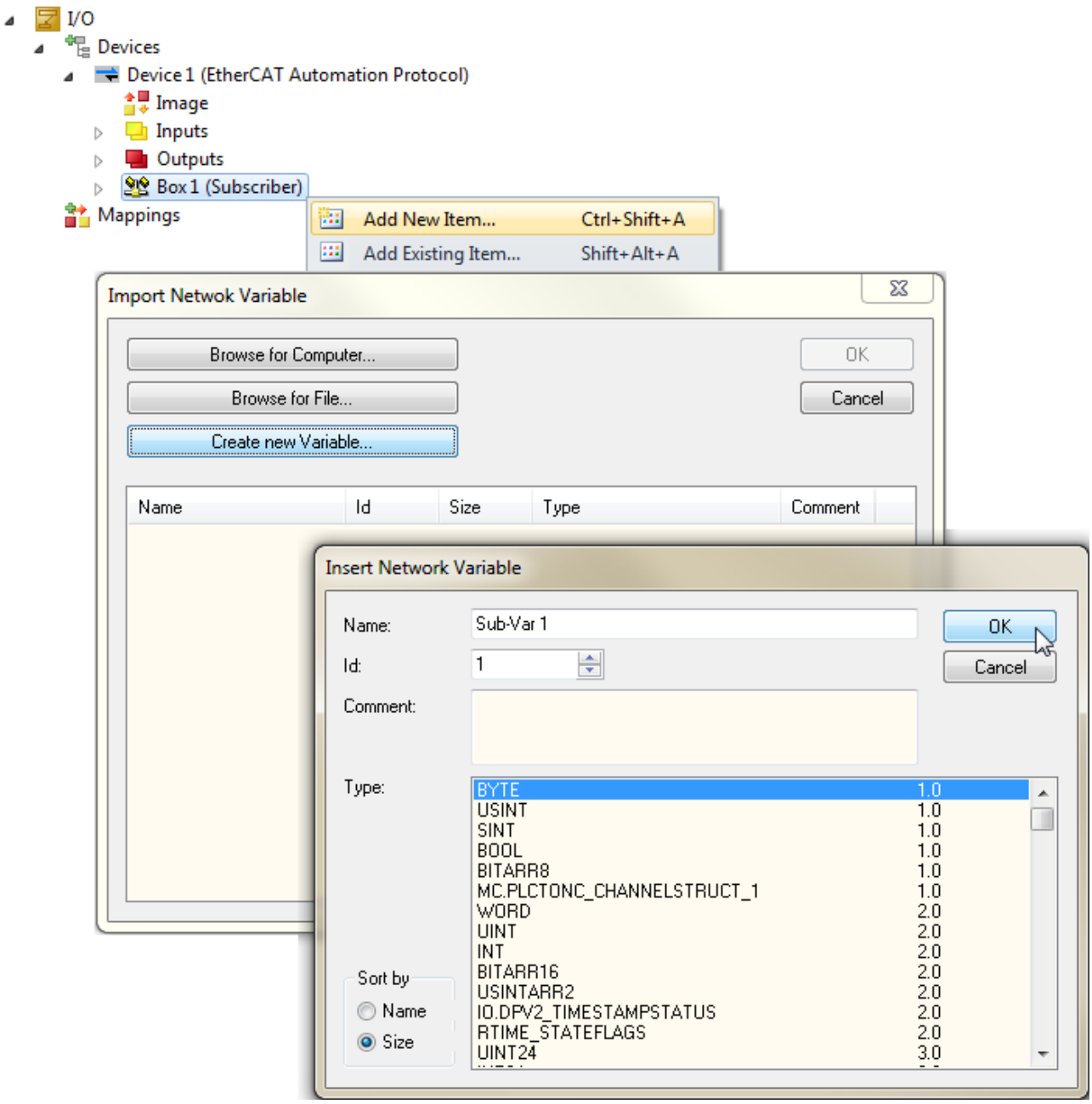
⇨ 创建*订阅服务器变量*以满足所选的*发布服务器变量*。

Browse for File（浏览文件）
作为 *Browse for Computer（浏览计算机）*的另一种选择，可以通过浏览（将要）在控制计算机上激活的项目文件来自动建立与另一台控制计算机变量的连接。

1. 单击［*Browse for File…(浏览文件...)*］。
   ⇨ 将打开用于浏览文件系统的对话框。
2. 浏览文件系统以找到所需的项目文件，然后单击［*OK（确定）*］确认选择。
   ⇨ 将列出此计算机提供的所有*发布服务器变量*。
3. 选择所需的*发布服务器变量*，并点击［*OK（确定）*］确认。

⇨ 创建*订阅服务器变量*以满足所选的*发布服务器变量*。

Create new Variable（创建新变量）
最后一个选项是手动配置*订阅服务器变量*（参见下图）。

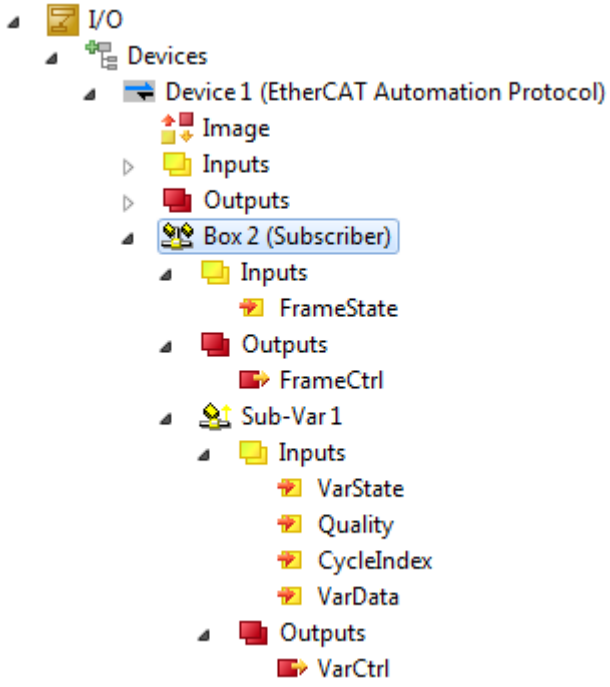


1. 单击 [*Create new Variable (创建新变量)*]。
   ⇨ 出现对话框 *Insert Network Variable (插入网络变量)*。
2. 选择*订阅服务器变量*的数据类型，在 *Name (名称)* 中输入名称，定义 *ID*，然后单击 [*OK (确定)*]。

⇨ 此*订阅服务器变量*的节点出现在*订阅服务器* 下方（参见下图）。

分配 ID

所选的*订阅服务器变量 ID* 必须与要接收的*发布服务器变量 ID* 相同。



无条件输入和输出变量的描述

FrameState/FrameCtrl
已保留*发布服务器*节点下的输入变量 *FrameState* 和输出变量 *FrameCtrl*，并且目前未使用。

VarState
*订阅服务器变量*下的输入变量 *VarState* 指示了 *RxData* 的当前状态。详细描述请参见订阅服务器 [▶ 19]部分。

VarCtrl
*订阅服务器变量*下的输出变量 *VarCtrl* 可用于控制接收。*VarCtrl* 可能有以下值：

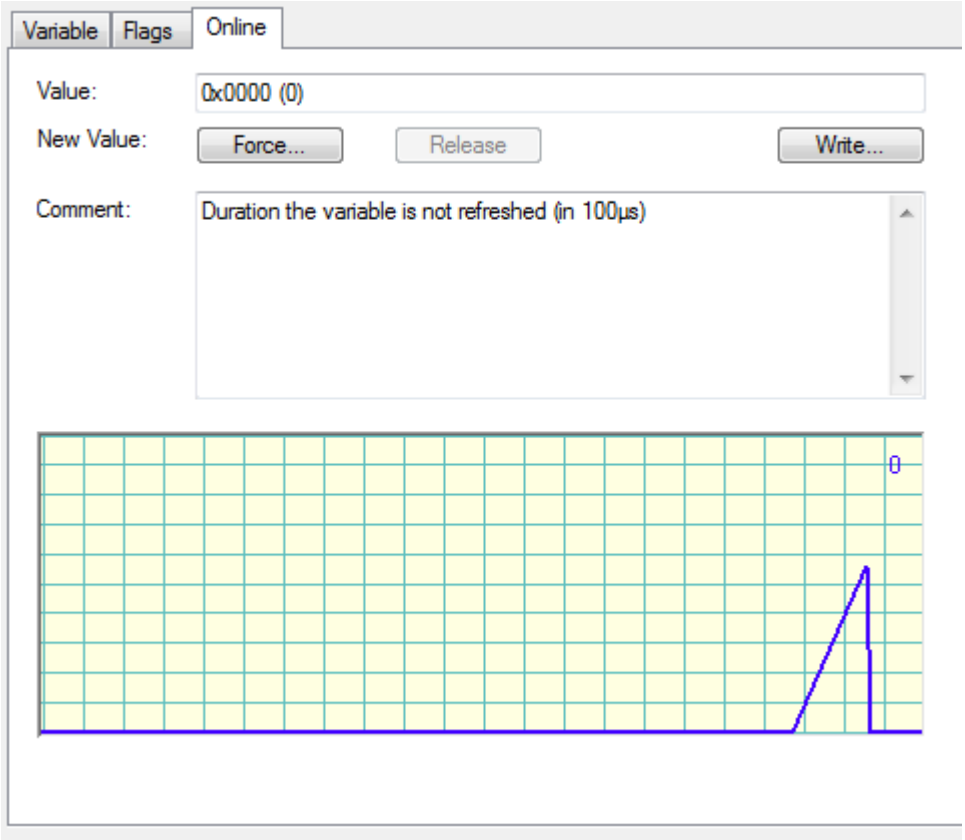
简短描述	位	描述
忽略散列/版本	VC.0	如果该位设为值 1，则在接收*过程数据*时取消版本检查。

输出变量 VarCtrl 的外观
通过 EL66xx 终端进行 EAP 通讯时，此变量不存在！

VarData
*订阅服务器变量*的输入变量 *VarData* 可以与 TwinCAT 中适合数据类型的任何所需变量链接（例如与 PLC 程序的变量链接）。

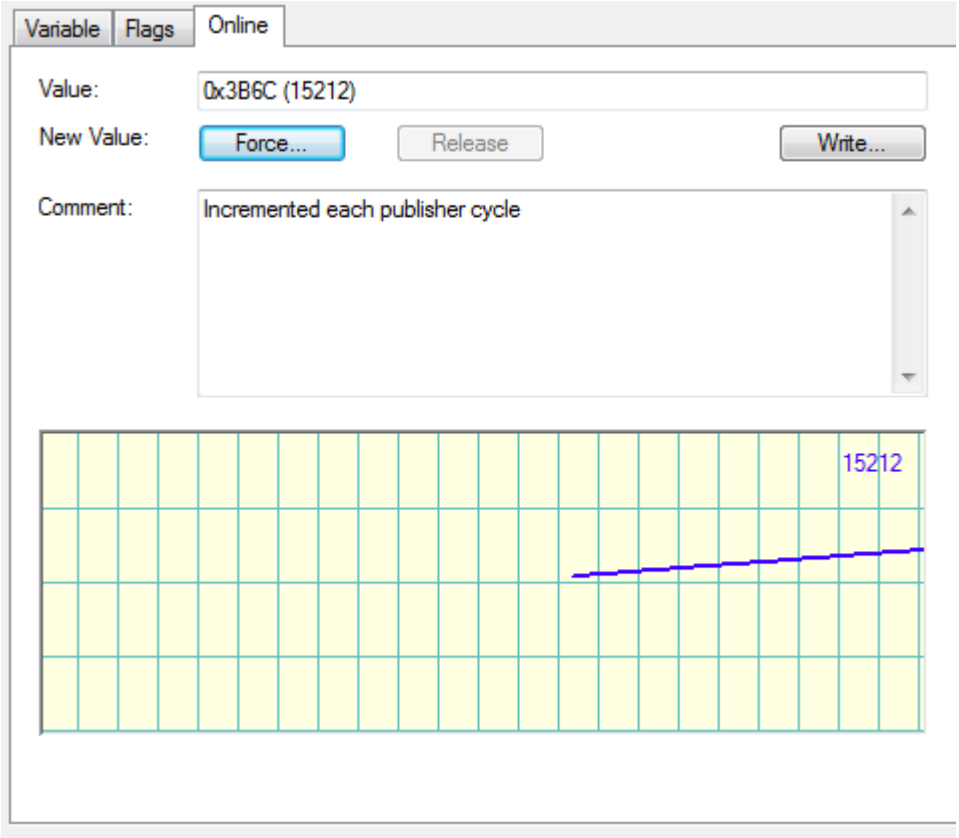
在接收端诊断变量的接收。为此可使用*订阅服务器变量*下的两个输入变量*质量*和 *CycleIndex*。

质量
*质量*变量提供了分辨率为 100 µs 的计数器。计数器值指示自该*订阅服务器变量*接收最后数据后的时间长度。下图中的示例显示了断开网络插口（计数增加）并重新连接（计数值 0）后的*质量*变量在线值



CycleIndex

CycleIndex 变量在每个*发布服务器*周期中递增。下图中的示例显示了查看 *CycleIndex* 变量在线值时的典型步骤。



条件输出变量的描述

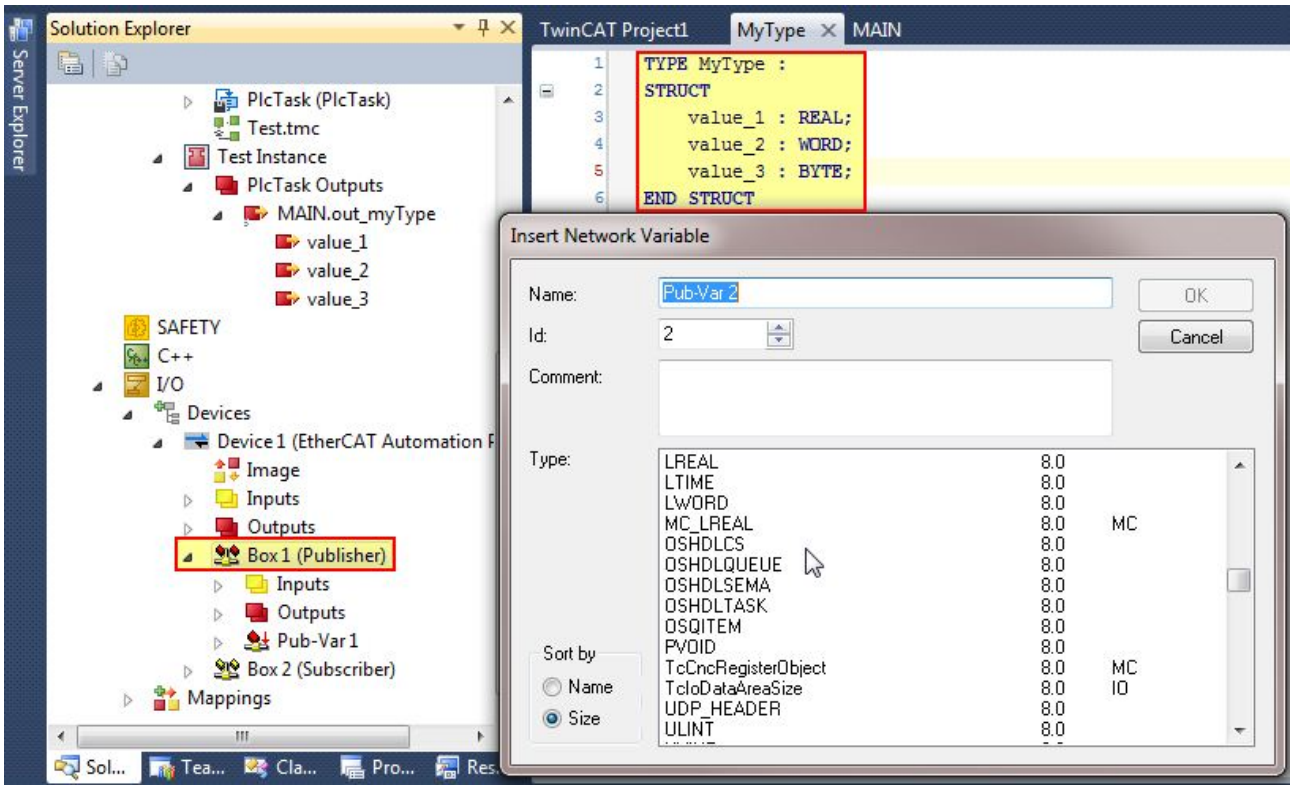
VarId
仅当激活*变量 ID* 的 *Online Changeable（可在线更改）*选项时，*订阅服务器变量*下的输出变量 VarId 才存在。在这种情况下，可以动态更改*订阅服务器变量 ID*（例如借助 PLC 程序）。

4.4 使用自定义数据类型

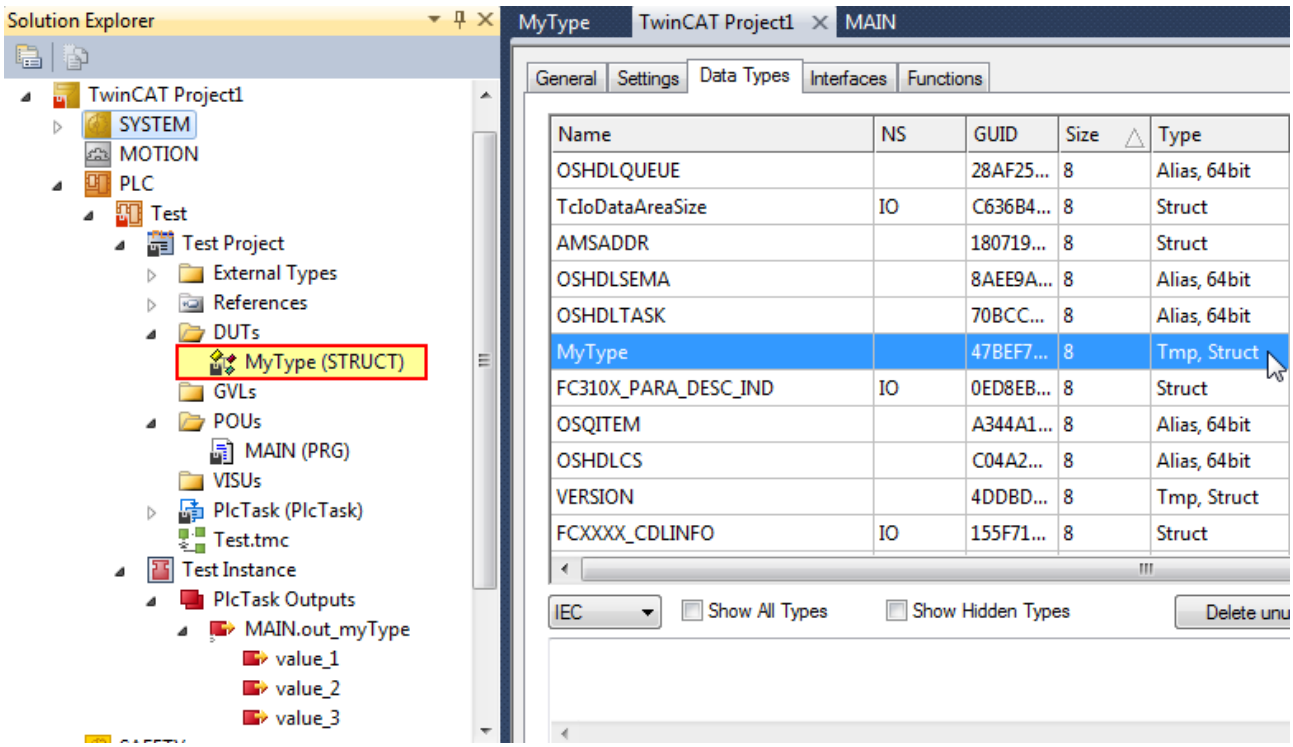
创建 EAP 变量

TwinCAT 中有两种常见方法来创建自定义数据类型。一方面，可以通过*数据类型*选项卡上项目树图的 *System（系统）*节点创建自己的数据类型（参见下一个插图）。然后，此数据类型可用于 TwinCAT 项目的所有模块。因此它是一个全局数据类型。
另一方面，用户通常通过定义 DUT（数据单元类型）在 PLC 项目中创建自己的数据类型。此数据类型最初仅可本地用于 PLC 项目。该数据类型对其他模块，例如 I/O 配置（因此也包括 EAP 设备）隐藏。

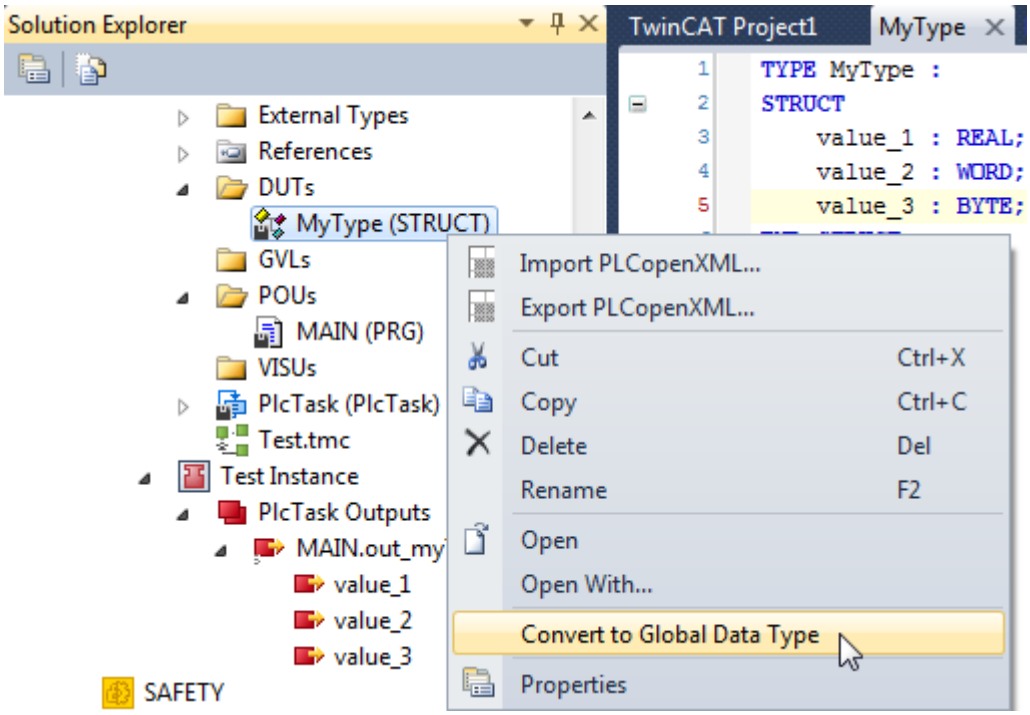
下图中，您可以看到在 PLC 中定义了数据类型 *MyType（我的类型）*。此数据类型还用于 PLC 程序的输出变量。但是，如果要在 EAP 设备的*发布服务器框*中创建此数据类型变量，则自定义数据类型不会出现在可用数据类型列表中。



可通过 *System（系统）*节点，可基于 *Data Types（数据类型）*选项卡上的数据类型列表找到数据类型仅用于本地的相应注释。在下图中，名称为 *MyType（我的类型）*的数据类型出现在列表中。但是，在 *Type（类型）*属性下有一条注解，即相关数据类型是临时数据 *（Tmp）*。这意味着相关数据类型是非全局的。



如果必须在全局上下文中使用来自 PLC 的本地数据类型，则可以将相应的 *DUT* 转换为全局数据类型（参见下图）。



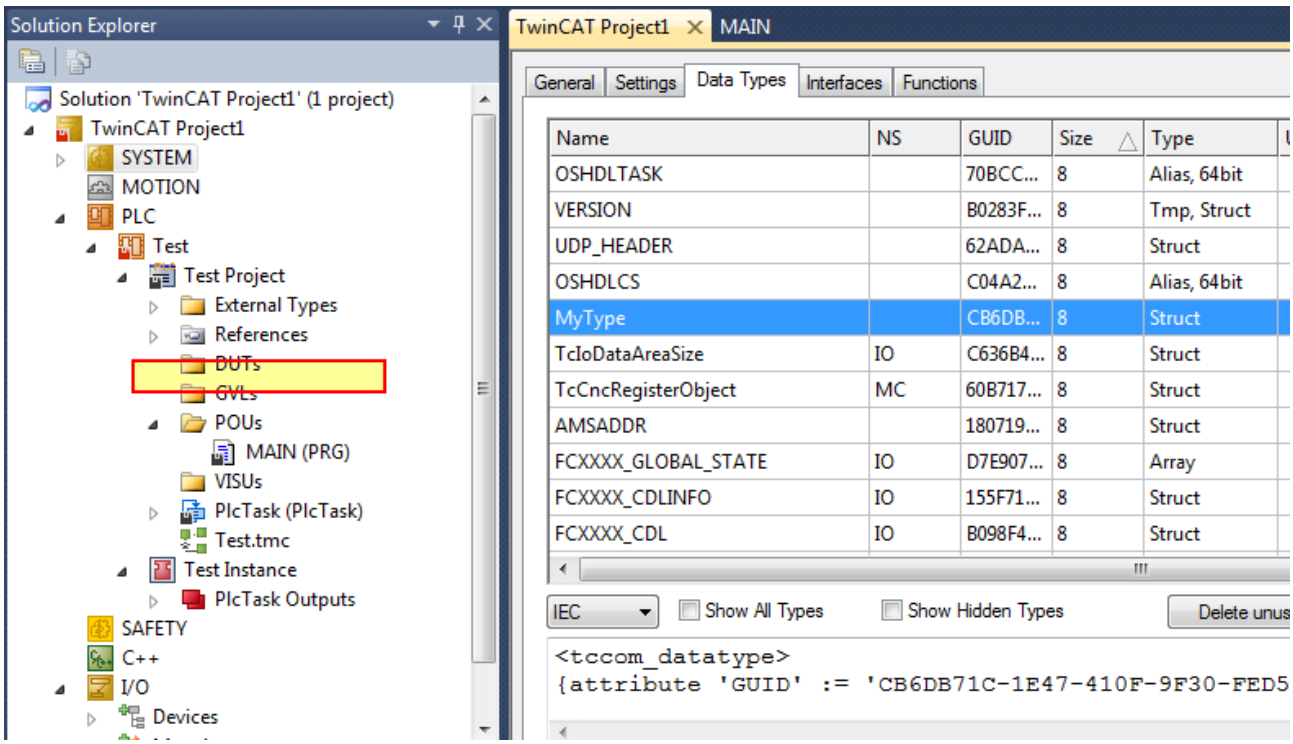
1. 在 *DUT* 的上下文菜单中单击命令 [*Convert to Global Data Type (转换为全局数据类型)*]。

⇨ 将自动从 PLC 项目中删除 *DUT* 节点，并创建该类型的全局声明（参见下图）。

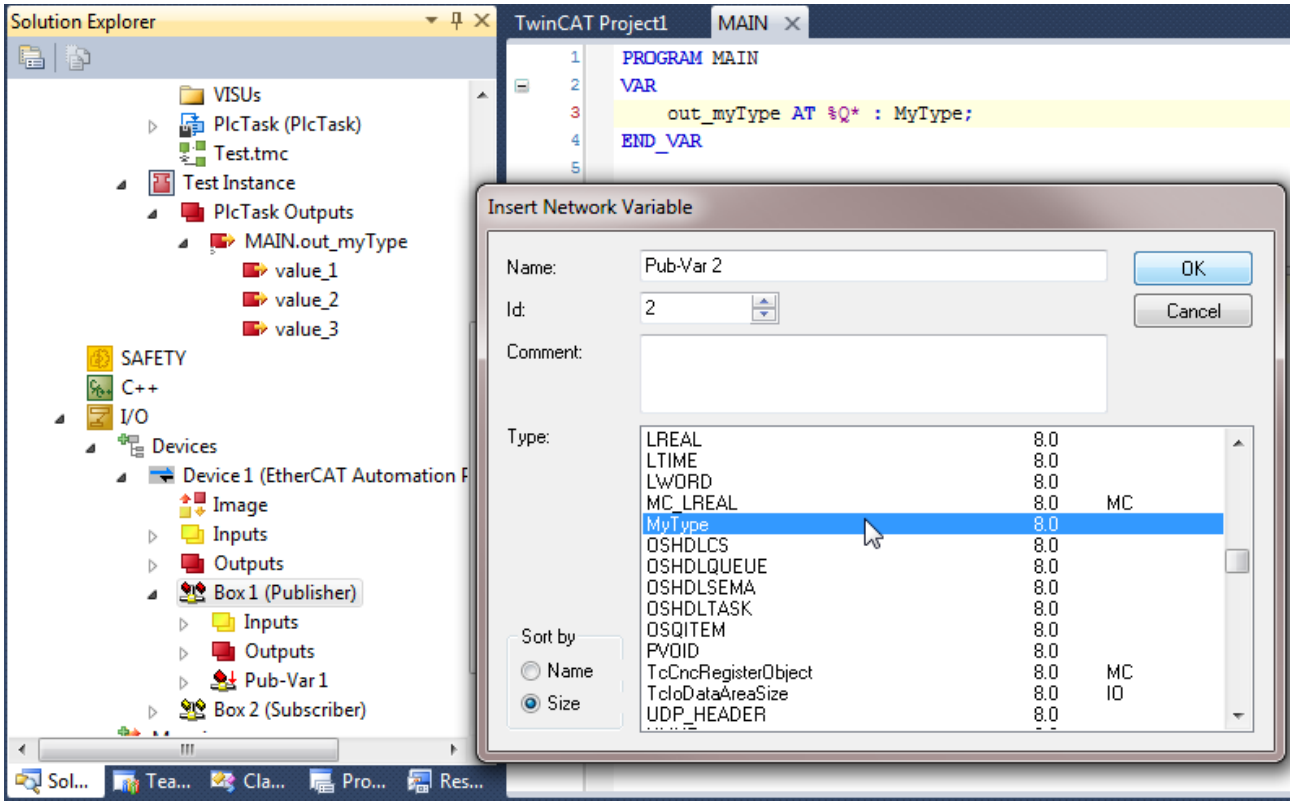
在 TwinCAT 中，同一种数据类型的描述必须是唯一的。因此，从 PLC 中删除了数据类型（原始 DUT）的定义。或者，可以在转换后在基于 XML 的 TwinCAT 项目文件中找到数据类型的定义，通过这种方式，整个 TwinCAT 项目就可以全局使用该数据类型。

数据类型作为本地和全局变量出现

将数据类型转换为全局数据类型后，后者在数据类型列表中出现两次：作为本地数据类型或全局数据类型。仅当不再引用本地数据类型时，才从列表中删除。转换后，通常需要再次编译 PLC 项目。从而删除从 PLC 程序到本地数据类型的引用，仅引用全局类型。因此，数据类型应仅在列表中出现一次，如下图所示。

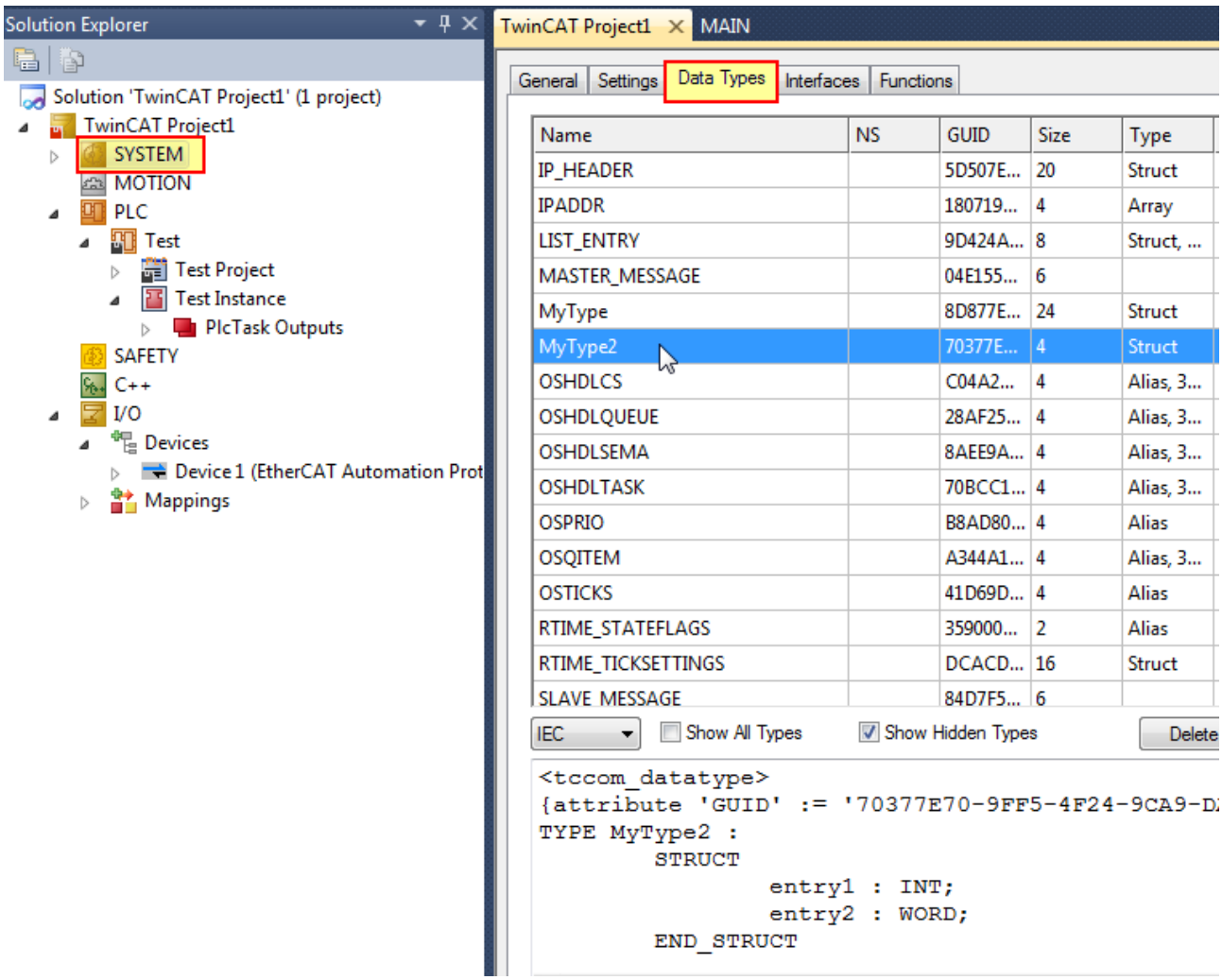


一旦将自定义数据类型转换为全局数据类型，即可在 EAP 设备中创建*发布服务器*或*订阅服务器变量*的过程中从可用数据类型列表中进行选择（参见下图）。

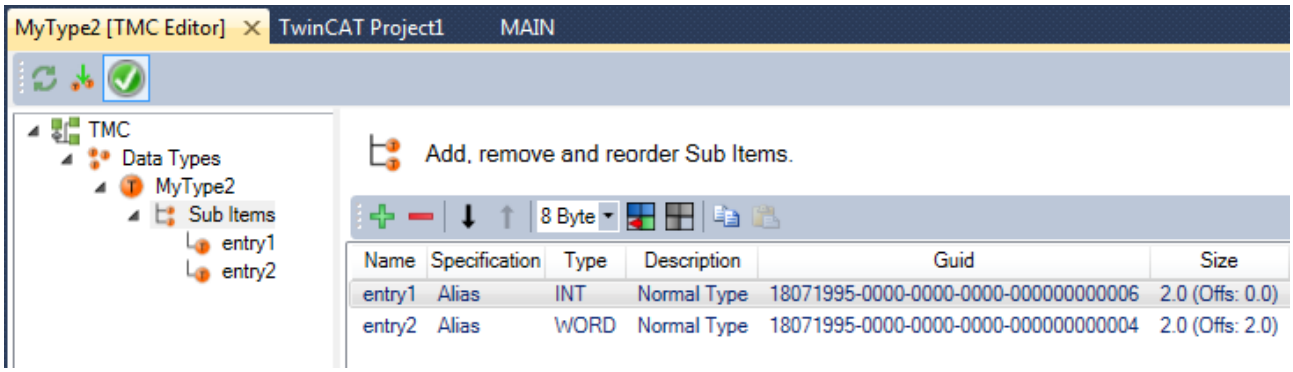


更改全局数据类型

一旦将数据类型转换为全局数据类型，其定义将不再受 PLC 项目的控制。数据类型的定义位于基于 XML 的 TwinCAT 项目文件中。现在必须通过 *System（系统）*节点的 *Data Types（数据类型）*选项卡，在 TwinCAT 项目中更改此数据类型（参见下图）。



1. 从列表中选择要更改的数据类型。
2. 右键单击，然后在上下文菜单中选择命令 [Edit (编辑)]。
   ⇨ *TMC Editor (TMC 编辑器)* 打开 (TMC = TwinCAT 模块配置)。



3. 借助 *TMC Editors (TMC 编辑器)*，您可以根据需要更改数据类型，然后将其保存。

⇨ 按照保存步骤创建数据类型的新版本，并将 TwinCAT 项目中的原始版本标记为 *Hidden (隐藏)*，即可在 TwinCAT 项目中接受更改。

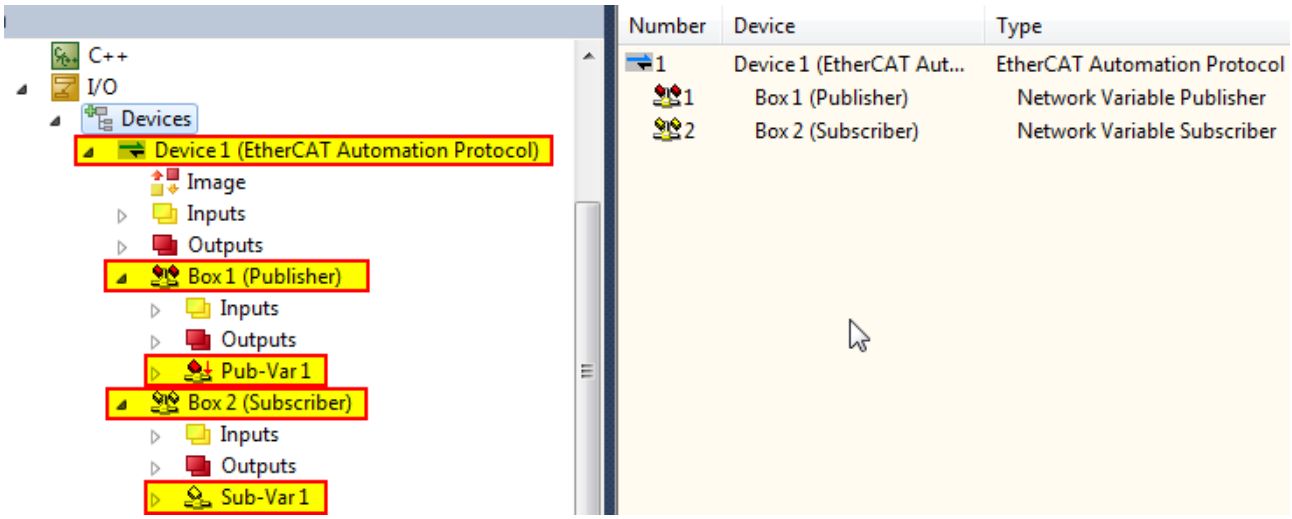
使用相关数据类型的 PLC 程序会自动使用最新版本的数据类型。为了在机器代码中也接受数据类型的更改，必须在更改后重新编译 PLC 项目。

如果存在此数据类型的 PLC 变量并将其与 EAP 设备的相应*发布服务器*或*订阅服务器变量*链接，一旦重新编译 PLC 项目，就会将最新版本的数据类型与*发布服务器/订阅服务器变量*一起使用。
如果 EAP 变量与 PLC 变量的链接不存在，则 EAP 设备将继续为其变量使用数据类型的原始旧版本。如果要使用新版本，则必须首先删除相关的 EAP 变量，然后再添加所需数据类型的新 EAP 变量。数据类型的旧版本将保留在 TwinCAT 项目中，直至不再引用该旧版本为止。

5 配置 EAP 设备

使用 TwinCAT 创建的 EAP 设备最初配置为标准设置，如章节创建 EAP 配置 [▶ 23]中所述。选择标准设置是为了使用户只需要确保接收器端数据变量的顺序与发送器端相同，且为要在发送器和接收器双方传输的数据变量选择相同的数据类型。以这种方式配置的 EAP 连接始终以*推送数据交换模式*通讯（参见通讯方式 [▶ 9]）。

可以借助 EAP 设备的配置选项和下级框（*发布服务器/订阅服务器，发布服务器/订阅服务器变量* – 参见下图）自由配置 EAP 通讯。此外，可以读取关于已启用 EAP 设备的当前配置信息。



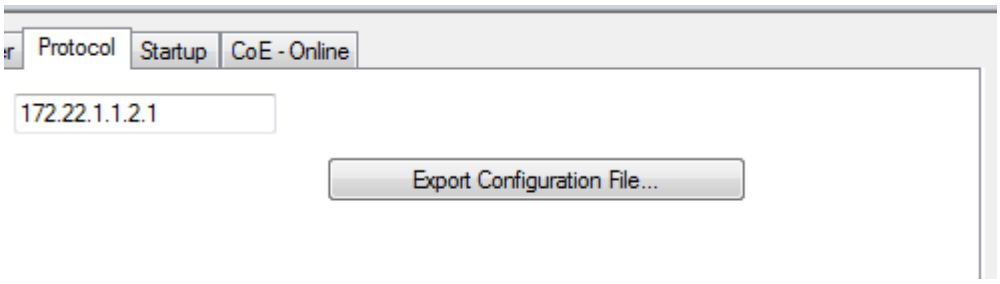
5.1 TwinCAT EAP 设备

借助 EAP 设备的配置选项，可以指定通过其发送 EAP 报文的网络接口卡（NIC）和通过 *ADS/AMS* 访问 EAP 设备的 *AMS NetID*。然后通过选择网络接口卡，自动指定可通过其访问 EAP 设备的 IP 地址（如果是 UDP/IP 通讯）。此外，还可以访问 EAP 设备的对象字典。

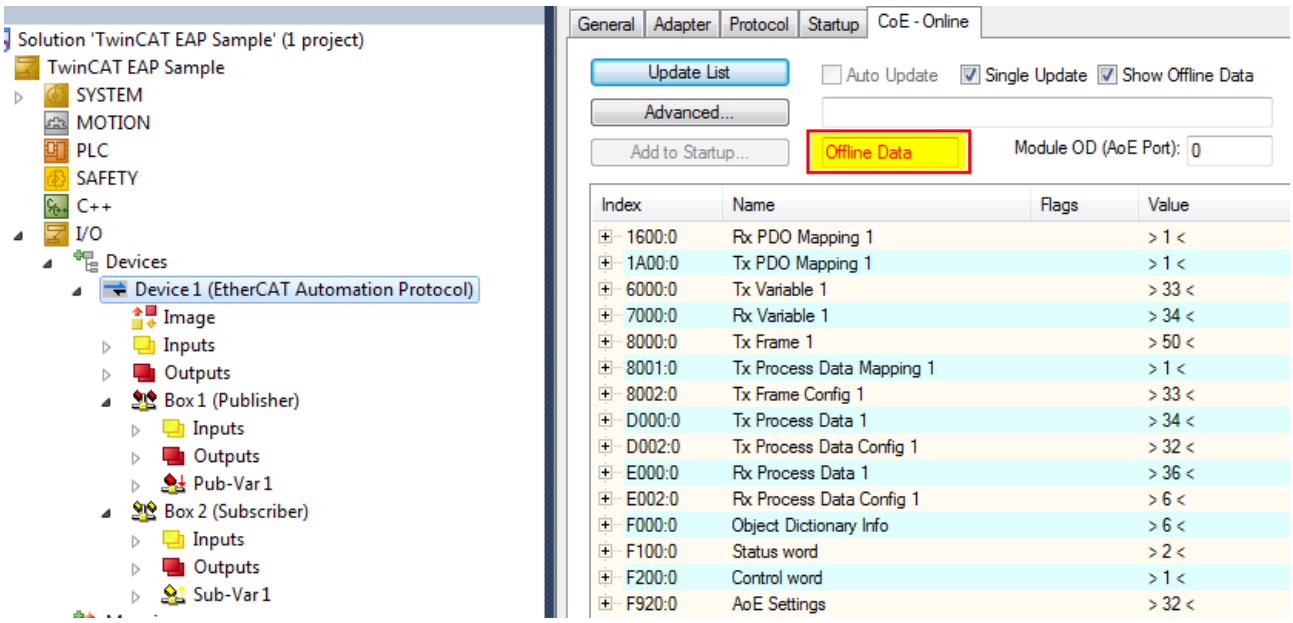
通用
*General（通用）*选项卡上的标准对话框适用于所有 TwinCAT 设备和框。可在此对话框中输入描述性名称和有关设备或框描述的有用注释。

适配器
*Adapter（适配器）*选项卡上的对话框显示所选网络接口卡，或启用分配的适配器。

协议
*Protocol（协议）*选项卡上的对话框（参见下图）允许分配特定的 *AMS NetID*，通过它，EAP 设备可在运行中由 *ADS/AMS* 寻址。
此外，还可以通过 [*Export Configuration File…(导出配置文件…)*] 按钮将加载项目的当前 EAP 配置修订导出至 XML 文件。此 XML 文件具有已定义的方案，也称为 *EAP 设备配置（EDC）*。

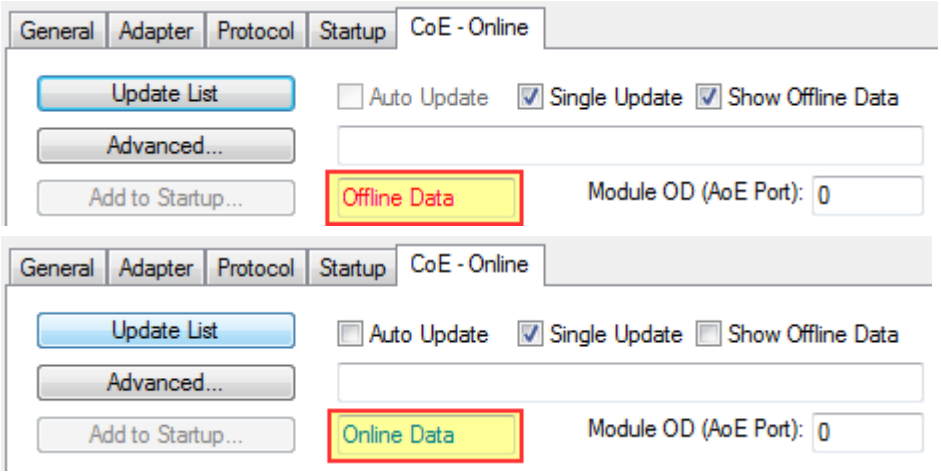


CoE – 在线
*CoE – Online（CoE – 在线）*选项卡上的对话框显示 EAP 对象字典（OD）。比较此处的章节 CANopen 对象字典 [▶ 46]和下图。借助 TwinCAT，一旦配置了 TwinCAT EAP 设备，将自动创建对象字典。它包含所有配置信息，并且一旦元素添加到当前配置中或从中移除，即会通过条目自动扩展或缩减。



CoE - Online（CoE - 在线）对话框的控件元素有以下含义：

状态
显示的对象字典状态在文本字段中输出（上图中的黄色背景）。如果 TwinCAT 未连接至已启用的 EAP 设备，则 *Offline Data（离线数据）*状态始终显示。例如，如果配置的 *AMS NetID* 不同于已启用 EAP 设备的实际 *AMS NetID*，则不会建立连接。否则，将显示 *Online Data（在线数据）*状态：



在在线目录中	在离线目录中
读取 EAP 设备的实际当前目录。这可能需要几秒钟，具体取决于大小和周期时间	显示 EAP 设备的离线目录。在这种情况下，修改无意义或无法进行。
可在 TwinCAT 对话框 *CoE - Online（CoE - 在线）*中看到绿色的 **Online（在线）**	可在 TwinCAT 对话框 *CoE - Online（CoE - 在线）*中看到红色的 **Offline（离线）**

读取另一个 EAP 设备实例的在线数据
可以将协议选项卡上的 *AMS NetID* 设置为网络中任何所需 EAP 设备的 *AMS NetID*，以便通过 *CoE - Online（CoE - 在线）*选项卡由 TwinCAT 读取在线数据。为此，必须启用 EAP 设备，并且该设备必须存在 *ADS/AMS 路由*。

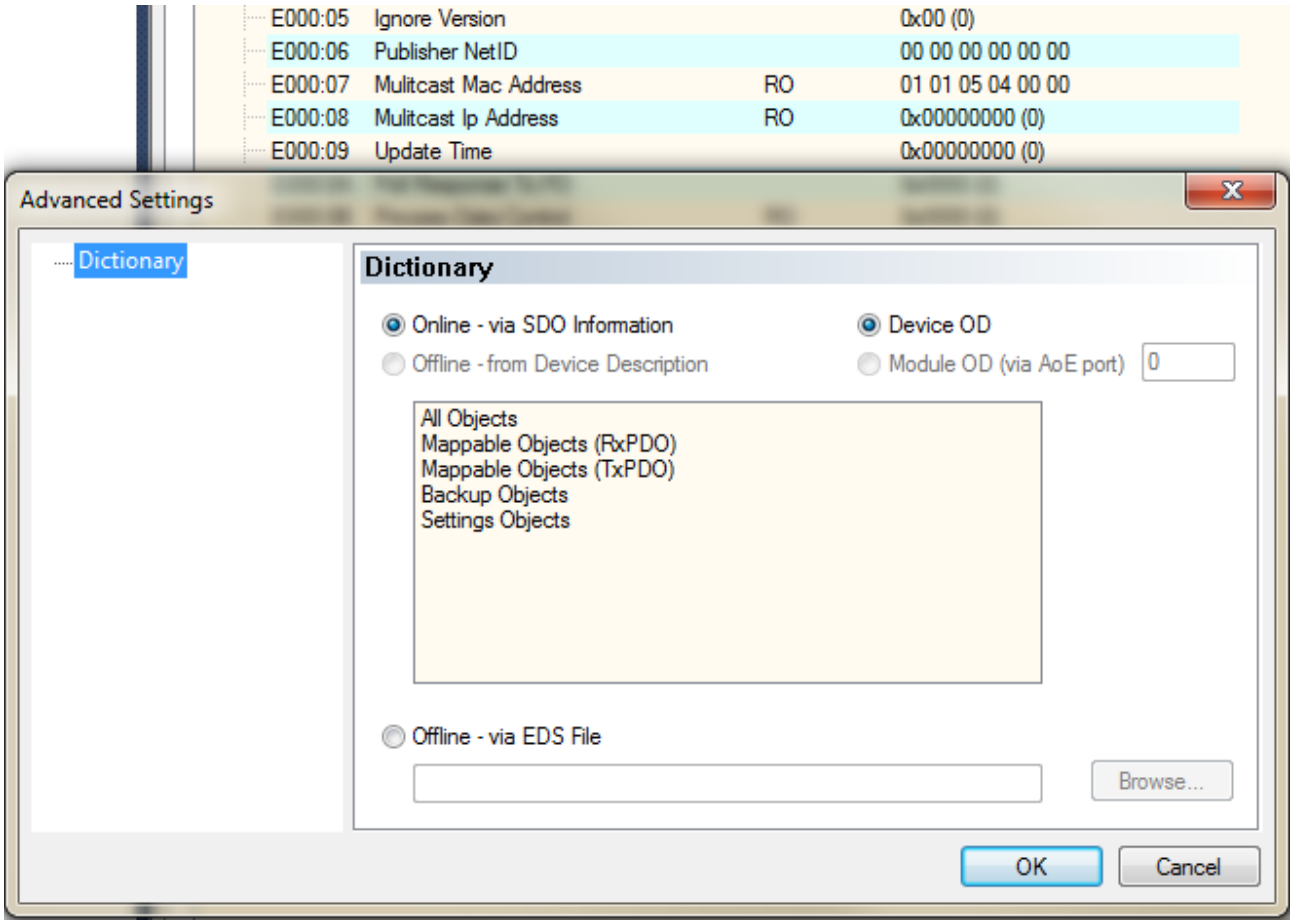
显示离线数据
借助 *Show Offline Data（显示离线数据）*选项，您可以设置在线显示或离线显示对象字典的内容。在线意味着从 EAP 设备读取并显示已启用配置的 *OD* 内容。离线意味着显示在当前加载的 TwinCAT 项目中借助 TwinCAT 配置的 *OD* 内容。

单次更新
如果标记了 *Single Update（单次更新）*选项，则始终在扩展对象（单击“+”符号）以显示子条目，或在窗口中滚动时，才从 EAP 设备精确读取 *OD* 内容。

自动更新
如果标记了 *Auto Update（自动更新）*选项，则会从 EAP 设备周期读取所有可见子条目的 *OD* 内容，并更新显示。

更新列表
*更新列表（Update List）*按钮用于从 EAP 设备读取所有可见对象条目的当前 *OD* 内容，并显示这些内容。

高级
单击 *Advanced（高级）*按钮可打开 *Advanced Settings（高级设置）*对话框（参见下图）。借助此对话框，可以从 EAP 设备读取整个或部分 *OD* 描述。该选项特别有利于在操作过程中将对象添加到对象字典中或从对象字典中删除，因为 TwinCAT 中显示的 *OD* 描述不再与 EAP 设备的当前 *OD* 一致。



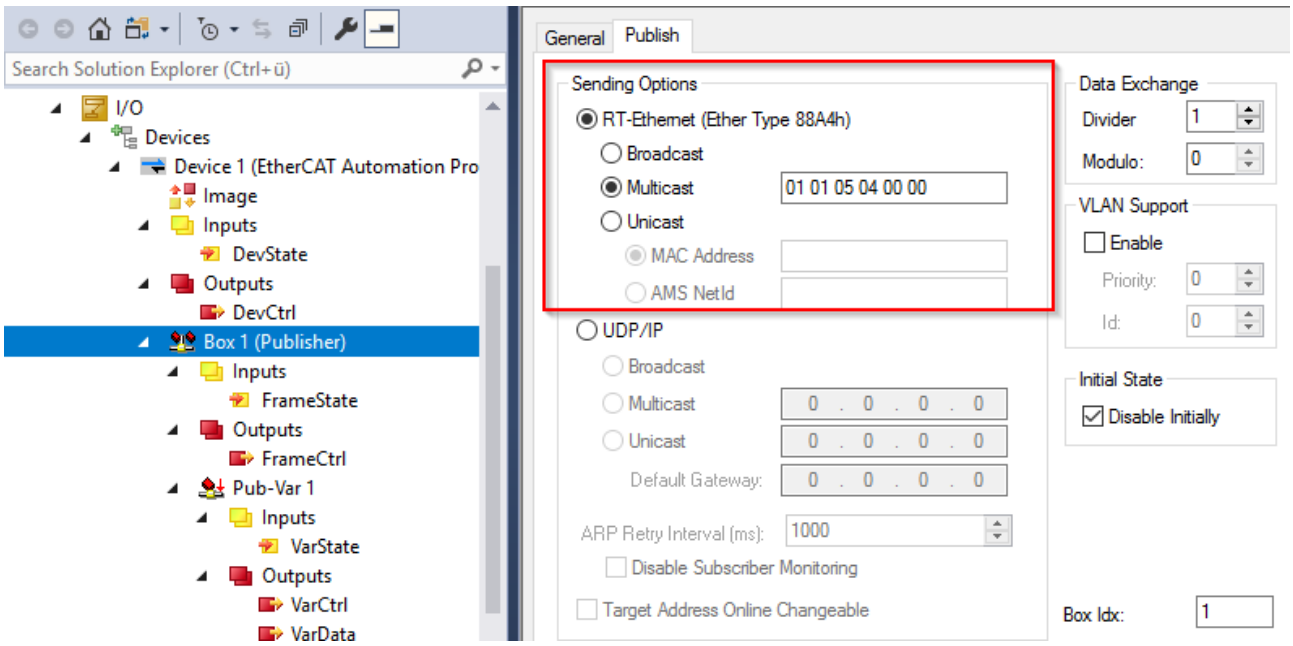
启动
TwinCAT 从对象字典中生成一个数据流，包括一系列发送到 EAP 设备的*启动*命令。一旦启用现有配置，将立即进行传输。生成的启动命令显示在 *Startup（启动）*选项卡的对话框中。

还请参阅有关此

🗎 TwinCAT EAP 设备 [▶ 38]

5.2 发布服务器框

用于发送 EAP 报文的基本协议在 *Sending Options（发送选项）*中的 *Publisher Box（发布服务器框）*配置页上进行设置（参见下图）。两种可用的基本协议：*以太网协议*和*用户数据报文协议*在章节基本原则 [▶ 7]中的章节通讯方式 [▶ 9]的网络协议部分中有所介绍。这两种协议中均可配置三种不同的连接方式 *Broadcast（广播）*、*Multicast（多播）*或 *Unicast（单播）*。如果是 *Multicast（多播）*和 *Unicast（单播）*连接方式，还必须定义一个目标地址，可借助该目标地址在网络中到达被访地址。



广播

*Broadcast（广播）*报文由一个网络设备传输到网络中的所有其他设备。*Broadcast（广播）*的每个收件人自行决定是否处理该信息。*以太网协议*级别的 *Broadcast（广播）*被发送到目标 *MAC* 地址 FF:FF:FF:FF:FF:FF，而 *UDP/IP* 级别的广播发送到 *IP* 地址 255.255.255.255。

多播

*Multicast（多播）*报文由一个网络设备传输到网络中所选的设备组。*Multicast（多播）*消息的收件人必须知道该消息发送到的 *Multicast（多播）*地址，并且必须将其报告给其网络接口卡。否则，网络接口卡将丢弃该 *Multicast（多播）*信息。
根据使用的基本协议，将*多播 MAC* 地址直接配置为目标地址，或者由 TwinCAT 转换为*多播 MAC* 地址的*多播 IP* 地址将配置为目标地址。*多播 IP* 地址必须在范围 224.0.0.0 到 239,255,255,255（IPv4）之间。

单播

*Unicast（单播）*报文由一个网络设备精确传输到另一个网络设备。如果寻址基于*以太网协议*进行，那么接收器的 *MAC* 地址将配置为目标地址。或者，也可以配置接收器的 *AMS NetID*。如果报文基于 *UDP/IP* 发送，那么接收器的 *IP* 地址将配置为目标地址（参见下图）。

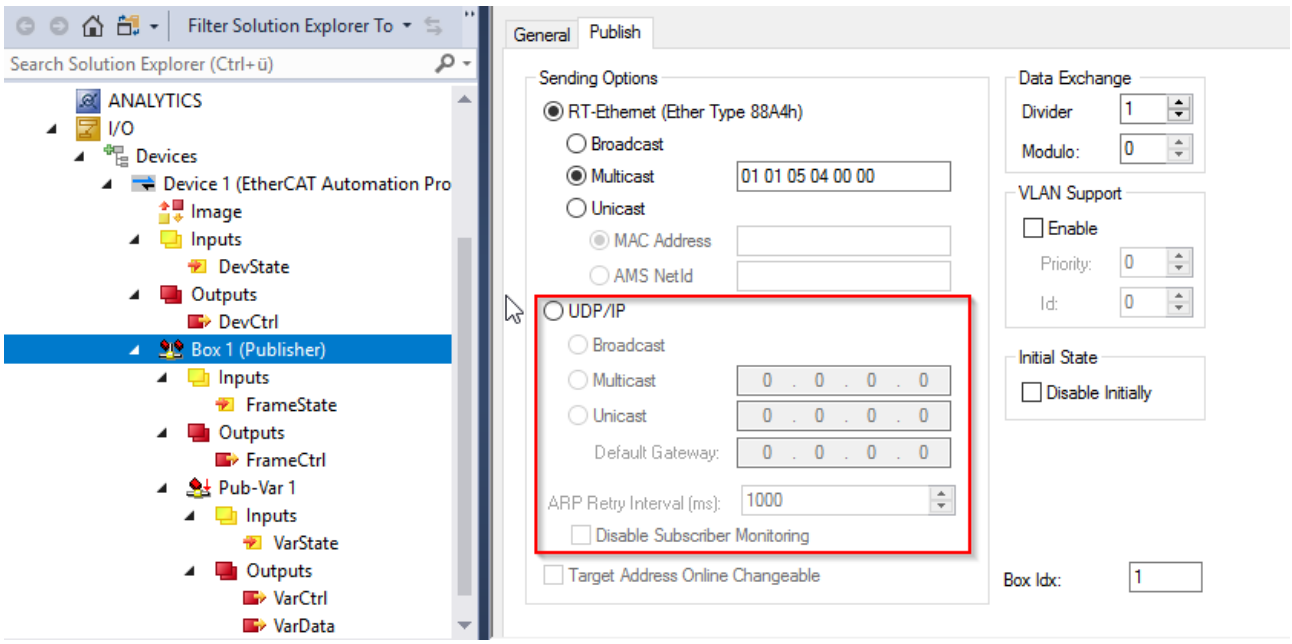
注意

使用广播和多播

在 *MAC* 级别或 *IP* 级别以*广播*或*多播*形式发送 EAP 报文会导致更高的网络负载，这取决于周期时间，因为这些报文被发送到所有网络设备！这可能会导致简单的网络设备（例如打印机）崩溃。在较短周期内，所有网络通信量可能被阻塞。
为避免网络过载或简单的非实时网络设备过载，建议

- 一方面使用*单播*寻址，
- 另一方面，将周期时间设置为绝对必要的最小时间。关于周期时间设置的说明可以在下文中找到。

⇨ 如果配置了连接类型*单播*，则默认情况下还将配置*订阅服务器监控*机制（参见通过 ARP 进行远程站监控 [▶ 11]）。

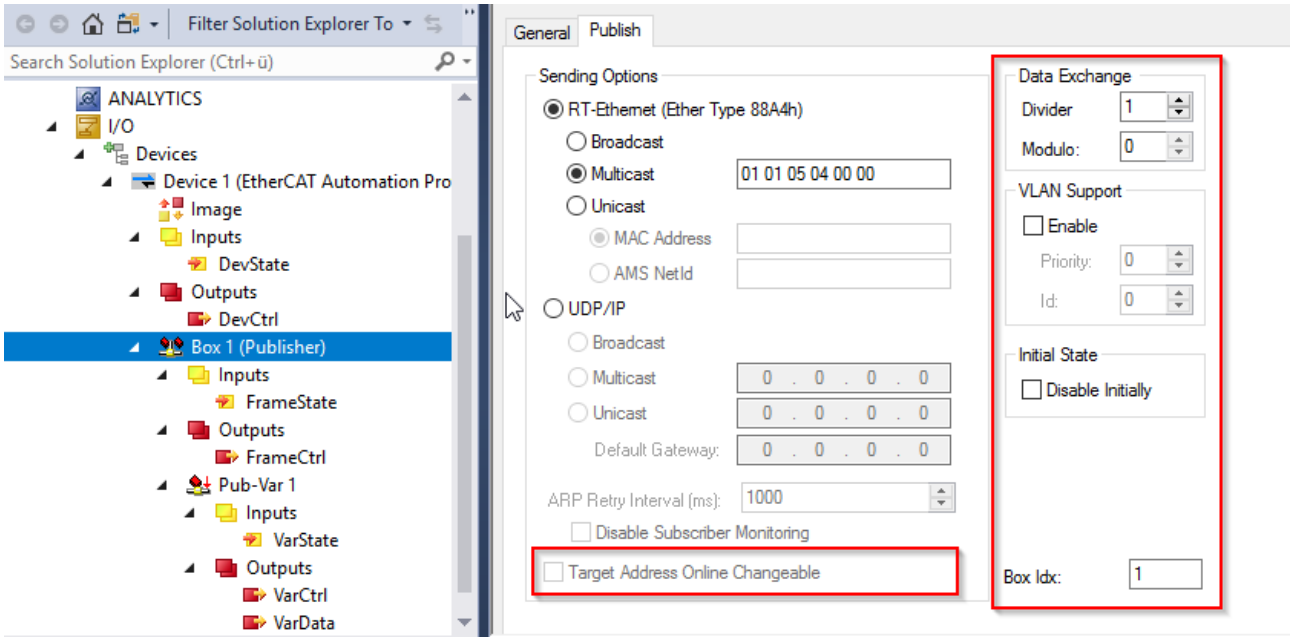


禁用订阅服务器监控

可以通过 *Disable Subscriber Monitoring（禁用订阅服务器监控）*选项禁用 *Subscriber Monitoring（订阅服务器监控）*机制。

ARP 重试间隔

输入字段 *ARP Retry Interval（ARP 重试间隔）*中的时间设置指定以毫秒（*ms*）为单位的时间间隔，在该时间间隔内将请求发送到接收器，以检查其可用性。



可在线更改目标地址

*Target Address Online Changeable（可在线更改目标地址）*选项也仅可用于*单播*。如果启用此功能，则在 EAP 设备的过程映像中还有另一个用于*发布服务器*的输出变量。根据配置的基本协议，此变量定义 *IP* 地址、*MAC* 地址或 *AMS NetID*。可借助 PLC 程序更改输出变量。此方法可用于动态更改已配置*发布服务器*的目标地址（另请参考章节添加发布服务器变量 [▶ 24]中的条件输出）。

数据交换

可借助 *Data Exchange（数据交换）*属性更改发送 EAP 报文的节奏（参见 EAP 发送机制 [▶ 11]）。

数据交换

如果正在使用 EL66xx，则无法使用 *Data Exchange（数据交换）*属性。

VLAN 支持
可以借助与管理型交换机连接的 *VLAN Support（VLAN 支持）*属性，为 EAP 报文指定通过 *VLAN（虚拟局域网）*的固定路由。如果启用了 *VLAN*，则 EAP 信息将带有 *VLAN 标题*。因此，有两个属性用于确定所需的 *VLAN* 并指定在虚拟网络中处理信息的*优先级*：

- *VLAN 信息 ID*：定义要在其中发送信息的 *VLAN ID*（介于 0 至 4095 之间），以及
- *VLAN 信息优先级*：定义信息在 *VLAN* 中的*优先级*（高优先级 = 7，低优先级 = 0）。

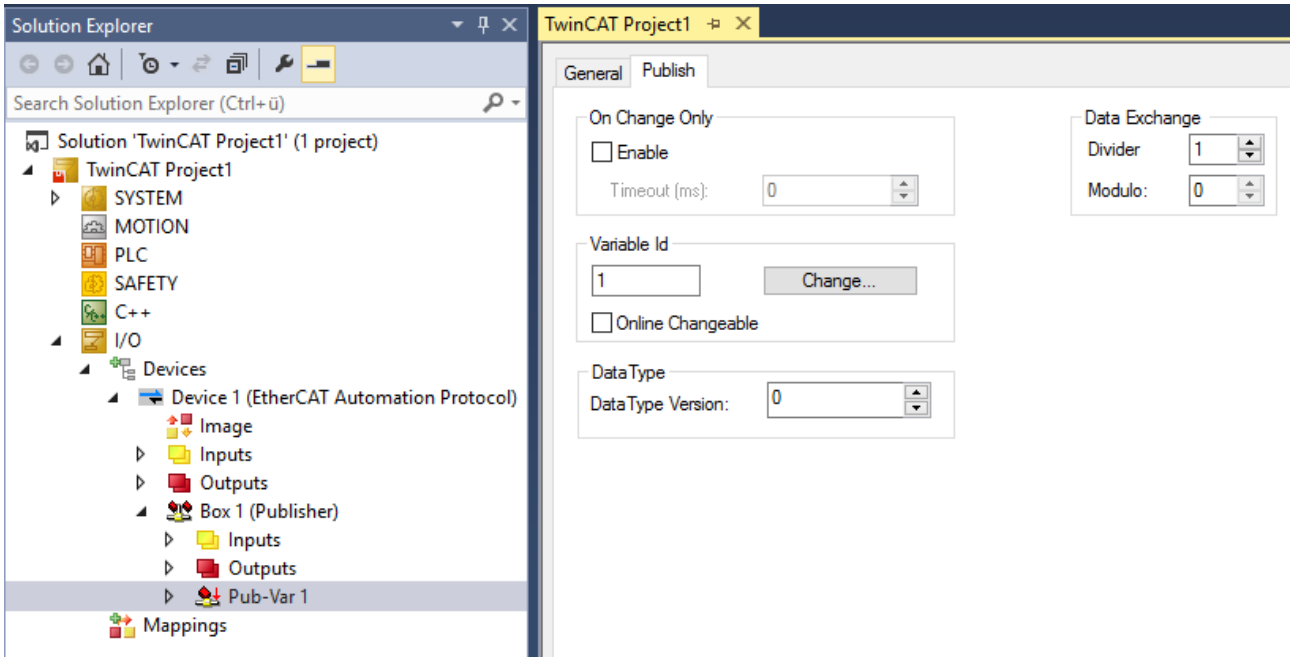
初始状态

借助属性“禁用初始化”，可以在启动系统后阻止发布服务器发送数据包。随后可通过将 FrameState 的值设置为 0 来启用发送。
框索引：

不同发布服务器和订阅服务器的顺序数，只读。

5.3 发布服务器变量

每个*发布服务器变量*还具有其自己的特殊属性，可以通过其自己的配置页进行参数化（参见下图）。



仅改变时
如果启用 *On Change Only（仅改变时）*选项，则只有*发布服务器变量*的值改变时，*TxProcessData* 才会与 *TxFrame* 一起发送。*Timeout（超时）*字段中的值指定最后一次发送*发布服务器变量*后且再次发送之前经过的毫秒数（ms），即使此时*发布服务器变量*的值不应改变（EAP 发送机制 [▶ 11]中的更多详细信息）。

仅改变时
如果正在使用 EL66xx，则无法使用 On Change Only（仅改变时）属性。

变量 Id
*Variable Id（=ProcessData ID）（变量 ID）*是*发布服务器变量*的标识号。如果标记了 *Online Changeable（可在线更改）*选项，则可以借助 PLC 程序在线更改。

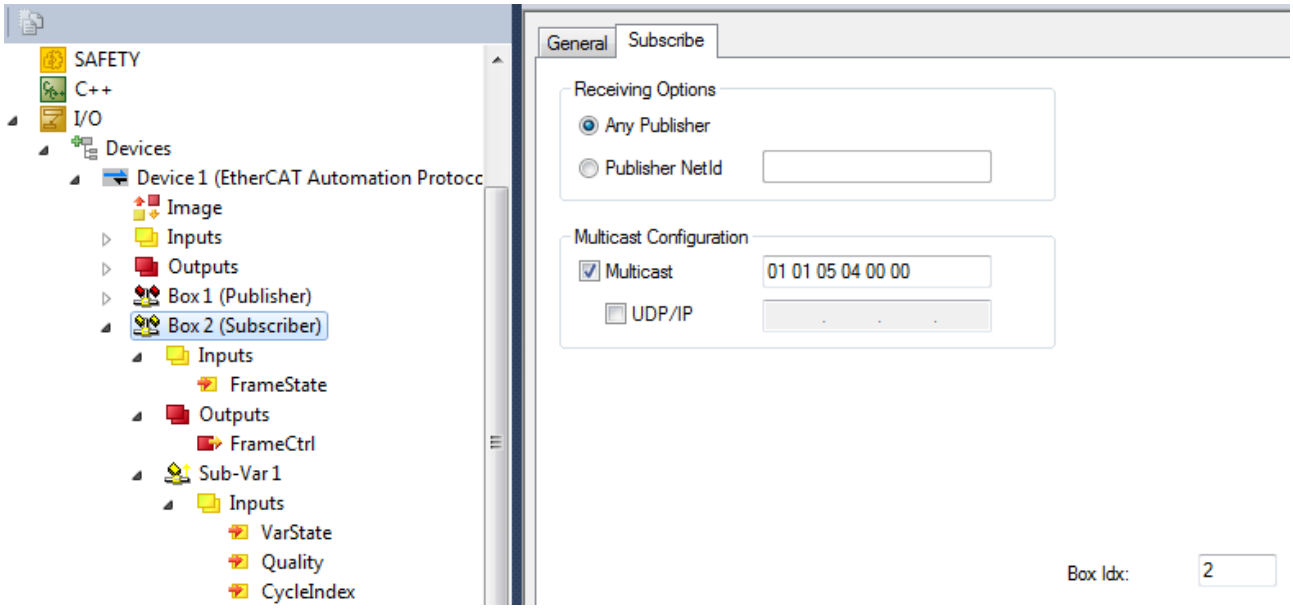
数据类型版本
可在此处指定版本号。如果在收到变量时检查了两个版本号是否相等，则必须在订阅服务器端配置相同的版本号（默认进行此比较）。版本号用于确保发布服务器变量的数据类型与相应的订阅服务器变量匹配。如果仅在发布服务器或订阅服务器端更改数据类型，则必须增加版本号，以防止接收该变量，从而防止用错误的数据类型解释其值。

数据交换
可以借助 *Data Exchange（数据交换）*属性更改发送*发布服务器变量*的节奏（更多信息参见 EAP 发送机制 [▶ 11]）。

数据交换
如果正在使用 EL66xx，则无法使用 *Data Exchange（数据交换）*属性。

5.4 订阅服务器框

通常与*订阅服务器*配置页面定义的 EAP 报文的接收有关的属性（参见下图）。这些属性包含：



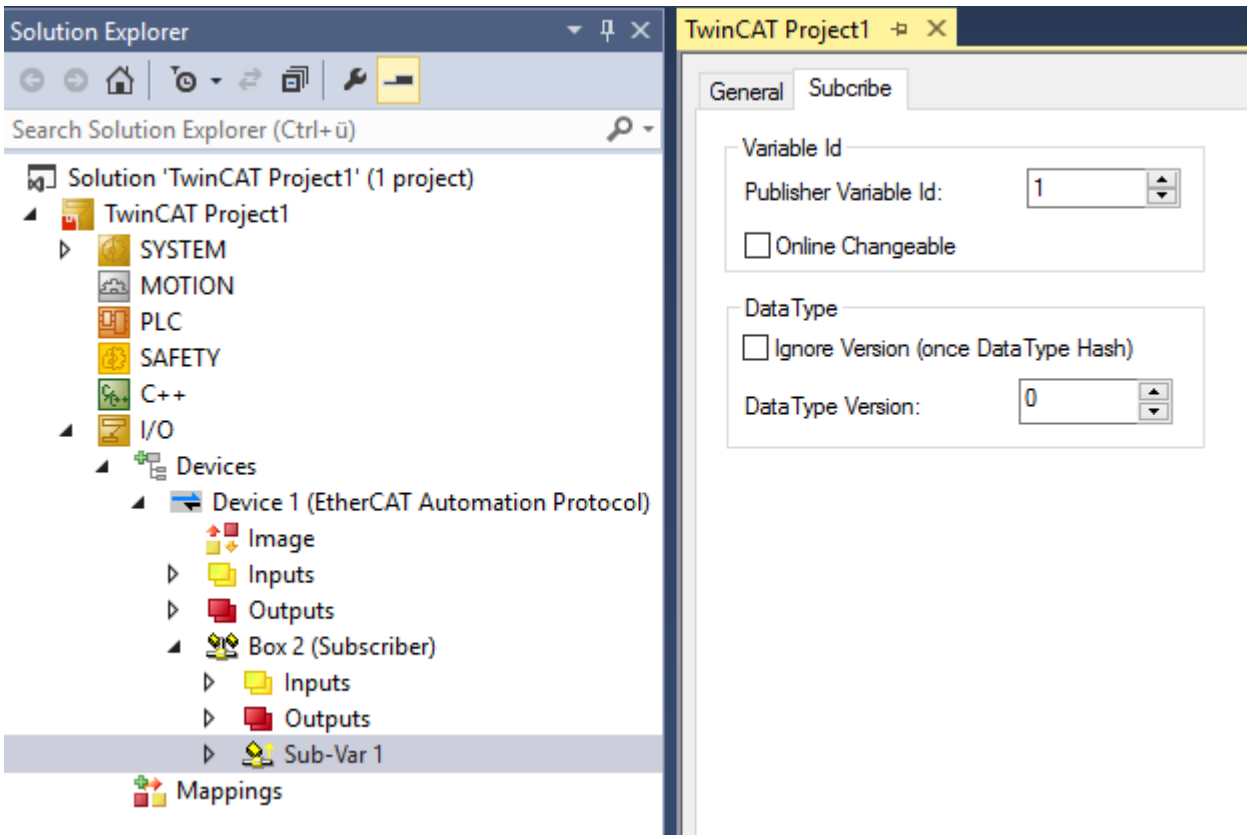
接收选项
通过选择 *Receiving Options（接收选项）*，您可以配置是否应接收所有进入的 EAP 报文 – 即不考虑发送器 – 还是仅接收来自特定发送器的 EAP 报文。在后一种情况下，应在 *Publisher NetId（发布服务器 NetId）*后的输入域中输入期望发送器的 *AMS NetID*。

多播配置
借助 *Multicast Configuration（多播配置）*选项，您可以指定是否应将输入的*多播*地址报告至配置的网络接口卡。此处参数化的*多播*地址必须与为发送器配置的*多播*地址完全相同，前提是要接收后者的 EAP 报文（参见章节发布服务器框 [▶ 39]中的*多播*）。

标准多播 MAC
即使未选择 *Multicast（多播）*选项，也始终向 TwinCAT EAP 设备的网络接口卡报告 Beckhoff 标准*多播*地址 01:01:05:04:00:00。

5.5 订阅服务器变量

特别与*订阅服务器变量*相关的属性在*订阅服务器变量*配置页面上定义（参见下图）。这些属性包含：



变量 Id

*Variable Id (=ProcessData ID)（变量 ID）*是*订阅服务器变量*的标识号。如果标记了 *Online Changeable（可在线更改）*选项，则可以借助 PLC 程序在线更改。

数据类型

每个*订阅服务器变量*都有一个版本号（参见章节 EAP 报文结构 [▶ 15]中的版本）。版本号可在“DataType Version（数据类型版本）”中配置。在接收到*订阅服务器变量*之前验证（参考通讯方式 [▶ 9]中的网络协议部分）。如果版本号不匹配，则丢弃传入的 *ProcessData*。如果启用了 *Ignore Version (once DataType Hash)(忽略版本)(一旦数据类型散列)*，则忽略此验证。

成功数据交换的配置设置

为了实现从*发布服务器变量*到*订阅服务器变量*的数据交换，涉及的控制计算机配置必须彼此匹配。章节通讯方式 [▶ 9]的网络协议部分包含对 EAP 报文或*发布服务器变量*的接收顺序描述。结合以下方面，可清楚了解如何保证数据交换：

- 必须选择*发布服务器*的目标地址，以便 EAP 报文到达收件人。带*广播*或*多播*寻址的报文到达每个网络设备。必须为*单播*报文配置收件人的准确目标地址。
- 一旦在接收器（*订阅服务器*）的 *Receiving Options（接收选项）*中选择了 *Any Publisher（任意发布服务器）*选项，则每个进入的 EAP 报文均会被接收并进行进一步处理，而不考虑其发送器。
  仅在以下情况下例外：
  - 在发送器端配置的*多播*地址与 TwinCAT EAP *多播 MAC*（01:01:04:05:00:00）中不同，且
  - 发送器端（*发布服务器*）的目标地址与*订阅服务器*端配置的*多播 MAC* 地址不匹配。
- 如果在接收器（*订阅服务器*）的 *Receiving Options（接收选项）*中指定了 *Publisher NetId（发布服务器 NetId）*，则仅接收和进一步处理来自指定发送器（*发布服务器*）的 EAP 报文。
- 在网络中，*发布服务器变量*和*订阅服务器变量*的 *ID* 必须完全相同且唯一。如果发送的*发布服务器变量 ID* 与*订阅服务器变量 ID* 不匹配，则在接收后丢弃该变量。
- *发布服务器变量*的版本（散列）和关联的*订阅服务器变量*必须匹配。除非接收器（*订阅服务器*）中的 *Ignore Data Type Hash（忽略数据类型散列）*已启用，否则其*版本*不匹配*订阅服务器变量版本*的已发送*发布服务器变量*将在接收时被丢弃。
- *发布服务器变量*的原始数据长度必须与*订阅服务器变量*的预期原始数据长度匹配。否则将在接收时丢弃*发布服务器变量*。

5.6 TwinCAT 2 和 3 之间的 EAP

EtherCAT 自动化协议与 TwinCAT 2 中的常规网络变量 (*NWV*) 兼容。但是，如果要在 TwinCAT 2 *NWV* 和 TwinCAT 3 EAP 之间建立通讯，则需要对 TwinCAT 3 下 EAP 的网络变量进行一些扩展。

遵守发布服务器 NetID

与 *NWV* 相反，TwinCAT EAP 设备是一个独立的软件设备，以 *AMS NetID* 作为目标地址。因此，在 EAP 设备发送的报文中，此 *AMS NetID* 在*过程数据帧标题*的*发布服务器 ID* 域中作为发送器地址输入。对于 *NWV*，在此位置输入控制计算机自身的 *AMS NetID* (即 TwinCAT 路由器 NetID)。

由于此差异，在接收 EAP 报文时，TwinCAT EAP 设备运作如下：

✓ 假设将具体的 *AMS NetID* 取代*任意发布服务器*输入接收器 (*订阅服务器*) 的 *Receiving Options (接收选项)* 类别中，作为可从中接收报文的发送器地址 *(发布服务器 NetID)*。

1. 如果配置的*发布服务器 NetID* 是一个路由器 NetID (在最后两个位置始终为值 .1.1)，则在接收时，EAP 设备仅将发送器地址的前 4 个字节与*发布服务器 NetID* 进行比较。
2. 但是，如果配置的*发布服务器 NetID* 属于 EAP 设备的 NetID (最后两个位置的数值从来不是 .1.1，而是更大数值，例如 .2.1)，则接收时，EAP 设备将发送器地址的所有 6 个字节与*发布服务器 NetID* 进行比较。

⇨ 使用此逻辑，可以在接收 EAP 报文时配置以下过滤：

- 可以接收来自整个网络的所有 EAP 报文，无论其来自哪个发送器 (*任意发布服务器*)。
- 可以接收来自特定控制计算机的所有 EAP 报文，即从该控制计算机上运行的所有 EAP 设备接收报文。然后，控制计算机的路由器 NetID 必须作为*发布服务器 NetID* 输入。
- 仅可接收来自特定 EAP 设备的 EAP 报文。然后，传输 EAP 设备的 *AMS NetID* 必须作为*发布服务器 NetID* 输入。

“Ignore Data Type Hash (忽略数据类型散列)”选项

需要注意的另一点是，TwinCAT 版本 2 和 3 之间的数据类型系统的扩展：

使用复杂 (即非本机) 数据类型时，TwinCAT 2 计算出一个明确识别该数据类型的 16 位散列值。对于本机数据类型，该数据类型散列始终为 0。发送发布服务器变量时，数据类型散列将作为版本号一起发送。然后，在接收器端将此版本号与已配置的订阅服务器变量的数据散列类型进行比较。如果比较报表相关性，则订阅服务器接受该数据。

在 TwinCAT 3 中，每个数据类型均分配了一个全局唯一标识符 (GUID)。其数据长度为 128 位。因此，在配置发布服务器或订阅服务器变量时，始终将数值 0 作为版本号。

由于 TwinCAT 2 和 3 之间的这一差异，在使用复杂数据类型时，必须遵循以下步骤：

✓ 假设要在 TwinCAT 2 和 TwinCAT 3 之间发送复杂的、非本机数据类型的变量。

1. 这种情况下，必须为订阅服务器变量激活“Ignore Data Type Hash (忽略数据类型散列)”。

⇨ 此设置取消了接收器端的比较操作，并且订阅服务器无需比较版本即可接受数据。

通过“Browse for Computer (浏览计算机)”或“Browse for File (浏览文件)”创建订阅服务器变量。

由于将 TwinCAT 2 项目的文件格式更改为 TwinCAT 3 项目，用于创建与现有发布服务器变量匹配的订阅服务器变量的两个变量“Browse for Computer (浏览计算机)”或“Browse for File (浏览文件)”目前仅向下兼容。也就是说，当前只能使用来自 TwinCAT 3 系统的两个变量，以便基于发布服务器变量从 TwinCAT 2 项目自动创建订阅服务器变量。相反，则必须手动创建订阅服务器变量 (变量“创建新变量”)。

6 CANopen 对象字典

CiA 组织（自动化中的 CAN）追求的目标之一是通过设备描述的标准化来创建同一类型设备之间的顺序和可交换性。为此目的定义了所谓的 *CANopen* 配置文件，其决定性地描述了设备可更改和不可更改的参数。此类参数至少包括下列特征：

- **索引编号** – 用于明确标识所有参数。
  索引编号分为主索引和子索引，用于标记并排列相关参数。子索引通过冒号“:”分隔。
  这实现了两级（逻辑程序段）排列。主索引始终以十六进制形式在数值范围 0…65535（0x0…0xFFFF）内使用。子索引通常以十进制形式在数值范围 0…255（0x0…0xFF）内使用。
- **正式名称** – 采用易于理解、自描述的文本形式
- **可访问性** – 例如参数是只能读取还是也可以写入
- **数据类型** – 根据参数不同，其类型可以是文本（字符串）、数字（整数、实数）、布尔或字节域。

CANopen 配置文件中定义了参数的索引编号分配。通过这种方式，所有参数如同在表格中一样按层次结构进行组织。然后，该表格包含所有特定于设备的参数。称为 *CANopen 对象字典*（*OD*）。

TwinCAT EAP 设备的所有参数均借助对象字典进行类似组织。原则上，其结构与 *CANopen OD* 的结构完全相同。EAP 设备的 *OD* 配置文件由 EtherCAT 技术协会（ETG）在 EtherCAT 自动化协议规定中指定（ETG 1005，参见网页 www.ethercat.org）。
此配置文件由配置文件编号 5002 识别。其定义了配置文件类型（主配置文件），并保存在 *OD 参数设备类型*的低位字（位 0-15）中。高位字（位 16-31）包含编号 1000。其定义模块配置文件（子配置文件）。这将为*设备类型*参数生成一个数值 0x03e8138a（65541002_{dec}），该值也保存在*标识对象*中的*产品代码*下（索引 0x1018:02）。

OD 中的对象示例：
TwinCAT EAP 设备的配置文件根据四个参数明确识别。这些内容汇总在名为 *Identity（标识）*的逻辑程序段中，其主索引为 4120（0x1018）。*Vendor ID（供应商 ID）*参数的索引编号为 4120:01（0x1018:01），输入值 *2* 作为 Beckhoff 设备的标识符。
逻辑程序段 *Identity（标识）*也被指定为对象，并从用户的角度表示如下：

1018:0	Identity		> 4 <
1018:01	Vendor ID	RO	0x00000002 (2)
1018:02	Product Code	RO	0x03E8138A (65541002)
1018:03	Revision Number	RO	0x00030000 (196608)
1018:04	Serial Number	RO	0x00000000 (0)

*Identity（标识）*对象的所有参数均具有属性 *RO*（只读），因为用户不得更改参数。

借助*对象字典*的参数，可以描述许多不同的属性。此类参数的示例包括供应商标识符、版本号、过程数据设置、设备名称和校准值等。EAP 设备的调试以及诊断需要 *OD* 内容，并且可能非常广泛。

6.1 EAP 对象字典（子配置文件 1000）

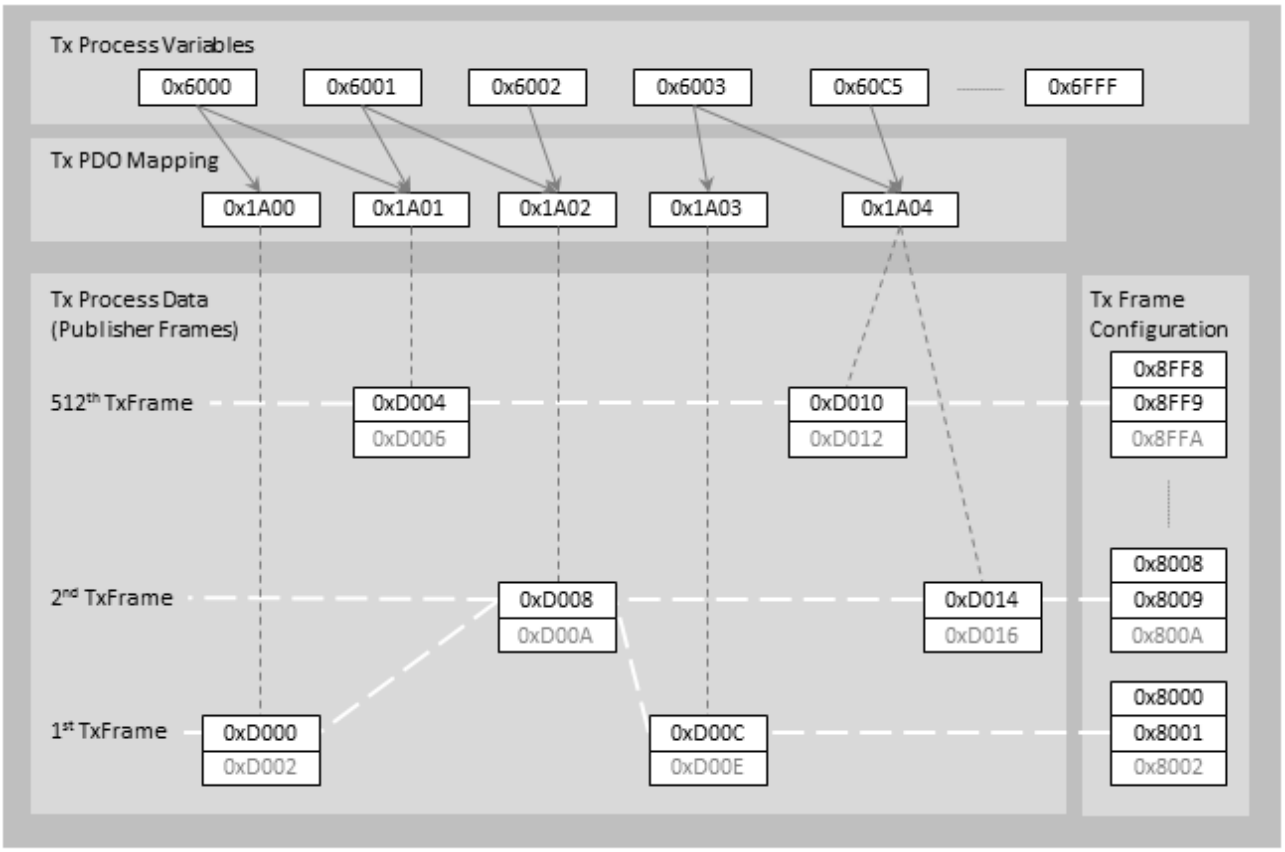
EAP *对象字典*分为标准对象和配置文件特定对象。标准对象对所有模块具有相同意义。配置文件特定对象对于支持配置文件类型 5002 的所有模块具有相同意义。除此之外，对象可以是静态或动态。只要 EAP 设备本身的实例存在，静态对象就存在。在 EAP 设备运行过程中，可以生成、也可以再次删除动态对象。

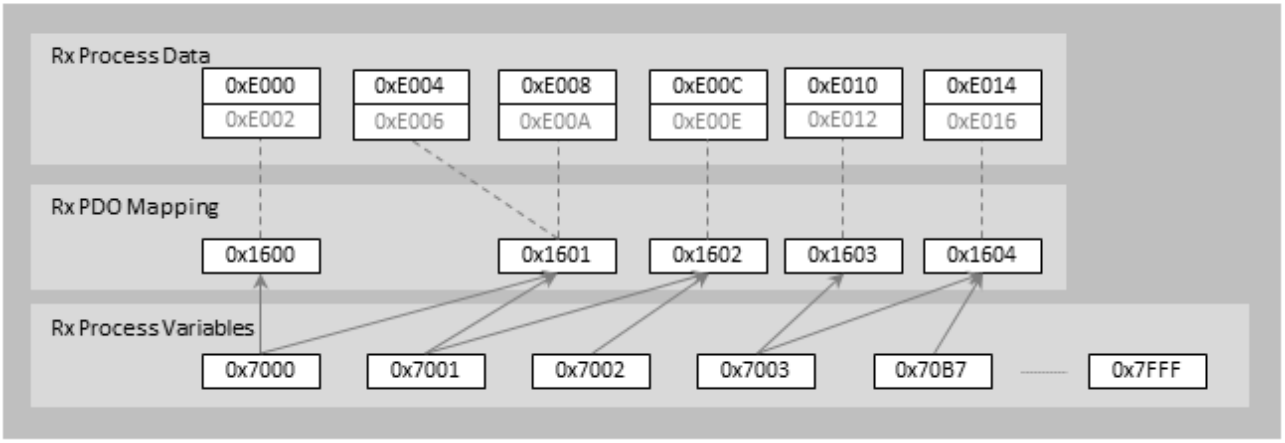
对象字典的划分

EAP 设备的*对象字典*分为以下范围：

- **索引 0x1000 – 0x1FFF**：描述通讯配置文件的范围。
  有关设备标识的常规信息（如名称、供应商和序列号等）保存在范围 0x1000 – 0x1018 内。
  此外，*PDO Mapping（PDO 映射）*对象（*PDO = ProcessDataObject*）在范围 0x1600 – 0x17FF 和 0x1A00 – 0x1BFF 内定义。*PDO Mapping（PDO 映射）*定义汇总 *OD* 其他对象的哪些内容以构成 *PDO*。然后，*PDO* 描述实时周期性发送的用户数据内容。
- **索引 0x6000 – 0x9FFF**：描述功能相关参数的范围。
  在 ETG 标准 1005 中指定功能相关参数。这些参数及其结构在设备配置文件编号 5002、模块配置文件 1000 下定义。此定义构成通过 EtherCAT 自动化协议进行数据交换的基础。以下区域涉及配置文件的各个对象类型以及其结构关系。

- 索引 0xF000 – 0xFFFF：描述设备特定属性的范围。
  在此范围内有一些对象，借助这些对象可以通过 TwinCAT EAP 设备执行诊断和控制功能。





标准化配置文件范围的对象类型及其结构

下面列出了动态对象，并解释了它们之间的关系。上图显示了这些关系：

用于参数化*订阅服务器*的对象：

- *RxVariable* [0x7000+n … 0x7FFF]：
  RxVariable 定义可与控制应用程序的相应输入变量链接的任何类型的变量（例如 PLC）。
- RxProcessDataObject（*RxPDO*）[0x1600+n … 0x17FF]：
  RxPDO 定义将一项过程数据表示为一个单元的 *RxVariables* 的订购数量。
- RxProcessData（*RxPD*）[0xE000+4*n … 0xEFFC]：
  RxPD 定义接收 *PDO* 的属性（参见订阅服务器框 [▶ 43]和订阅服务器变量 [▶ 43]）。因此，*RxPD* 代表了 EAP 通讯的主接收单元。
- RxProcessDataInfo [0xE002+4*n … 0xEFFE]：
  RxPDInfo 对象通过 EAP 规范中未找到的单个属性（特别是属于 TwinCAT EAP 设备的属性）扩展 *RxPD* 对象。

用于参数化发布服务器的对象：

- TxVariable [0x6000+n … 0x6FFF]：
  TxVariable 对象定义可与控制应用程序的相应输出变量链接的任何类型的变量（例如 PLC）。
- TxProcessDataObject（*TxPDO*）[0x1A00+n … 0x1BFF]：
  TxPDO 定义将一项过程数据表示为一个单元的 *TxVariables* 的订购数量。
- TxProcessData（*TxPD*）[0xD000+4*n … 0xDFFC]：
  TxPD 对象定义用于传输 *PDO* 的属性（参见发布服务器变量 [▶ 42]）。因此，*TxPD* 代表了 EAP 通讯的主传输单元。
- TxProcessDataInfo [0xD002+4*n … 0xDFFE]：
  TxPDInfo 对象通过 EAP 规范中未找到的单个属性（特别是属于 TwinCAT EAP 设备的属性）扩展 *TxPD* 对象。
- TxFrame [0x8000+n*8 ... 0x8FF8]：
  TxFrame 对象定义用于在网络中传输一个或多个 *TxPDs* 的传输属性（参见发布服务器框 [▶ 39]）。
- TxPD 分配 [0x8001+n*8 … 0x8FF9]：
  将 *TxPDAssignment* 对象分配给每个 *TxFrame* 对象。*TxPDAssignment* 对象的索引高于 *TxFrame* 对象的索引。分配对象指定在相应的 *TxFrame* 中一起发送哪些 *TxPDs*。
- TxFrameInfo [0x8002+n*8 … 0x8FFA]：
  TxFrameInfo 对象通过 EAP 规范中未找到的单个属性（特别是属于 TwinCAT EAP 设备的属性）扩展 *TxFrame* 对象。

标准对象（0x1000-0x1FFF）

静态对象

索引 1000 设备类型

索引	名称	含义	数据类型	标记	缺省
1000:0	设备类型	EAP 设备类型： 低位字包含使用的 CoE 配置文件（5002）。 高位字包含使用的 CoE 配置文件（1000）。	UINT32	RO	0x03E8138A (65541002_{dec})

索引 1008 设备名称

索引	名称	含义	数据类型	标记	缺省
1008:0	设备名称	EAP 设备名称	字符串 [256]	RO	EtherCAT 自动化协议

索引 100A 软件版本

索引	名称	含义	数据类型	标记	缺省
100A:0	软件版本	EAP 设备软件版本	UINT32	RO	0x00000000 (0_{dec})

索引 1018 标识

索引	名称	含义	数据类型	标记	缺省
1018:0	标识	EAP 设备标识信息	UINT8	RO	0x04 (4_{dec})
1018:01	供应商 ID	EAP 设备供应商 ID	UINT32	RO	0x00000002 (2_{dec})
1018:02	产品代码	EAP 设备产品代码	UINT32	RO	0x03E8138A (65541002_{dec})
1018:03	产品修订	EAP 设备修订号	UINT32	RO	0x00030000 (196608_{dec})
1018:04	序列号	EAP 设备序列号。 0 表示未使用	UINT32	RO	0x0 (0_{dec})

动态对象

索引 1600-17FF RxPDO 映射

索引	名称	含义	数据类型	标记	缺省
1600+n:0	所用元素编号	RxPDO 映射对象中的条目编号	UINT8	RW	#(分索引)
1600+n:01-255	RxVariable m	位 0-7：输入对象的位长（如果 PDO 有间隙，对应于间隙位长） 位 8-15：输入对象的分索引（如果 PDO 有间隙，则为 0） 位 16-31：输入对象的索引（如果 PDO 有间隙，则为 0）	UINT32	RW	-

索引 1A00-1BFF TxPDO 映射

索引	名称	含义	数据类型	标记	缺省
1A00+n:0	所用元素编号	TxPDO 映射对象中的条目编号	UINT8	RW	#(分索引)
1A00+n:01-255	TxVariable m	位 0-7：输入对象的位长（如果 PDO 有间隙，对应于间隙位长） 位 8-15：输入对象的分索引（如果 PDO 有间隙，则为 0） 位 16-31：输入对象的索引（如果 PDO 有间隙，则为 0）	UINT32	RW	-

配置文件特定对象（0x6000-0xFFFF）

静态对象

索引 F100 EAP 状态信息

索引	名称	含义	数据类型	标记	缺省
F100:0	EAP 状态	EAP 设备状态信息	UINT8	RO	0x02 (2_{dec})
F100:01	状态字	低字节代码表示 EAP 设备的当前状态： 0 = 无效 1 = 初始化 2 = 操作前 4 = 安全操作 8 = 操作 高字节代码表示是否发生错误： 0 = 无错 1 = 出错	UINT16	RO	0x0008 (8_{dec})
F100:02	状态错误代码	用于标识已发生错误的错误编号。0 表示未识别到错误。	UINT32	RO	0x03E8138A (65541002_{dec})

索引 F200 EAP 控制信息

索引	名称	含义	数据类型	标记	缺省
F200:0	EAP 控制	用于检查 EAP 设备状态的参数	UINT8	RO	0x01 (2_{dec})
F200:01	控制字	对将 EAP 设备置于期望状态的请求进行编码： 1 = 初始化 2 = 操作前 4 = 安全操作 8 = 操作	UINT16	RO	0x0008 (8_{dec})

索引 F020-F022 帧列表

索引	名称	含义	数据类型	标记	缺省
F020+n:0	所用元素编号	已配置 TxFrame 的编号	UINT8	RW	#(分索引)

索引	名称	含义	数据类型	标记	缺省
F020+n:01-254	框 1（发布服务器）	值 0x0000 0000 = 第一个 TxFrame 对象（索引 8000）不存在 值 0x0000 03E8（ = 1000）：第一个 TxFrame 对象（索引 8000）存在 不允许其他值。 该对象可用于生成/删除 TxFrames	UINT32	RW	0x000003E8 (1000 $_{dec}$)

索引 F800 EAP 信息

索引	名称	含义	数据类型	标记	缺省
F800:0	所用元素编号	EAP 信息对象中的条目编号	UINT8	RW	0x08 (8_{dec})
F800:01	可用的 Tx Var	指示配置的 TxVariable 对象的最大编号 (0x6nnn)。	UINT16	RW	–
F800:02	可用的 Rx Var	指示配置的 RxVariable 对象的最大编号 (0x7nnn)。	UINT16	RW	–
F800:03	可用的 Tx 过程数据	指示配置的发送过程数据对象的最大编号 (0xDnnn)。	UINT16	RW	–
F800:04	可用的 Rx 过程数据	指示配置的 RxProcessData 对象的最大编号 (0xEnnn)。	UINT16	RW	–
F800:05	可用的 Tx PDOs	指示配置的 TxPDO 对象的最大编号 (0x1Ann)。	UINT16	RW	–
F800:06	可用的 Rx PDOs	指示配置的 RxPDO 对象的最大编号 (0x16nn)。	UINT16	RW	–
F800:07	可用的 Tx Frames	指示配置的 TxFrame 对象的最大编号 (0x8nnn)。	UINT16	RO	–
F800:08	设备周期	指示操作 EAP 设备的周期。 过程数据周期（例如 0xDnnn:07）仅可采用该值的整数倍。	UINT32	RO	–

索引 F801 位图

索引	名称	含义	数据类型	标记	缺省
F801:0	所用元素编号	位图对象中的条目编号	UINT8	RW	0x06 (6_{dec})
F801:01	索引位图 Tx Var	现有 TxVariable 对象的位代码映射。 如果设置了位 n，则存在索引 0x 6000 + n。	八进制字符串 [512]	RW	–
F801:02	索引位图 Rx Var	现有 RxVariable 对象的位代码映射。 如果设置了位 n，则存在索引 0x 7000 + n。	八进制字符串 [512]	RW	–
F801:03	索引位图 Tx 过程数据	现有 TxProcessData 对象的位代码映射。 如果设置了位 n，则存在索引 0x D000 + 4*n。	八进制字符串 [128]	RW	–
F801:04	索引位图 RxProcess 数据	现有 RxProcessData 对象的位代码映射。 如果设置了位 n，则存在索引 0x E000 + 4*n。	八进制字符串 [128]	RW	–
F801:05	索引位图 Tx PDOs	现有 TxPDO 对象的位代码映射。 如果设置了位 n，则存在索引 0x1A00 + n。	八进制字符串 [64]	RW	–

索引	名称	含义	数据类型	标记	缺省
F801:06	索引位图 Rx PDOs	现有 RxPDO 对象的位代码映射。 如果设置了位 n，则存在索引 0x1600 + n。	八进制字符串 [64]	RW	-

索引 F920 AoE 设置

索引	名称	含义	数据类型	标记	缺省
F920:0	所用元素编号	AoE 设置对象中的条目编号	UINT8	RW	0x05 (5_{dec})
F920:01	本地 AoE NetID	EAP 设备的本地 AoE NetID	八进制字符串 [6]	RW	-
F920:02	路由器 NetID	关联的 AoE 路由器的 AoE NetID	八进制字符串 [6]	RO	-
F920:03	本地 MAC 地址	此 EAP 设备所用网卡的本地 MAC 地址。	八进制字符串 [6]	RO	-
F920:04	本地 IP 地址	此 EAP 设备所用相应网卡的本地 IP 地址。	UINT32	RW	-
F920:05	本地端口名称	EAP 设备及其 AoE 端口注册到 TwinCAT ADS 路由器的名称	字符串 [31]	RW	EtherCAT 自动化协议

动态对象

索引 6000-6FFF TxVariables

索引	名称	含义	数据类型	标记	缺省
6000+n:0	所用元素编号	TxVariable 对象中的条目编号	UINT8	RW	0x22 (34_{dec})
6000+n:01	大小	以位为单位的数据长度（分索引 2）	UINT16	RW	-
6000+n:02	数据	变量的当前数据	八进制字符串 [尺寸/8]	RO	-
6000+n:03	名称	变量名	字符串 [256]	RW	VarData
6000+n:04	类型	作为 GUID 的对象的数据类型	GUID	RW	-
6000+n:05	保留	-	UINT32	RW	-
6000+n:29	符号名称	应用中链接变量的符号名称（例如 PLC 任务 - PLC-Projectname.MAIN.iCounter）	字符串 [256]	RW	-
6000+n:30	AoE 地址	Octet 7..2：AoE NetID Octet 1..0：AoE 端口 包含当前过程变量的对象字典	八进制字符串 [8]	RW	-
6000+n:32	映像配置	代码指示过程映像的哪些输入/输出变量属于该对象	UINT32	RO	-
6000+n:33	数据偏移量	输出过程映像中的字节偏移量	UINT32	RO	-
6000+n:34	保留	-	UINT32	RO	-

索引 7000-7FFF Rx 变量

索引	名称	含义	数据类型	标记	缺省
7000+n:0	所用元素编号	RxVariable 对象中的条目编号	UINT8	RW	0x22 (34_{dec})
7000+n:01	大小	以位为单位的数据长度（分索引 2）	UINT16	RW	-

索引	名称	含义	数据类型	标记	缺省
7000+n:02	数据	变量的当前数据	八进制字符串 [尺寸/8]	RW	-
7000+n:03	名称	变量名	字符串 [256]	RW	VarData
7000+n:04	类型	作为 GUID 的对象的数据类型	GUID	RW	-
7000+n:05	保留		UINT32	RW	-
7000+n:29	符号名称	应用中链接变量的符号名称（例如 PLC 任务 - PLC-Projectname.MAIN.iCounter）	字符串 [256]	RW	-
7000+n:30	AoE 地址	Octet 7..2：AoE NetID Octet 1..0：AoE 端口 包含当前过程变量的对象字典	八进制字符串 [8]	RW	-
7000+n:32	映像配置	代码指示过程映像的哪些输入/输出变量属于该对象	UINT32	RO	-
7000+n:33	保留	-	UINT32	RO	-
7000+n:34	数据偏移量	输出过程映像中的字节偏移量	UINT32	RO	-

索引 8000-8FF8 TxFrame

索引	名称	含义	数据类型	标记	缺省
8000+n*8:0	所用元素编号	TxFrame 对象中的条目编号	UINT8	RW	0x32 (50_{dec})
8000+n*8:03	名称	帧名称	字符串 [256]	RW	-
8000+n*8:04	设备类型	子配置文件类型（与对象 0xF020-0xF022 中的相应条目完全相同）	UINT32	RO	0x03E8 (1000_{dec})
8000+n*8:05	目标地址供应商 ID	用于对等通讯： 0 = 未用 轮询连接： 通讯合作伙伴的供应商 ID 推送连接： 未用	UINT32	RW	-
8000+n*8:06	目标地址产品代码	用于对等通讯： 0 = 未用	UINT32	RW	-
8000+n*8:07	目标地址修订编号	用于对等通讯： 0 = 未用	UINT32	RW	-
8000+n*8:08	目标地址序列号	用于对等通讯： 0 = 未用	UINT32	RW	-
8000+n*8:30	目标 AMS NetID	AoE NetID（订阅服务器网络 ID） 如果数值不为 0，则目标地址 SI 32 和 SI 33 必须具有数值 0。	八进制字符串 [6]	RW	-
8000+n*8:31	网关 IP 地址	如果 SI 33 不具有数值 0，则必须设置标准网关 IP 地址。	UINT32	RW	-
8000+n*8:32	目标 MAC 地址	MAC 地址 如果数值不为 0，则目标地址 SI 30 和 SI 33 必须具有数值 0。 MAC 地址可以是单播、多播或广播地址。	八进制字符串 [6]	RW	01 01 05 04 00 00

索引	名称	含义	数据类型	标记	缺省
8000+n*8:33	目标 IP 地址	IP 地址 如果数值不为 0，则目标地址 SI 30 和 SI 32 必须具有数值 0。 IP 地址可以是单播、多播或广播地址。	UINT32	RW	–
8000+n*8:34	VLAN 信息	VLAN 信息由以下域组成： 位 0-15：VLan 类型 (81 00) 位 16-18：优先级 位 19：保留 位 20-31：VLan ID 如果数值为 0，则不使用 VLAN 标题	UINT32	RW	0x00000000 (0_{dec})
8000+n*8:35	订阅服务器监控	如果数值为 1，则 ARP 请求会定期发送到已配置的目标地址，以确保收件人仍在回复。否则将停止发送 TxFrame。 订阅服务器监控只能与单播通讯一起使用。	UINT8	RW	0x00 (0_{dec})
8000+n*8:36	可更改目标	如果可更改目标具有数值 0，则过程映像中的目标地址不显示任何变量。否则，在显示目标地址时采用以下映射： 1：目标 MAC 地址 2：目标 AMS NetID 3：目标 IP 地址	UINT8	RO	0x00 (0_{dec})
8000+n*8:37	监控重试周期	废弃	UINT32	RO	–
8000+n*8:38	监控重试周期	如果 SI 35 = 0x01，则在等待时间 (以 µs 为单位) 之后发送新的 ARP 请求。	UINT32	RW	0xF4240 (1000000_{dec})
8000+n*8:39	帧控件	位 0 = 1：停止发送 TxFrame 位 1 = 1：目标 MAC 地址已从 ARP 高速缓冲存储器中删除	UINT16	RW	0x0000 (0_{dec})
8000+n*8:40	帧状态	位 0 = 1：TxFrame 未发送 位 1 = 1：错误 (帧过大) 位 2 = 1：订阅服务器不再应答 (仅当 SI 35 = 0x01 时)	UINT16	RO	0x0000 (0_{dec})
8000+n*8:48	控件符号名称	应用中已链接变量的符号名称 (例如 PLC 任务)	字符串 [256]	RW	–
8000+n*8:49	状态符号名称	应用中已链接变量的符号名称 (例如 PLC 任务)	字符串 [256]	RW	–
8000+n*8:50	目标地址符号名称	应用中已链接变量的符号名称 (例如 PLC 任务) 如果设置了符号名称，则会在过程映像中显示配置的目标地址，并相应地设置 SI 36。	字符串 [256]	RW	–

索引 8001-8FF9 TxProcessData 分配对象

索引	名称	含义	数据类型	标记	缺省
8001+n*8:0	所用元素编号	TxPD 分配对象中的条目编号	UINT8	RW	#(分索引)
8001+n*8 :01-255	条目 n	1. -255. TxFrame 的 TxProcessData	UINT16	RW	–

索引 8002-8FFA TxFrame 信息

索引	名称	含义	数据类型	标记	缺省
8002+n*8:0	所用元素编号	TxFrameInfo 对象中的条目编号	UINT8	RW	0x21 (33_{dec})
8002+n*8:01	映像配置	代码指示过程映像的哪些输入/输出变量属于该对象 低位字 = 输入过程映像 位 0 = 1：状态 高位字 = 输出过程映像 位 0 = 1：控件 位 1 = 1：目标 MAC 地址 位 2 = 1：目标 AMS NetID 位 3 = 1：目标 IP 地址	UINT32	RO	0x00010001 (65537_{dec})
8002+n*8:02	控件偏移量	输出过程映像中的字节偏移量	UINT32	RO	–
8002+n*8:03	状态偏移量	输入过程映像中的字节偏移量	UINT32	RO	–
8002+n*8:04	NetID 偏移量	输出过程映像中的字节偏移量	UINT32	RO	–
8002+n*8:32	IoDivMod	分隔符/模数值定义了发送下一个 TxFrame 之前的周期等待时间。 位 0-7（分隔符）：等待的周期数 位 8-15（模数）：指定从其开始计算的开始周期	UINT16	RW	0x0000 (0_{dec})
8002+n*8:33	CoE 索引	用于未来用途	UINT16	RW	–

索引 D000-DFFC TxProcessData

索引	名称	含义	数据类型	标记	缺省
D000+n*4:0	所用元素编号	TxPD 对象中的条目编号	UINT8	RW	0x22 (34_{dec})
D000+n*4:01	名称	帧名称	字符串[256]	RW	–
D000+n*4:02	PDO 编号	PDO 编号定义了已分配 TxPDO 的对象索引	UINT16	RW	–
D000+n*4:03	过程数据 ID	PD ID 定义了 0…65535 范围内的一个值，该值必须明确位于通讯网络内。 此 ID 是过程数据帧标题的一部分。	UINT16	RW	–
D000+n*4:04	版本	该版本是 0…65535 范围内的一个值，并且一旦对此 TxPD 做出更改（例如对另一个 TxPDO 的引用），该值应一致地递增。 此版本是过程数据帧标题的一部分。	UINT 16	RW	–
D000+n*4:05	CoS 改变时周期	废弃，参见分索引 8。	UINT16	RO	–
D000+n*4:06	CoS 禁止时间	禁止时间指定了时间间隔（以 µs 为单位），在该时间间隔内，即使已分配 PDO 的过程变量值已改变，也不会再次发送 TxPD。 如果值为 0，则不禁止传输 TxPD。 如果值 > 0，则分索引 8（CoS 改变时超时）的值也必须 > 0；但是，分索引 7 和 10 的值必须为 0。	UINT32	RW	–

索引	名称	含义	数据类型	标记	缺省
D000+n*4:07	周期	周期定义了周期发送 TxPD 的时间间隔(以 µs 为单位)。如果周期值大于 0，则分索引 6、8 和 10 必须为 0。 如果值为 0，则根本不会发送 TxPD。	UINT 32	RW	–
D000+n*4:08	CoS 改变时超时	改变时超时指定时间间隔的最大持续时间(以 µs 为单位)，在该时间内不会发送 TxPD，除非在该时间内，已分配 PDO 的过程变量值发生改变。 如果值为 0，则在状态改变时不发送过程变量。 如果值 > 0，则分索引 7 和 10 的值必须为 0。	UINT32	RW	–
D000+n*4:10	轮询请求 Rx PD	轮询请求 RxPD 定义了 RxProcessData 的对象索引，一旦定义的 RxPD 接收到新值，该对象索引将触发此 TxPD 的传输。然后，TxPD 在轮询数据交换模式中作为服务器运行。 如果值为 0，则停用轮询数据交换模式。 如果值 > 0，则分索引 6、7 和 8 的值必须为 0。	UINT16	RW	–
D000+n*4:11	过程数据控制	位 0 = 1：停止传输 TxPD	UINT16	RW	0x0000 (0_{dec})
D000+n*4:12	过程数据状态	位 0 = 1：未发送 TxPD	UINT16	RO	–
D000+n*4:32	控件符号名称	应用中已链接变量的符号名称 (例如 PLC 任务)	字符串 [256]	RW	–
D000+n*4:33	状态符号名称	应用中已链接变量的符号名称 (例如 PLC 任务)	字符串 [256]	RW	–
D000+n*4:34	ID 符号名称	应用中已链接变量的符号名称 (例如 PLC 任务)	字符串 [256]	RW	–

索引 D002-DFFE TxProcessData 信息

索引	名称	含义	数据类型	标记	缺省
D002+n*4:0	所用元素编号	TxPDInfo 对象中的条目编号	UINT8	RW	0x20 (32_{dec})
D002+n*4:01	映像配置	代码指示过程映像的哪些输入/输出变量属于该对象 低位字 = 输入过程映像 位 0 = 1：状态 高位字 = 输出过程映像 位 0 = 1：控件 位 1 = 1：过程数据 ID	UINT32	RO	0x00010001 (65537_{dec})
D002+n*4:02	控件偏移量	输出过程映像中的字节偏移量	UINT32	RO	–
D002+n*4:03	状态偏移量	输入过程映像中的字节偏移量	UINT32	RO	–
D002+n*4:04	ID 偏移量	输出过程映像中的字节偏移量	UINT32	RO	–
D002+n*4:32	IoDivMod	分隔符/模数值定义了再次发送 TxPD 之前的周期等待时间。 位 0-7 (分隔符)：等待的周期数 位 8-15 (模数)：指定从其开始计算的开始周期	UINT16	RW	0x0000 (0_{dec})

索引 E000-EFFC RxProcessData

索引	名称	含义	数据类型	标记	缺省
E000+n*4:0	所用元素编号	RxPD 对象中的条目编号	UINT8	RW	0x25 (37_{dec})
E000+n*4:01	名称	帧名称	字符串 [256]	RW	–
E000+n*4:02	PDO 编号	PDO 编号定义了已分配 RxPDO 的对象索引	UINT16	RW	–
E000+n*4:03	过程数据 ID	PD ID 定义了 0…65535 范围内的一个值，该值与接收的过程数据 ID 匹配。	UINT16	RW	–
E000+n*4:04	版本	该版本是 0…65535 范围内的一个值，并且一旦对此 RxPD 做出更改（例如对另一个 RxPDO 的引用），该值应一致地递增。	UINT 16	RW	–
E000+n*4:05	忽略版本	如果值为 0，则根据分索引 4 的版本检查接收的过程数据版本（散列值）。 如果值为 1，则停用版本检查。	UINT8	RW	0x00 (0_{dec})
E000+n*4:06	发布服务器 NetID	发布服务器 NetID 的定义。只有当 EAP 报文从具有此 NetID 的发送器发送时，才会对其进行处理。 如果发布服务器 NetID 的值为 0，则停用此过滤器。	八进制字符串 [6]	RW	00 00 00 00 00 00
E000+n*4:07	MAC 地址	可以定义一个使用 NIC（网络接口卡）作为多播信息包接收过滤器的多播 MAC 地址。 如果值为 0，则停用过滤器功能。	八进制字符串 [6]	RW	01 01 05 04 00 00
E000+n*4:08	IP 地址	可以定义一个使用 NIC（网络接口卡）作为多播信息包接收过滤器的多播 IP 地址。 如果值为 0，则停用过滤器功能。	UINT32	RW	0x00000000 (0_{dec})
E000+n*4:09	更新时间	更新时间用于指定时间间隔（以 µs 为单位），在该时间间隔内必须接收新的过程数据。 如果值为 0，则停用此机制。	UINT32	RW	0x00000000 (0_{dec})
E000+n*4:10	轮询请求 TxPD	轮询请求 TxPD 定义了作为请求发送的 TxProcessData 的对象索引，以便接收带有适当过程数据的 EAP 报文。 然后，TxPD 在轮询数据交换模式中作为服务器运行。 如果值为 0，则停用轮询数据交换模式。	UINT16	RW	0x0000 (0_{dec})
E000+n*4:11	过程数据控制	位 0 = 1：停用该版本编号或散列值的检查。	UINT16	RW	0x0000 (0_{dec})
E000+n*4:12	过程数据状态	位 0 = 1：接收到具有无效版本编号（散列值）的过程数据 位 1 = 1：接收到具有无效长度的过程数据 位 2 = 1：超出超时轮询响应	UINT16	RO	–
E000+n*4:13	过程数据质量	该质量指示自上次更新此 RxProcessData（也就是说自上次接收到数据以来）的时间（以 100 µs 为单位）	UINT16	RO	–
E000+n*4:14	过程数据周期索引	接收到有效过程数据时，会从 EAP 报文中为周期索引分配已发送周期索引（参见过程数据帧标题）	UINT16	RO	–

索引	名称	含义	数据类型	标记	缺省
E000+n*4:32	控件符号名称	应用中已链接变量的符号名称（例如 PLC 任务）	字符串 [256]	RW	–
E000+n*4:33	状态符号名称	应用中已链接变量的符号名称（例如 PLC 任务）	字符串 [256]	RW	–
E000+n*4:34	ID 符号名称	应用中已链接变量的符号名称（例如 PLC 任务）	字符串 [256]	RW	–
E000+n*4:35	质量符号名称	应用中已链接变量的符号名称（例如 PLC 任务）	字符串 [256]	RW	–
E000+n*4:36	周期索引符号名称	应用中已链接变量的符号名称（例如 PLC 任务）	字符串 [256]	RW	–
E000+n*4:37	超时轮询响应	指定必须接收到对轮询请求响应的最大时间间隔（以 µs 为单位）。 如果值 > 0 并且在发送轮询请求后该超时已过期，则在 PD 状态（分索引 12）中设置位 2。 如果值为 0，则停用此监控。	UINT32	RW	0x00000000 (0_{dec})

索引 E002-EFFE RxProcessDataInfo

索引	名称	含义	数据类型	标记	缺省
E002+n*4:0	所用元素编号	RxPD 信息对象中的条目编号	UINT8	RW	0x06 (6_{dec})
E002+n*4:01	映像配置	代码指示过程映像的哪些输入/输出变量属于该对象 低位字 = 输入过程映像 位 0 = 1：状态 高位字 = 输出过程映像 位 0 = 1：控件/周期索引 位 1 = 1：过程数据 ID	UINT32	RO	0x00010001 (65537_{dec})
E000+n*4:02	控件偏移量	输出过程映像中的字节偏移量	UINT32	RO	–
E002+n*4:03	状态偏移量	输入过程映像中的字节偏移量	UINT32	RO	–
E002+n*4:04	ID 偏移量	输出过程映像中的字节偏移量	UINT32	RO	–
E002+n*4:05	质量偏移量	输入过程映像中的字节偏移量	UINT32	RO	–
E002+n*4:06	周期索引偏移量	输入过程映像中的字节偏移量	UINT32	RO	–

6.2 EAP 设备的 TwinCAT ADS 接口

TwinCAT EAP 设备为其他通讯合作伙伴（例如虚拟现场设备或 Windows 程序）提供 TwinCAT *ADS/AMS* 接口，并充当 *ADS/AMS* 服务器。*ADS* 表示*自动化设备规范*。它描述了独立于设备和现场总线的接口。*AMS* 表示*自动化信息规范*，并启用待寻址的中央和分散系统（例如 PC 或总线控制器）。*ADS/AMS* 由 Beckhoff 指定，并由 TwinCAT 路由器支持。在计算机边界外的网络中发送的消息通过 *TCP/IP* 传送。

也可以通过该接口使用 *CANopen* 通讯通道 *SDO*（*服务数据对象*）。*SDO* 的主要用途是读取和写入 *CANopen 对象字典*（*OD*）的参数。*SDO* 的传输作为已确认数据、以两个通讯合作伙伴之间点对点连接的形式进行，并嵌入 ADS 中：

使用端口 0xFFFF 和 EAP 设备（“协议”标签）NetID 上的 *ADS 读取*或 *ADS 写入*指令，分别读取或写入 *CANopen OD* 的参数或 *SDO* 描述。如表中所示，通讯通道 *CANopen SDO* 嵌入 *ADS* 协议中，如下所示：

CANopen SDO 通讯	ADS 指令	索引组	索引偏移量	含义
SDO 上传	读取	0xF302	SDO 的索引和分索引。 位 16-31：索引 位 8：完成访问 位 0-7：分索引 采样： 0x16010001： 索引 = 0x1601 完成访问 = 0 分索引 = 1 如果完成访问 = 1，则分索引没有意义。	数据类型：UINT8[n] SDO 上传请求： 该对象根据索引偏移量进行寻址，并可借助 ADS 读取来读取其内容。
SDO 下载	写入	0xF302	SDO 的索引和分索引。 位 16-31：索引 位 8：完成访问 位 0-7：分索引 采样： 0x16010001： 索引 = 0x1601 完成访问 = 0 分索引 = 1 如果完成访问 = 1，则分索引没有意义。	数据类型：UINT8[n] SDO 下载请求： 该对象根据索引偏移量进行寻址，并可借助 ADS 写入来写入其内容。
SDO 信息 获取对象列表	读取	0xF3FC	位 16-31：列表类型 采样： 0x00000000：返回各列表类型所有现有对象的编号 0x00010000：返回指定列表类型所有对象的索引	返回属于索引偏移量中指定列表类型的对象的索引 可能的列表类型为： ALL_OBJECTS = 1 RXPD_OBJECTS = 2 TXPD_OBJECTS = 3 BACKUP_OBJECTS = 4 SETTING_OBJECTS = 5 数据类型：UINT16[6] 如果列表类型 = 0000： 元素 = 0：列表类型编号 元素 > 0：属于第 n 个列表类型的现有对象编号 数据类型：UINT16[n] 如果列表类型 > 0000： 元素 n=0：属于此列表类型的现有对象数量加一。 元素 n>0：属于此列表类型的第 n 个对象索引
SDO 信息 获取对象描述	读取	0xF3FD	位 16-31：索引	读取具有指定索引的完整对象的 SDO 描述。
SDO 信息 获取条目描述	读取	0xF3FE	位 16-31：索引 位 0-7：分索引	读取具有指定分索引的单个条目的 SDO 描述，该分索引属于具有指定索引的对象。

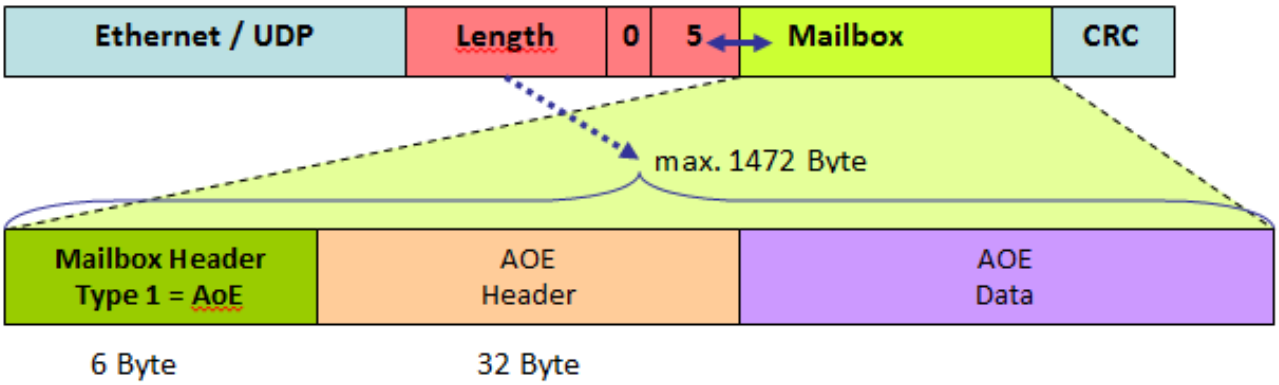
6.3 ADS over EtherCAT (AoE)

TwinCAT EAP 设备也支持 *AoE* 协议。AoE 协议规范可在 EtherCAT 协议增强版 (ETG 1020) 中找到。ADS/AMS 通讯和 AoE 通讯之间的区别在于，与 ADS/AMS 通讯相比，AoE 通讯不需要 TwinCAT 路由器。AoE 协议是类别信箱通讯下分类为 TwinCAT 的协议之一。类型 5 (信箱通讯) 的 EtherCAT 报文用于 AoE 通讯。可以从 TwinCAT EAP 设备或向其发送信箱报文：

通过以太网（以太网类型 = 0x88A4）或

通过 UDP/IP（UDP 端口 = 0x88A4）

可通过信箱通讯传输（通道）各种协议。根据*信箱标题*中的域类型定义要通过通道传输的协议。AoE 协议由数值 1 指定（参见下图中的*信箱标题*）。*信箱标题*后紧跟着 *AoE 标题*，然后是 *AoE* 数据。其结构与 *ADS/AMS* 协议（另请参考 Beckhoff 信息系统中的 TwinCAT *ADS/AMS* 规范）完全相同。这增加了通过信箱协议使用 *CANopen SDO* 通讯的可能性。类似于 *ADS/AMS* 协议，可按照章节 EAP 设备的 TwinCAT ADS 接口 [▶ 57]中的描述访问 TwinCAT EAP 设备的 *CANopen OD*。



采样：
为了从 EAP 对象字典中读取标识对象的*供应商 ID*，可基于*指令 ID* 在 *AoE 标题*中定义 *ADS* 指令。

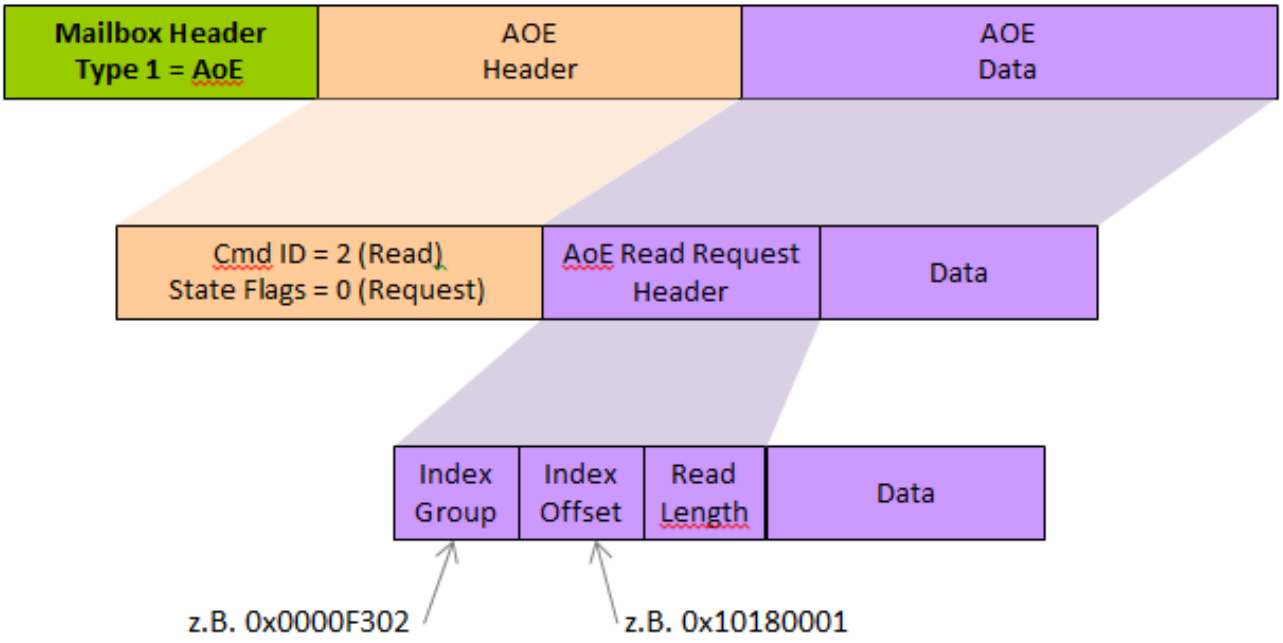
AoE 指令	指令 ID	描述
ADS 读取	2	用于读取数据的 ADS 读取指令
ADS 写入	3	用于写入数据的 ADS 写入指令

根据 *AoE 标题状态标记*中的第 0 位，对*请求 = 0* 和*响应 = 1* 进行区分。随后，将在 *AoE 数据*域中填充以下内容（参见下图）：

开头有一个 *ADS 读取请求标题*，其中包含

- 索引组 = 0x0000F302，用于读取参数内容，
- 索引偏移量 = 0x10180001，用于寻址*供应商 ID* 参数，以及
- 读取长度 = 4，用于提示可用数据缓冲区的尺寸。

其后面是待读取数据的数据缓冲区。



6.4 TwinCAT EAP 设备的在线配置

CANopen OD 提供的巨大优势是，可以通过标准化接口对 TwinCAT EAP 设备进行几乎无限的在线配置。执行在线配置意味着 TwinCAT EAP 设备已启用，并且需要在更改配置之前切换到安全状态。在安全状态下不会进行过程数据通讯。

通过设置参数完成配置，并

- 通过 TwinCAT 实现配置：
  然后直接在在线对象字典中更改这些数值。
  这对于系统调试很有用。通过单击要参数化的相应索引行并在 *Set Value（设置数值）* 对话框中输入合适的数值，可以在相应参数中设置新数值。
- 或通过 ADS 或 AoE 的应用实现：
  建议在系统运行时或没有可用 TwinCAT 或操作员时进行更改。

一方面，更改可能只涉及现有 *OD* 对象中的条目。另一方面，更改可能导致生成新对象或删除现有对象。

更改现有对象

仅涉及现有对象的更改适应当前操作特性（例如传输周期、目标地址等），或为了更好地理解为单个对象分配目的相关的名称。此更改所需步骤可通过一个实例来说明：

更改发送发布服务器变量的时间间隔。

假设 EAP 设备配置有 *TxPD*（0xD000），其中将 10000 µs 的间隔配置为*周期*，并且现在将此间隔增加至 30000 µs。这种情况下，必须执行以下步骤：

1. 将 EAP 设备置于安全状态。
   设置对象条目 0xF200:01（*控制字*）为数值 2（=操作前）
   检查对象条目 0xF100:01 以确定状态是否为 2（=操作前）
   多次执行此检查（轮询），超时时间为，例如 200 ms。
2. 将 TxPD 的时间间隔更改为新数值
   将 *TxPD*（0xD000:07）的对象条目*周期*设置为数值 30000
3. 再次将 EAP 设备置于操作状态。
   设置对象条目 0xF200:01（*控制字*）为数值 4（=安全操作）
   检查对象条目 0xF100:01 以确定状态是否为 4（=安全操作）
   多次执行此检查（轮询），超时时间为，例如 200 ms。
   设置对象条目 0xF200:01（*控制字*）为数值 8（=操作）
   检查对象条目 0xF100:01 以确定状态是否为 8（=操作）
   多次执行此检查（轮询），超时时间为，例如 200 ms。

生成及删除对象

生成新对象可用于扩展现有通讯连接或添加新通讯连接。例如，此功能可以在两个控制器之间建立临时通讯连接。在配置过程中，仅短暂中断 EAP 设备的传输和接收。控制器其他组件和模块的过程不受此影响，并始终保持启用。

借助对象*帧列表*（索引 0xF020 – 0xF022）和*位图*（索引 0xF801）生成或删除对象
*帧列表*对象中共有 512 个条目 – 正好与 *TxFrame* 对象中的最大可能数量一样多（参见 EAP 对象字典（子配置文件 1000）[▶ 46] 中的索引 0xF020）。因此，每个条目都精确分配一个 *TxFrame* 对象。然后，将该数值 1000（子配置文件数）保存到每个现有 *TxFrame* 对象的相应条目中。所有其他条目的数值均为 0。假设存在三个具有索引 0x8000、0x8008 和 0x87F0 的 *TxFrame*。然后，该分配应理解为：

编号	帧列表条目	数值	分配的 TxFrame 对象
1	0xF020:01	1000	0x8000
2	0xF020:02	1000	0x8008
…	…	0	…
254	0xF020:254	0	0x87E8
255	0xF021:01	1000	0x87F0
…	…	0	…
512	0xF022:04	0	0x8FF8

条目值设置为 1000 时，将立即生成相应的 *TxFrame* 对象。该值复位为 0 时，将再次删除该对象。

除 *TxFrame* 外，对象*位图*中的每个动态对象类型都存在一个独立的条目。因此，将找到 6 个条目（条目 0 除外 – 参见 EAP 对象字典（子配置文件 1000）[▶ 46] 中的索引 0xF801）。条目包含足够长的位域，以便能够保存单目符号中相应对象类型的最大数目。

假设存在三个具有索引 0xD000、0xD00C 和 0xD014 的 *TxPD* 对象。然后，将总长度为 1024 位的位序列 10010100000000…0 保存到条目 0xF801:03 中。该序列解释如下：

位 1	位 2	位 3	位 4	位 5	位 6	位 7	位 8		位 1024
1	0	0	1	0	1	0	0	…	0
0xD000	0x D004	0x D008	0x D00C	0x D010	0x D014	0x D018	0x D01C	…	0x DFFC

位设置为 1 时，将立即生成相应的对象。复位时，将再次删除该对象。

注意

删除离线配置的对象。

也可以删除调试前通过 TwinCAT（离线配置）配置的对象，从而在激活控制器时已生成对象。在这种情况下，必须考虑以下几点：

某些对象管理位于 EAP 设备过程映像中的变量（例如*循环次数*、*质量*、*状态字*和*控制字*等）。显示在线值时，TwinCAT 通过 *ADS* 访问过程映像。配置后，将计算借助 TwinCAT 生成并激活的过程映像中的各个变量地址。

如果在线配置时删除了离线配置的对象，则此更改对 TwinCAT 保持隐藏。因此，其仍然访问计算的地址，以便在在线视图中显示内容。此处应明确，显示的在线数值不再与 TwinCAT 中生成的配置相匹配。

因此，这种情况下，在线视图或各个过程映像的地址不再是监控或甚至是评估过程映像值的有效参考源。

建议取消激活但不删除离线配置的对象。

如果借助 TwinCAT 删除或添加了对象，则新对象不会直接显示在对象字典中。已删除对象保留在 *OD* 中，并且不再显示其条目的数值。为了显示新对象且不再显示已删除对象，必须在 TwinCAT 中再次读取 *OD* 的结构。这可以借助 *Advanced Settings（高级设置）* 对话框完成（参见章节 TwinCAT EAP 设备 [▶ 37]中的图片 *CoEOnline*）。

激活/取消激活对象

如果某些*过程数据/PDO* 或完整*帧*仅从现有 EAP 通讯中暂时删除，则建议取消激活相关对象而不是将其删除。

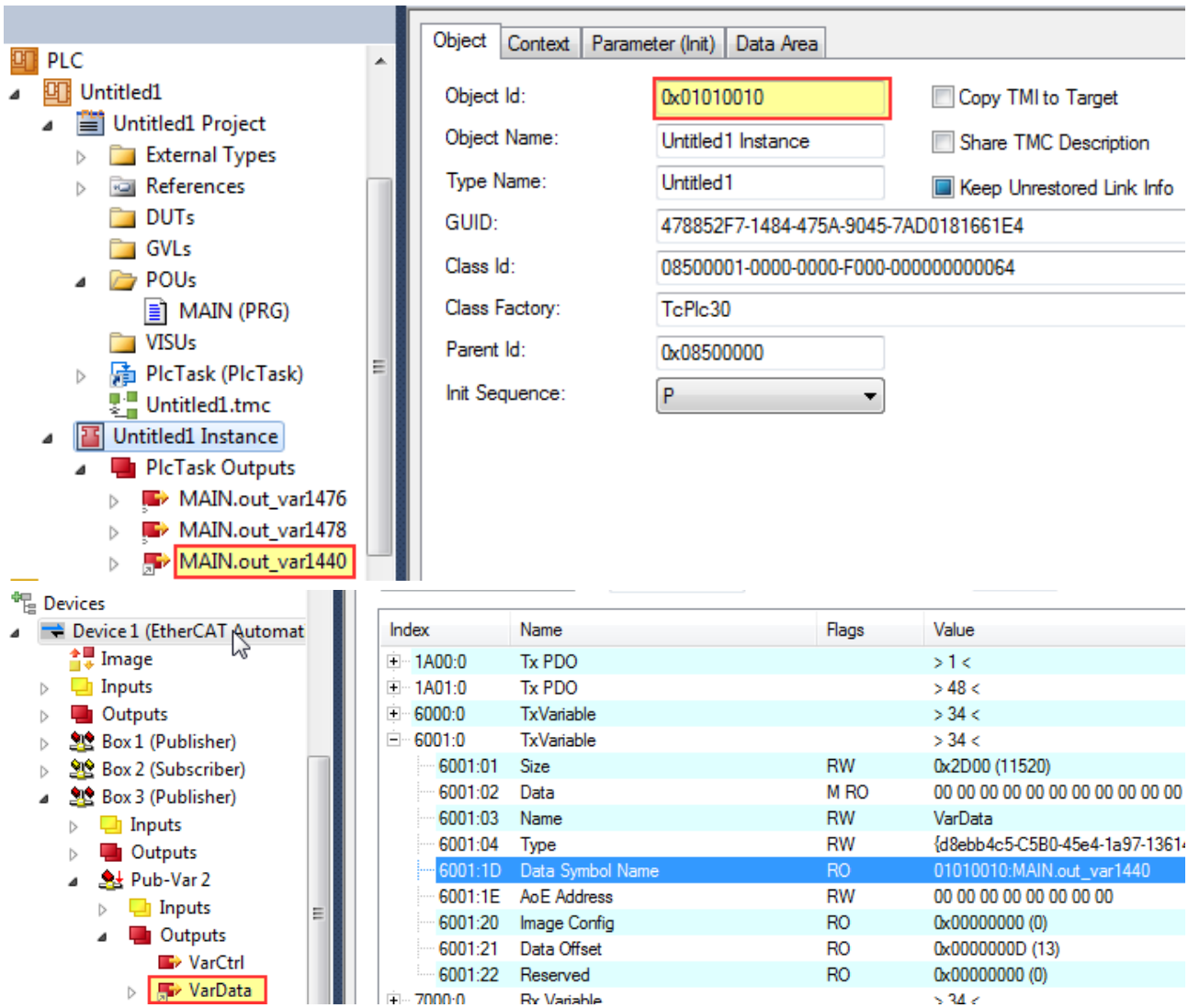
可以通过将其分索引 0 设置为数值 0 来取消激活和重新激活对象 *TxFrame*（0x8000）、*TxPD 分配*（0x8001）、*TxPD*（0xD000）、*TxPDO*（0x1A00）、*RxPD*（0xE000）和 *RxPDO*（0x1600）。然后通过设置原始数值或任何其他大于 0 的数值来进行激活。

基于列表的对象

请注意，基于列表的对象 *TxPD 分配*和 *Rx/TxPDO* 的条目始终包含对其他对象的参考。因此，对分索引 0 所做的任何更改都将影响实际引用的其他对象及其数量。

将 EAP 过程映像的变量与 PLC 变量链接。

TwinCAT I/O 层的变量与 PLC 程序变量的链接在 TwinCAT 中作为标准进行。如果将 PLC 程序的*输入/输出变量*与 EAP 设备的*发布服务器或订阅服务器变量*链接，则在 EAP 设备的相应 *CANopen* 对象中输入此变量的符号名称（参见下图）。



如上图所示，符号名称由 PLC 实例的*对象 ID (OID)* 和 PLC 程序变量的唯一名称组成，用冒号分隔。在线配置时，也可以使用同样的语法为 EAP 对象输入符号名称。然后，将 PLC 变量与 *Rx/TxVariable* 相应地链接。

离线链接的在线配置

离线配置期间已经生成的链接在在线配置期间既不能更改也不能删除。因此，这些符号名称用*只读 (RO)* 标记进行标记。

原因：
创建配置时，会借助映射对象由 TwinCAT 定义这些变量的链接。但是无法在操作期间更改此映射对象。

通过单次访问/完全访问来访问 CANopen 对象字典

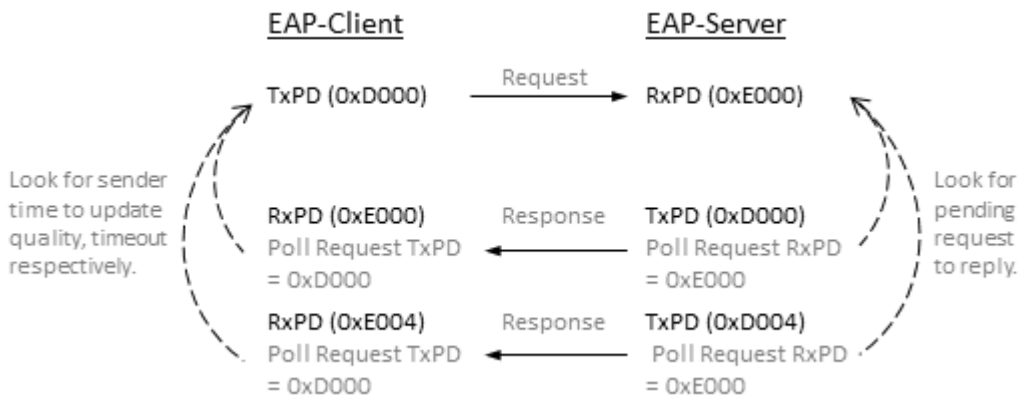
在通讯通道 *CANopen SDO*（参见 EAP 设备的 TwinCAT ADS 接口 [▶ 57]）的支持下，*ADS 接口*允许分别读取和写入单个对象条目的数值和完整对象的数据。关于访问单个对象条目（*单次访问*）的说明，参见章节 TwinCAT EAP 设备的在线配置 [▶ 60]中的*更换现有对象*部分。
*完全访问*表示访问完整对象。在这种情况下，所有对象条目的数值均以二进制在一个块中连续发送。为了安全起见，在启用访问之前，将检查块长度与对象尺寸的相关性。为了确定必要的块长度，可以参考对象字典文档，或将离线配置导出至 *EAP 设备配置*（EDC）文件（参见章节 TwinCAT EAP 设备 [▶ 37]中的“协议”部分）。后者是 XML 格式文件，其中包含当前配置的完整对象字典（参见章节 EAP 设备配置（EDC）文件 [▶ 65]）。

6.5 轮询数据交换的配置

借助 EAP 对象字典配置轮询数据交换模式。为此，必须在一个 EAP 设备中配置要接收服务器响应的 *RxPD*（作为客户端），同时必须在另一个 EAP 设备中配置定义客户端请求响应的 *TxPD*（作为服务器）。

如下图所示，在请求的客户端上配置了一个 *TxPD*，并如同推送数据交换模式中那样周期性发送。为了接收服务器响应，此采样中在客户端上配置了两个 *RxPD*。用两个 *RxPD*，在*轮询请求* *TxPD* 属性中输入用作请求的 *TxPD* 索引。基于请求的传输时间，RxPD 可以确定质量值，根据该质量值可以读取请求和响应之间的时间差。

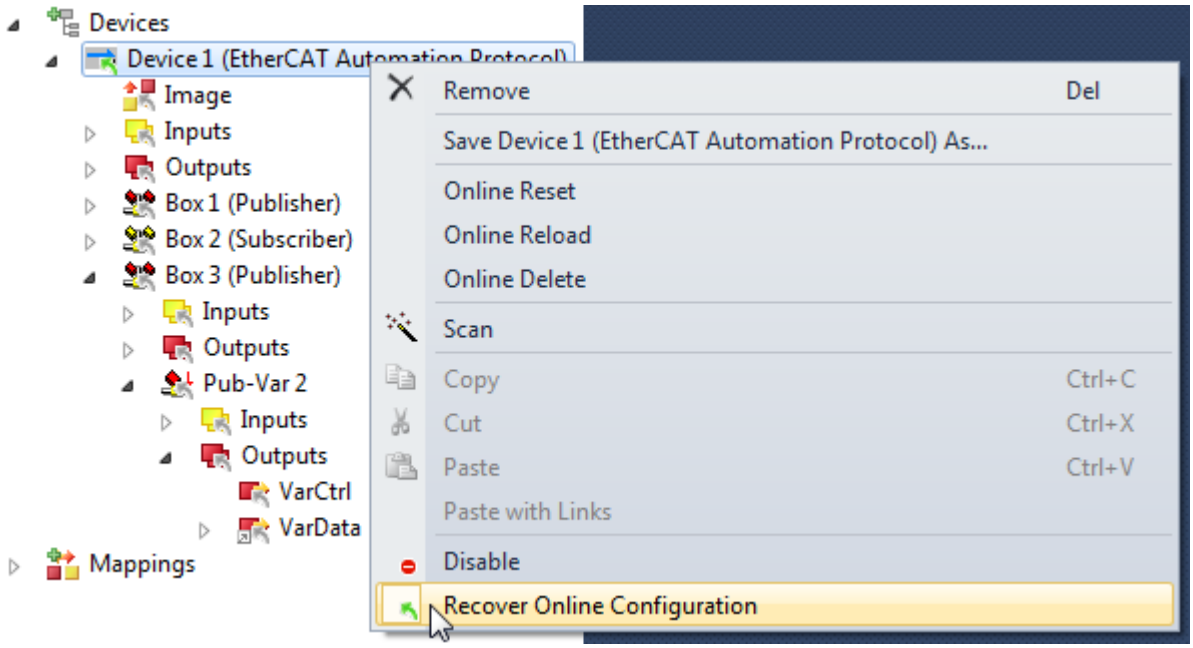
在服务器端，为了接收与推送数据交换模式工作方式相同的请求，配置了 *RxPD*。在此实例中，为了向客户端请求传输响应，配置了两个 *TxPD*。用两个 *TxPD*，在*轮询请求* *TxPD* 属性中输入用作请求收件人的 *RxPD* 索引。在每个过程周期中，*TxPD* 都会通过其配置的*轮询请求* *RxPD* 检查是否已接收到请求。如果已接收到，则在该周期中将 *TxPD* 作为响应发回。



6.6 还原在线配置

如果在线更改了 EAP 设备的 *CANopen OD* 参数，则一旦 TwinCAT 切换回*停止/配置*模式，*CANopen OD* 的当前状态将立即在硬盘上归档。在*运行*模式下重启动 TwinCAT 时，创建的归档文件通常是无关的，因为离线配置只需重启动即可。放弃所有在线配置的更改。

如果要在重启动 TwinCAT 时还原在线配置，则必须在 TwinCAT 中为此 EAP 设备打开 *Recover Online Configuration（恢复在线配置）*选项（参见下图），然后必须重新激活该项目。这种情况下，开启控制器时，首先加载离线配置，然后从硬盘加载归档文件，以便在线配置在开启过程结束时激活。



如果设置了 *Recover Online Configuration（恢复在线配置）*选项，则无法在 TwinCAT 中更改 TwinCAT EAP 设备的配置。为此，必须首先再次取消激活该选项。该阻断机制的唯一目的是保护离线配置不被客户更改，即使由于还原了在线配置，此更改在激活控制器时无效。

恢复在线配置

该归档的 *CANopen* 对象字典已与 TwinCAT 中的 EAP 设备实例关联。一旦在 TwinCAT 中删除该实例并创建新实例，将无法再为该实例还原之前归档的 *OD*。

7 EAP 设备配置（EDC）文件

借助 TwinCAT，进行 *EAP 设备配置（EDC）*。至少必须通过 TwinCAT 创建 EAP 设备本身，以便在开启 TwinCAT 时，相应地生成 EAP 设备实例。随后，也可以通过 *ADS 客户端*应用创建 *TxFrame*、*Tx/RxProcessData*、*Tx/RxPDO* 和 *Tx/RxVariable*。为此，TwinCAT 必须处于*运行*模式。但是，在多数情况下，TwinCAT EAP 设备通过 TwinCAT 完整配置，并按照章节创建 EAP 配置 [▶ 23]和配置 EAP 设备 [▶ 37]的描述开启。

具有图形用户界面（GUI）的配置器 – TwinCAT EAP 配置器（TE1610）– 已开发用于 EAP 设备在线配置或 EAP 设备整个网络。该配置器允许网络中所有 EAP 设备的图形化表达及其相互之间的通讯连接。另外，每个 EAP 设备都可以通过 GUI 进行配置。
在 TwinCAT 中创建并已导出 EDC 文件的配置包含导入配置器中以显示 EAP 设备当前配置时所需的所有信息。有关 TwinCAT EAP 配置器的文档，可在 Beckhoff 信息系统中的 *TE1610 EAP 配置器*下找到。