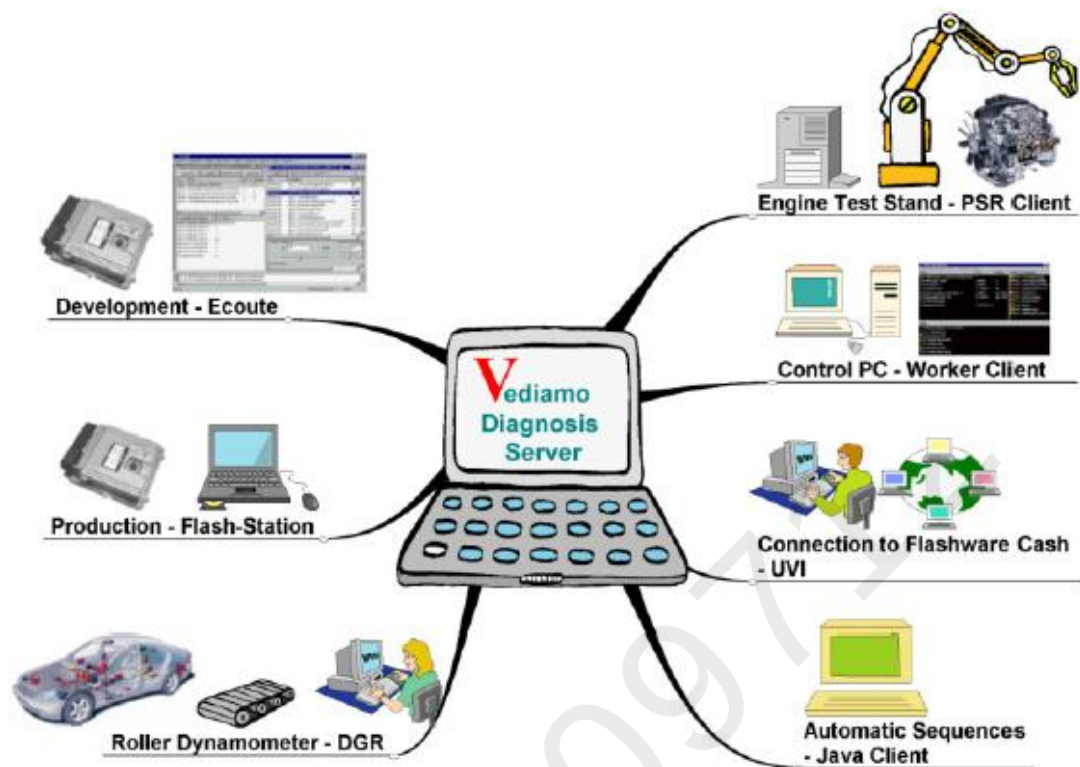


Vediamo 手册



版本	05.00
创建日期	2012 年 9 月
版权©	梅赛德斯-奔驰, PWT/VEP Werk 010, HPC H152
主页	http://diagnose.intra.daimlerchrysler.com/vediamo/
传真	+49 -(0)711 17 - 7908 1949

目录

1、引言	4
1.1 应用程序范围、重点	5
1.1.1 开发	5
1.1.2 生产	5
1.1.3 Java 程序	5
1.2 入门	6
1.2.1 基本术语	6
1.2.2 Hasty 操作说明	7
1.3 开发周期	10
1.4 硬件以及其它软件的相关信息	11
1.4.1 CAESAR	11
1.4.2 LUCA	12
1.4.3 PSR	12
1.4.4 UNIPAS	12
1.4.5 Java 运行时	12
1.5 观点	13
1.6 订购、许可证、支持	14
1.7 安装	15
1.7.1 系统需求	15
1.7.2 早期版本	15
1.7.3 安装操作步骤	15
1.7.4 安装以及检查 eCom 硬件（部件 P）	15
1.7.5 安装选项/参数	16
1.7.6 其它必要的硬件/软件	18
1.8 设计与工作模式	19
1.8.1 C/S 架构	19
1.8.2 层模型	19
1.8.3 DCOM	20
1.8.4 多客户端	20
1.8.5 多任务（多个 ECU）	21
1.8.6 日志	21
1.8.7 Vediamo 配置	22
2、Vediamo 模块	23
2.1 DiagServer	24
2.1.1 引言	24
2.1.2 配置（INI 参数）	24
2.1.3 诊断参数化	24
2.1.4 示例：操作示范	25
2.1.4.1 关闭服务器（“Kill”）	25
2.1.4.2 更换 CAESAR 硬件	25
2.1.4.3 采用其它固件对 CAESAR 硬件进行闪存处理	25
2.1.4.4 不具备硬件以及许可证条件下尝试 DiagServer（仿真模式）	26
2.2 StartCenter	27
2.2.1 结构	27
2.2.2 StartCenter 的功能	27
2.3 系统配置	30
2.3.1 引言	30
2.3.2 结构	31
2.3.3 功能	35
2.3.3.1 File 菜单	35
2.3.3.2 Edit 菜单	36
2.3.3.3 菜单视图	38
2.3.3.4 菜单帮助“?”	38
2.3.4 采用系统配置进行工作	44
2.3.4.1 创建新的系统说明	44
2.3.4.2 采用预存系统说明进行工作	44
2.3.4.3 更新系统说明内容	44
2.3.4.4 检查系统说明的内容一致性	44
2.3.4.5 搜索功能	44
2.3.4.6 批量模式	46
2.3.4.7 设定命令行参数之外的选项	47
2.3.5 特别特性	48
2.3.5.1 服务过滤器	48
2.3.6 配置（INI 参数）	48
2.4 Ecoute	49
2.4.1 引言	49
2.4.2 GUI 结构	50
2.4.3 Ecoute 文件	54
2.4.4 Ecoute 功能	54
2.4.4.1 选择以及关闭系统	55
2.4.4.2 关闭系统	55
2.4.4.3 联系 ECU	55
2.4.4.4 执行服务	58
2.4.4.5 读取错误	59
2.4.4.6 读取永久性错误	62

2.4.4.7 状态读取的错误.....	62
2.4.4.8 系统快速测试.....	64
2.4.4.9 服务组	67
2.4.4.10 变体编码.....	78
2.4.4.11 闪存处理.....	80
2.4.4.12 OBD2.....	83
2.4.4.13 配置 Ecout 以及服务器选项	88
2.4.4.14 宏	93
2.4.4.15 Java 程序	94
2.4.4.16 程序生成器.....	95
2.4.4.17 标准对象.....	102
2.4.4.18 显示跟踪以及监测数据	102
2.4.4.19 手动命令输入.....	104
2.4.4.20 CAN 总线仿真	109
2.4.4.21 快照文件存储.....	112
2.4.4.22 ECU 通讯仿真.....	113
2.4.4.23 夹子 15 处理方式.....	117
2.4.5 Ecout 菜单	119
2.4.6 键盘操作	125
2.5 Java 句柄功能	127
2.5.1 Vediamo Java 接口	127
2.6 Java 程序 (Java 程序)	128
2.6.1 引言	128
2.6.2 从 Ecout 或者其它客户端执行 Java 程序	128
2.6.3 作为独立客户端的 Java 程序	129
2.6.4 示例: 程序、编译以及执行简单的程序.....	129
2.6.5 特殊情况	130
2.6.6 配置 (INI 参数)	131
2.7 黑盒.....	132
2.7.1 引言	132
2.7.2 结构以及功能.....	132
2.7.3 黑盒功能	132
2.7.4 BlackBoxViewer: 运行时显示的日志	133
2.7.5 与其它应用程序链接.....	133
2.7.6 配置 (INI 参数)	134
2.8 PSR 适配器	135
2.8.1 引言	135
2.8.2 Vediamo 与 PSR 之间的通讯.....	135
2.8.3 PSR 客户端的功能.....	137
2.8.4 配置	139
2.8.5 发动机表	139
2.8.6 示例 - 如何.....	141
2.9 工人客户端	144
2.9.1 引言	144
2.9.2 结构	145
2.9.3 功能说明	146
2.9.4 示例: 如何.....	146
2.9.5 指令行参数	147
2.9.6 配置 (INI 参数)	147
2.10 其它客户端	149
2.10.1 闪存处理工作站.....	149
2.10.2 DiMeLo.....	149
2.10.3 UVI	149
2.10.4 更多的客户端以及实用程序	149
2.11 INI 编辑器	151
2.11.1 菜单	151
2.11.2 用户界面区域.....	152
2.11.3 输入元素.....	152
2.11.4 所有 INI 参数	153
3 如何	154
3.1 连接车辆	155
3.2 没有车辆情况下连接 ECU	156
3.3 对 ECU 进行闪存处理.....	158
3.4 重启服务器	159
3.5 从 ECU 中读取测量结果.....	160
3.6 读取 ECU ID 块	161
3.7 读取并且清空 ECU 错误存储器	162
3.8 执行快速测试	163
3.9 执行变体编码	164
3.10 执行 Java 程序 (Java 程序)	165
3.11 更改 K-线路与 CAN 之间的连接.....	166
3.12 在我关闭的相同状态下打开 Ecout	167
4、词汇表	168
5、INI 参数.....	170
6、PSR 消息.....	176
7、示例: Java 程序	182

1、引言

Vediamo – 发动机专用的分布式诊断应用程序（Verteilte Diagnose Anwendung für Motoren）- 是一种电子控制单元（ECU）诊断功能专用的软件系统，该诊断功能整合在 CAESAR/DIOGENES 处理链中。本系统可以通过 K-line 以及 CAN 诊断所有 ECU，其中包括了 RTMD+、MBISO、KWFB、KW2000 乃至 UDS 的所有协议。Vediamo 的诊断对象不仅限于发动机的 ECU。

Vediamo 主要基于梅赛德斯-奔驰 DCDI 标准通讯平台 CAESAR，因此支持部件 C、部件 A、部件 Y 等所有通讯硬件，本公司所有诊断应用程序使用的而且可以从中央 TAMINO 数据库获取的 DIOGENES 数据均可用作诊断数据。

Vediamo 后续版本将基于 GSP/ODE 开发的 MVCI（ASAM 标准），而且还会支持 DOX 诊断数据格式。

客户端/服务器端架构的实现方式还具有以下功能：

- 采用多个、同时运行的特殊客户端应用程序进行分布式诊断，比如，在 PSR 适配器上进行测试运行控制，同时还可以采用 Ecoute 进行验证；
- 通过添加客户端来扩展系统；
- 同时对多个 ECU 进行诊断、编码或者闪存处理。

1.1 应用程序范围、重点

Vediamo 最初采用了两个用户组：开发以及生产。

系统配置根据用户特定需求提供 ECU，此类需求包括：选择标准接头、整合自动执行初始化服务、从较长的名单中选择已选中的服务等等。

系统架构设计允许通过添加功能甚至整个应用程序的方式对系统进行轻松扩展。主要采用的是 Java 程序，针对此类语言 Vediamo 提供了简单但仍然完整的连接，其中包括了所有 ECU 功能。用户即便 Java 标准语言编程能力有限，仍然可以自行开发应用程序。Vediamo 开发人员后续会将具有一般适用性的功能整合进客户端中（比如，图形测量显示或者快速测试），或者针对此类功能开发新的 DCOM 客户端。

1.1.1 开发

Ecoute 客户端允许采用各种专用工具交互式访问一个或者多个 ECU 的所有可用功能，比如，可以访问所有可用服务、采用选定的各种类型服务定制窗口（此类服务可以一次性执行或者定期执行）；可以更改所有通讯参数，可以将任何请求消息发送给 ECU，可以采用多种专用格式对所有数据流、操作步骤以及事件进行日志记载。

1.1.2 生产

测试台控制器（PSR）适配器以及 UVI（Unipas-Vediamo 接口）允许在不对软件进行手动更改的情况下进行自动测试。整套测试运行均由测试测试台控制器进行控制，而 Vediamo 诊断服务器则提供测试运行所需的一切数据，并且可以评估测试结果。为了达到无需专业技术仍可测试诊断软件的目的，Vediamo 客户端的 TS 适配器以及 UVI 采用了非常简单的接口，通过精简以满足测试台的需求，这样可以采用高效方式通过串行线会或者网络连接访问诊断数据。

闪存站可以对 ECU 进行自动闪存处理。

1.1.3 Java 程序

Vediamo-Java 接口允许采用 Java 标准编程语言生成定制的应用程序，同时无需了解诊断协议、CAESAR-DIOGENES 接口、客户端-服务器端架构。此类 Java 程序可支持客户端 Ecoute（开发）或者 TS 适配器（发动机测试装置），还可以单独用作独立应用程序。

1.2 入门

1.2.1 基本术语

通讯硬件	硬件组件（CAESAR、DCDI），连接 PC 以及 ECU。计算机采用 CAESAR 软件（CAESAR-从库 c32s.dll）操作本硬件以及硬件驱动程序。
ECU	电子控制单元，控制发动机或者其它系统功能的车辆电子单元，可以采用直接或者通过另外一个 ECU（网关）与外部设备进行通讯从而提供诊断功能（测试仪、装有通讯硬件的 PC）。
系统	本术语是指同时诊断的一套 ECU。其中包括单个或者多个 ECU。Vediamo 允许同时与多个 ECU 进行通讯，需要取决于 ECU 的类型（K-line 或者 CAN）以及诊断硬件的配置（连接的数量以及类型、硬件组件的个数等等）。特别是在快速测试过程中，我们将整车视为一个系统，并且与所有安装的 ECU 进行通讯。 系统文件（VSB-Vediamo 系统二进制文件）是针对每个系统生成的。仅当与包括的 ECU 进行参数化连接的时候才有效（CBF 文件）。 系统这一术语大体对应是 ASAM 系统中的项目。
诊断服务	ECU 的一种功能，可以引起特定的动作或者从 ECU 向测试仪（PC）发送信息。典型的服务包括读取测定结果（发动机温度等）、设定控制器（节流阀等）等等。 对于服务来说，一般是从测试仪向 ECU 发送消息，然后接收并评估回复。但是，消息的实际发送数量可能很大。
资源	硬件连接，对于 K-line 通讯（串行通讯）来说，每个 ECU 均需要单独线路，可以采用多工器或者手动采用香蕉插头切换线路。CAN 通讯总是使用 CAN 总线，但是通讯硬件需要针对每个 ECU 采用内部通道。通道的数量主要取决于采用的 CAESAR 硬件以及软件。
联系	采用 ECU 可以进行的数据交换状态。为了建立联系，必须对 ECU 进行初始化（激活）。诊断硬件主动维持本状态。ECU 只有处于联系状态才可回复测试仪发来的消息。
变体编码	特殊的诊断服务，在 ECU 中采用变体存储各种信息（比如，针对车辆选配件）。
闪存处理	Vediamo 可以采用各种软件（固件）对多数 ECU 进行编程。这一过程称为闪存处理。
DiagServer	Vediamo 系统的主要模块。控制诊断数据、ECU 状态并且代表客户端协调执行各种服务。
客户端	一种应用程序，执行特定任务并且通过 DiagServer 与 ECU 进行通讯。

1.2.2 Hasty 操作说明

管理员提示

Vediamo 安装完毕之后（需要管理员权限），无需管理员权限即可运行。由于 Vediamo 可以在工作过程中访问各种文件，还可以写入此类文件，因此必须授予某些访问权限。

Vediamo 安装到 ALLUSERSPROFILE 文件夹中。环境变量中定义了该文件夹，具体情况主要取决于操作系统的版本。

在 Windows XP Professional 中，路径如所示：

C:\Documents and Settings\All Users\Application Data\Vediamo

在 Windows 7 中，文件夹路径如下所示：

C:\ProgramData\Vediamo

该文件夹中共有 4 个子文件夹，任何用户均应当具有完全访问权限：

- VediamoData – 诊断数据（CBF、VSB 等）
- VediamoShorttestData – Shorttest 数据文件
- Config – INI 文件、许可证文件以及其它配置文件
- Log – 运行时创建的日志文件

安装时可以更改前两个文件夹。

如果安装过旧版的 Vediamo，那么设置会采用之前的文件夹而非新的文件夹。

如果您没有诊断硬件及/或许可证

您可以采用仿真模式运行 Vediamo，并且测试各种功能。这样可以确定 Vediamo 是否能提供您需要的工具。

- 采用标准设置安装 Vediamo
- 从 Vediamo 官网主页中载入选定 ECU 的数据，将此类数据复制到数据文件夹中（一般是...\VediamoData）
- 打开 Vediamo StartCenter
- 从 ECU 列表中选择“Simulation”，点击按钮启动 Ecouste。
- 使用 System / Select 打开系统（包括一个或者多个 ECU）
- 尝试菜单中的所有功能。此类功能提供了 CAESAR 的各种实际功能。
- 仿真模式中可能无法使用所有功能。其中包括，闪存处理、手动指令输入以及监测。其它功能则均可执行，但是结果未必令人满意 – 比如，测量服务总是返回带有“？”的结果。Read Errors 服务返回“no Errors”结果。但是，如果您拥有相关 ECU 的仿真文件，那么仍然可以在仿真模式中获取其它结果。
- 可以完全卸载 Vediamo。前往 System Control / Software / Vediamo Diagnostic System，选择 Remove。随后可以删除路径 Vediamo in ALLUSERSPROFILE 以及 C:\Program Files\Vediamo 中的所有文件。

如果您拥有诊断硬件但是没有 Vediamo 许可证

您可以安装 Vediamo 并且采用仿真模式进行测试（详情参见上文）。如果您希望购买许可证之前在实际操作中测试 Vediamo，那么采用以下操作步骤：

- 安装 CAESAR 硬件以及必要的驱动程序。详情参见 Vediamo 官网主页，操作步骤与使用的硬件类型无关。
- 从 StartCenter 中选择“update Vediamo Server / CAESAR”
- 确保选中您使用的 CAESAR 组件。如果无法确定，则可以选择多项内容。一旦连接上特定的组件而且其地址可以访问，那么软件会自动识别此类组件。
- 启动“update”功能。稍等一会，然后会显示检测到的硬件信息。
- 记下显示的硬件序列号。

- 订购测试许可证，采用该许可证启动硬件。多种硬件组件可以使用一个许可证。
- 我们会在适当的情况下（通过电子邮件）发送定制的 VLicence.inf 文件。将该文件复制到...\Vediamo\Config 文件夹中（通常位于 C:\Documents and Settings\All Users\Application Data\Vediamo\Config）
- 再次打开 StartCenter，启动 Ecouté。如果未显示您需要有效许可证或者未发现硬件等信息，那么就说明您的 Vediamo 可以使用了。
- 连接 ECU。具体目录如下所示。
- 必要的情况下，从 Vediamo 官网主页中下载您的 ECU 专用数据，将此类数据复制到...\VediamoDaten 文件夹中。

如果您拥有硬件以及许可证

按照上述步骤验证已经安装了 Vediamo 之后，复制您的 ECU 专用数据，正确连接 ECU，正确安装 CAESAR 硬件专用许可证，然后即可采用 StartCenter 开始工作。您可以访问所有已安装的 Vediamo 组件以及相关设置。

Ecouté 是使用频率最高的程序，该交互式客户端提供了各种诊断功能。该客户端以及其它客户端仅当 DiagServer 运行的时候才可使用。一旦某个客户端（这种情况下是 Ecouté）需要访问此项功能的时候，则通过 DCOM 自动启动该功能。由于每台计算机仅可激活一个 CAESAR 实例，因此这种服务器属于“单一模式”。根据具体的客户端/服务器架构，可以同时使用多种 Ecouté 客户端（或者可以随 Vediamo 一并安装的客户端或者您自行开发的 Java 应用程序）。

使用系统配置生成或者更改系统文件（可以不采用此类文件进行工作，但是此类文件可以降低工作的难度）。更改了某个系统文件之后，如果正在运行 DiagServer，那么必须重启 DiagServer。最简单的方式为，关闭并且重启 Ecouté 客户端。

您还可以需要使用 INI 编辑器。可以从 StartCenter 中点击 Options 图标启动该应用程序。

如果软件出现崩溃、意外事件、联系问题等，应当设定 Vediamo.ini 中的参数 [BLACKBOX]run=1，采用这种方式进行 BlackBox。然后服务器会启动 BlackBox，并且在后台运行该应用程序，以便对发生的情况进行日志记载。必要的情况下，可以将 BlackBox 的日志发送给 Vediamo 团队进行分析。

可以根据您的需求以及仅采用 Ecouté 的工作方式设定所有 INI 参数，然后无需 StartCenter 即可直接启动 Ecouté。从 Windows 菜单中选择 Start / Programs / Vediamo / Ecouté，或者在桌面上放置 Ecouté.exe 的快捷方式。

启动 Ecouté（或者其它 Vediamo 给定客户端）即可自动启动 DiagServer。关闭最后一个客户端也会同时关闭 DiagServer。

注意：

停止服务器可能需要 20 秒。如果在服务器完全停止之前启动了 Vediamo 客户端，那么可能导致出现崩溃以及未识别硬件等等不稳定的程序状况。这种情况下，请使用任务管理器确保已经终止了 DiagServer 进程（或者手动终止），移除所有 Ecouste 或者其它客户端的“残余程序”，然后重新启动客户端。无需重启计算机。

如果出现问题...

...您可以首先启动 BlackBox。该程序会创建日志。然后检查 Vediamo 文件夹。为了对错误进行跟踪，Vediamo 团队需要使用的软件版本、设置（Vediamo.ini 文件位于程序文件夹...\Vediamo\Config 之中）、您的 DCDI 硬件规格（类型、序列号）的准确信息，以及所有日志文件以及使用的 CAESAR 数据的准确信息。如果您采用标准配置安装了 Vediamo，那么可以采用下列路径进行查找：

- CAESAR 数据以及系统文件位于文件夹...\VediamoDaten 之中
- Vediamo 系统生成的文件，比如测量结果以及控制器组位于文件夹...\VediamoDaten 之中，名称与 Vediamo 系统相同。
- 程序日志以及 ECU 通讯跟踪位于文件夹...\Vediamo\Log 及其子文件夹之中
- 通过文件属性/版本，或者通过启动程序并且调用 Info on....来获取程序版本信息。该菜单项位于 Ecouste 的主菜单“?”之中。对于任务栏菜单中没有用户界面的程序

（DiagServer、BlackBox、PSR 客户端）：采用鼠标右键点击相关的 、 或者 

图标，选择 Info on....该信息窗口会告知您正在运行的程序以及采用的 INI 文件。如果您同时安装了多个版本的 Vediamo，那么此类信息可以避免出现各种问题。

1.3 开发周期

我们每年至少推出一个新的 Vedioamo 版本，其中整合了最新的 CAESAR 硬件与软件，并且支持原有的以及新的各种协议。

一旦出现重大错误、紧急变更或者新的 CAESAR 版本，则会发布额外的版本。对于后一种情况来说，Vedioamo 会在数周测试以及质量控制完成之后发布最新的 CAESAR 版本。

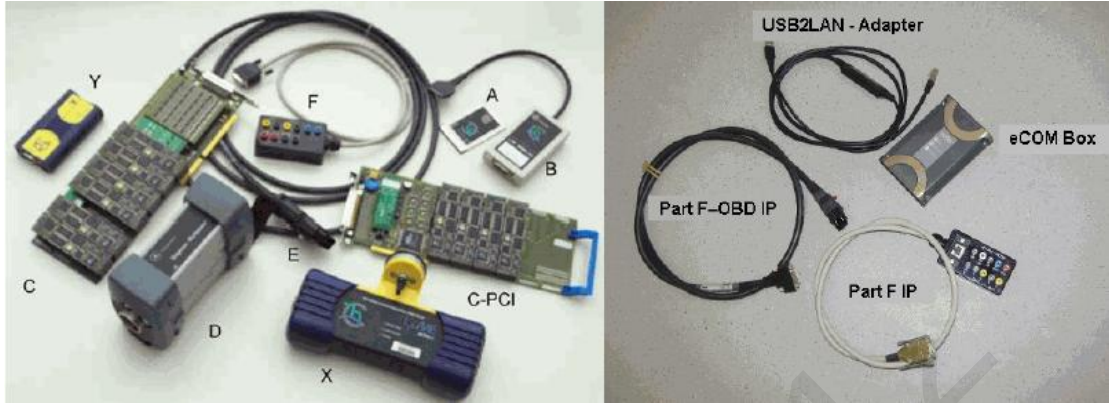
2Q 1850971v

1.4 硬件以及其它软件的相关信息

1.4.1 CAESAR

通过适当驱动程序以及库与诊断硬件（CAESAR、DCDI）建立的连接，这是 Vediamo 诊断服务器的基础。

CAESAR 硬件以及软件供应商如下所示：



支持下列硬件组件：

- 部件 A：PCMCIA 卡，还需要部件 B 以及驱动程序。
- 部件 C：ISA 或者 PCI 插入卡，采用各种负载封装（用于平行诊断的处理器）以及集成的多工器。
- 部件 D：“第一代”诊断硬件，配有串行接头。
- 部件 E：OBD 连接线，配有集成的多工器，与部件 A+B、Y 或者类似部件相连。
- 部件 F：实验室装置专用连接盒，用于部件 E 位置，但是未配备多工器。
- 部件 J：通过器械。
- 部件 P-eCom：也称为 MVCII 器械，PDUAPI。
- 部件 X：诊断部件，配有无无线 PC 接头。
- 部件 Y：诊断部件，配有 USB 或者串行 PC 接头。
- 部件 W：SDConnect...

与使用的硬件组件无关（还可以同时采用多种不同类型组件），需要各种 CAESAR 文件（动态链接库、Release.cmf、INI 文件、协议定义等）、每个 ECU 需要诊断的 CBF 文件格式的 DIOGENES 参数，以及其它文件（CCF 编码文件、CFF 闪存文件等）。

使用之前尤其应当安装硬件传送的驱动程序软件。

对于部件 P-eCom 来说，配置文件 slave.ini 必须报适当的 RootPath-Entry，主要取决于驱动程序软件必须安装的文件夹。

详情请查阅用户硬件部分的文档。

注意：

CAESAR 硬件：Windows 7 不支持部件 A/部件 B/部件 C/部件 Y。

1.4.2 LUCA

Langner 通用通讯 API (LUCA) 是汉堡 Langner Communications AG 生产制造的。针对不同通讯协议提供标准化编程接口，允许有效增强协议，同时不会干扰程序流。采用 LUCA.DLL 的情况下，PSR 以及 PSR 适配器通过串行线路 (3964R 协议) 或者通过 LAN (HDLC、TCP/IP) 进行通讯。LUCA.DLL 库随 Vediamo 一并交付，并且在必要的时候进行安装 (配有 PSR 适配器)。

建议在开发测试台软件的时候实现相同的通讯软件。必要的情况下，Vediamo 开发团队会提供代码示例，以便对 LUCA 进行整合。

1.4.3 PSR

对于测试台的交互而言，最重要的是简便性、健壮性、清晰性、高效性、可靠性以及容错性。同时还必须可以适应各种测试程序、各种品牌测试台以及持续变化的 ECU 及其数据，而且要求采用大量的方式优化测试时间，以便在更大的流程中进行整合。

这样就要求 PSR 适配器具有高弹性，而且必须依照 Vediamo PSR 规范采用灵活方式设计测试台软件。测试台控制器的整个流程不可能处于静态，因为某个动作或者消息序列之间的时间不可能总是相同的。比如，载入新的 ECU 数据可能随着每次测试运行而发生很大变化。而选择可视化或者日志窗口，仍然可能在很大程度上加快反应速度。

1.4.4 UNIPAS

Vediamo 使用本测试台控制软件进行通讯，同时使用 UNIPAS 中的特殊客户端 UVI 指令。采用 XML 格式，通过 DCOM (通过 LAN) 传送 Vediamo 的回复。UVI 的功能相对有限。

1.4.5 Java 运行时

运行 Java 程序需要采用 Java 运行时环境。从理论上讲，每个标准化 Java 运行时均应当正确地执行程序。但是，建议采用已包括的以及自动安装的版本。采用包括的版本测试了 Vediamo-Java 接口。此外，在安装了 Vediamo 的情况下，会采用正确的方式自动向 Vediamo.ini 输入所有 Java 相关参数。

1.5 观点

Vediamo 今后会继续与诊断流程进行整合，同时也会支持所有新的诊断硬件，并且会参考最新的可用 CAESAR 软件。Vediamo 会支持将要引入的标准化 ASAM 服务器软件，并且会持续提供已熟悉测试台的用户界面以及连接。

系统文件（VSB）等 Vediamo 特定数据、测量以及控制组（MWG、STG）可能采用原有版本，或者 Vediamo 将现有文件转换为新格式。

2Q 1850971v

1.6 订购、许可证、支持

Vediamo 官网主页中可以找到 Vediamo 最新版本以及 ECU 的专用数据。

Vediamo 许可证对特定的通讯硬件有效，与计算机无关。CAESAR 初始化之后，诊断服务器读取硬件 ID 编号，并且与许可证文件中的输入项进行对比。如果多个不同硬件组件与计算机相连，则仅需对其中一种组件授予许可证，Vediamo 仍然可以访问所有部件。Ecouté 启动之后，状态窗口中会显示所有找到的硬件组件及其 ID 编号。

如果获得了 Vediamo 的许可证，那么可以单独或者同时使用任何数量的客户端。但是客户端 DGR 以及 FlashStation 需要单独的许可证。

许可证对于特定版本有效，但是仅前两个数字对具有相关性，即，05.00.00 许可证适用于所有后续的升级版本 05.00.xx，但是不适用于 04.02.xx 或者 05.01.xx。

采用仿真模式工作无需许可证。这一模式中不会与 ECU 进行通讯，而是在服务器采用仿真方式与 ECU 进行数据交换。这一模式仅用于演示、培训以及测试 DIOGENES 参数化情况。

可以申请功能无限的临时许可证用于评估。

如需订购单个、多个许可证或者报告错误，请联系：

梅赛德斯-奔驰

PWT/VEP 部，

Werk 010 HPC H152，

传真+49 -(0)711 17 - 7908 1949

可以从 Vediamo 官网主页的 Contact 下获取订购表、最新信息或者问题帮助信息。

1.7 安装

1.7.1 系统需求

- Pentium-III-PC 1GHz CPU（推荐 1.5GHz）
- Windows 2000、Windows XP 或者 Windows 7
- 1024 兆 RAM 存储器
- 250 兆可用硬盘空间用于安装 Vediamo, 此外还需要一定的空间放置诊断数据(推荐 5GB)
- 根据需要使用的通讯硬件, 需要使用空闲的 PCI 或者 ISA 接头、USB 接口、COM 端口或者 PCMCIA 插槽。
- 分布式应用程序必须采用网络连接（服务器以及客户端位于不同的计算机上）。
- 针对与测试台控制器的通讯而采用网络连接或者串行接口（COM 端口）。

在安装 Vediamo 之前以及之后, 必须单独安装诊断硬件以及必要的驱动程序。在安装之后, 必须在 Vediamo.ini 设置文件中建立适当的项, 以便允许硬件配合 Vediamo 发生相关功能。

PC 性能过低或者从其它进程中载入（打开的应用程序、SMS 服务、病毒扫描程序等）可能会降低 Vediamo 的性能。

1.7.2 早期版本

根据客户端-服务器架构, 不可能同时安装两个 Vediamo 版本。如果已经安装了早期版本, 那么在安装新的版本之前必须卸载之前的版本。但是, 无需删除设置的副本(Vediamo.ini) 以及用户生成的文件（系统文件 BSB、测量以及控制器组等）。

如果在早期版本 Vediamo 安装文件夹中安装了新的版本, 那么新版本 Vediamo 会采用之前版本的 Vediamo.ini 设置文件。

必要的情况下, 新版 Vediamo 的文件格式变更可能自动应用于相关文件。

1.7.3 安装操作步骤

总是采用单独的设置程序来提供 Vediamo。统一所有系统组件, 而且不可能与其它版本的组件混合使用（相同版本号的发布版本以及调试版本也不可同时使用）。安装的时候需要具有管理员权限。如果采用 Windows 7, 那么应当采用管理员权限运行设置程序。

采用典型操作步骤运行设置程序, 用户可以选择组件的相关路径。安装之前, 设置程序会检查用户是否具有管理员权限, 以及是否提供了正确的操作系统。如果不满足其中一个条件, 那么就会放弃设置程序。

安装过程中, 设置程序会检查 Vediamo.ini 或者 Vediamo_ini.old 设置文件是否处于规定路径中。如果符合上述要求, 那么会采用新版本仍然有效的所有设置, 而仅需输入新项。其中包括, ECU 参数化文件的路径。如果未找到 INI 文件, 那么用户必须选择参数化所在的路径, 然后即可开始安装。与其它设置相同, 后续可以采用 INI 编辑器更改此类设置。

采用 Startmenu / Settings / System Control / Software Uninstall Vediamo 卸载应用程序。

1.7.4 安装以及检查 eCom 硬件（部件 P）

- 在使用具有“USB2LAN Adapter”的 eCom 的情况下, 必须先安装驱动程序。驱动程序位于 Vediamo 主页“DCDI Treiber”之中。或者, 还可以直接从 ASIX 中获取驱动程序（<http://www.asix.com.tw/products.php?op=pltemdetail&PItemID=97;71;101&PLine=71>）。
- 接口配置:

- 针对 eCom 硬件切换电源。
- 在 Network 中, 查找正确的连接(“ ASIX AX88772A USB2.0 to Fast Ethernet Adapter”等), 编辑连接属性。在 TCP/IP 属性中输入下列**固定 IP 地址**:
 - IP 地址 169.254.255.40
 - 子网掩码 255.255.0.0
- 防火墙设置:
 - 根据使用的防火墙, 检查是否打开了下列端口。详情参见诊断主页中的 Tools|Diagnose Hardware|Driver & Installation eCOM 相关信息。

用途	协议	端口
eCom	UDP	1024
	UDP	4011
	UDP	4012
DoIP	TCP	13400
	UDP	13400
	UDP	13401
	UDP	13402
	UDP	13403

- 功能检查:
 - 针对 eCom 硬件切换电源。(eCom 硬件上的绿色 LED 灯必须亮起)
 - 将 USB 插入 LAN 适配器中
 - 在...\Program Files\Vediamo\Caesar\driver\eCom\I+ME Actia GmbH\XS D PDU API 中执行 IME_D_PDU_API_Tester.exe 应用程序。在顶部选择 “D-PDU API DLL” PDUAPI_I+ME_ACTIA_XS.dll, 按下底部的 START 按钮。如果一切正常, 那么会在此后选中列表中的所有复选框。
 - Vediamo StartCenter 现在可以识别 eCom 了 (否则请检查是否选中了 eCom)。

1.7.5 安装选项/参数

日志会记载安装过程中的所有用户输入项。形成的回复文件可用于执行静默安装, 即, Vediamo 可以在无需用户干预的情况下自动安装到其它计算机上。然后根据日志文件进行相关选择。

如需采用日志在 vediamo.iss 文件中记载用户输入, 那么应当采用下列参数启动程序:
VediamoSetup.exe /r /f1"c:\temp\vediamo.iss"

此项回复文件可以用于在其它计算机上自动执行设置程序而无需用户干预:
VediamoSetup.exe /s /f1"c:\temp\vediamo.iss"

无需输入文件名。这种情况下, 会在 Windows 系统文件夹中创建/读取 “Setup.iss” 文件。

注意:

如果输入了文件名, 那么即使文件处于当前文件夹中, 必须包括完整的路径。未输入路径或者无关路径则可能出现无法预见的结果。

为了完成预设, 建议采用预先安装的 Vediamo.ini 文件组合静默安装方式, 即, 采用下列方式生成批处理文件:

- 1、采用子文件夹 BIN 生成安装路径。

2、将准备好的 **Vediamo.ini** 复制到 **BIN** 文件夹中。

3、采用上述参数启动设置程序。

- 采用新的回复文件登录时，删除/重命名之前的回复文件。
- 禁止使用之前 **Vediamo** 安装形成的回复文件。
- 设置流程的变化主要取决于是否可以提供 **INI** 文件。采用 **INI** 的回复文件应当仅可与该 **INI** 配合使用。
- 静默安装过程中在 **Windows** 任务栏中会显示 **InstallShield** 图标。如果未看到该图标，或者该图标一闪即逝，那么表示并未运行安装程序。

QQ 18509714

1.7.6 其它必要的硬件/软件

DCDI 硬件

表格以及订购流程包括在 Vediamo 官网主页 Ordering Information / CAESAR Hardware 之中。

关于诊断硬件，详情参见诊断门户。

启动程序 (DCDI)

位于 Vediamo 官网主页 Downloads / DCDI Driver 之中。

诊断数据 (VSB、CBF 以及 Co.)

位于 Vediamo 官网主页 Downloads / Diagnostic Data 之中。

Java 开发环境

如需自行编写 Java 程序，那么就需要开发环境。可以从 Eclipse、Sun Microsystems 免费获取该环境，或者也可以从 Borland 订购 JBuilder。

QQ 18509714

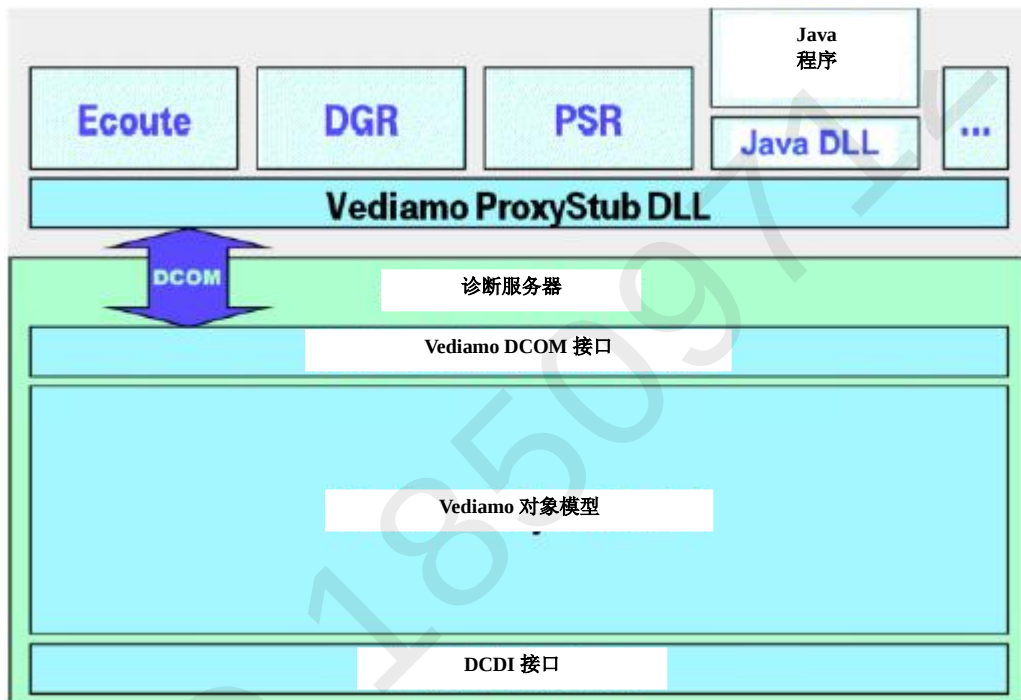
1.8 设计与工作模式

1.8.1 C/S 架构

通过将 Vediamo 诊断系统分为服务器以及多个客户端组件，可以达到以下目的：

- 同时采用多个特殊客户端应用程序进行诊断，比如采用 PSR 适配器测试负载控制，同时采用 Ecouté 在字节级别检查通讯情况。
- 通过添加客户端（比如，Java 程序，自动运行某些进程，或者评估数据并将其传送给其它应用程序）扩展系统。
- 分布式安装：同时在多个计算机中检查测试流程：在测试台（工人）以及办公室（倒班经理）。
- 同时操作多个 ECU，比如，在对一个 ECU 进行闪存处理的时候，同时校准另外一个 ECU。

1.8.2 层模型



Vediamo 系统架构要求采用最简单的复用性：

- 我们对诊断硬件进行了封装。如果引入了新的硬件，那么必须采用 DCDI 接口。
- 根据用户需求以及诊断中的流程调整 Vediamo 对象模型，与硬件（CAESAR）以及中间件（DCOM）无关。
- 更换 DCOM 级别（服务器端 DCOM 接口以及客户端 ProxyStub）不会影响 Vediamo 对象模型以及客户端代码。这样就可以在不更改 Vediamo 客户端的情况将其用于今后的新系统（比如，ASAM 3D）。

1.8.3 DCOM

DCOM（分布式组件对象模型）是微软用于实现分布式客户端/服务器应用程序的一项技术。属于 Windows 98 以及 NT4.0 操作系统的组成部分。Vediamo 采用此项技术用于客户端与诊断服务器之间的通讯。

安装之后，Vediamo 可以立即投入工作。在同一台计算机上可以同时运行 DiagServer 以及客户端，而无需重启计算机或者更改系统配置。

必须采用下列方式配置诊断服务器方可进行分布式工作（客户端以及服务器位于不同的计算机）：

操作系统提供 DCOMCNFG.EXE 来配置 DCOM。

注：用户必须具有管理员权限方可运行 DCOMCNFG.EXE。

必须采用 DCOMCNFG.EXE 进行下列设置：

- 在诊断服务器应用程序运行的情况下设置用户招呼。通过（Vediamo DiagServer）Properties 以及 Identity 实现此类操作，选择 This User 并且指定一个账户。
- 在诊断服务器上设置访问权限。通过 Security 的（Vediamo DiagServer）Properties 实现此类操作；选择 Use User Defined Access Privileges，并且使用 Add 来授予适当用户或者组的各种访问权限。
- 设定授权级别为（None）。通过 General 中的（Vediamo DiagServer）Properties 实现此类操作。
- 在诊断服务器设定启动权限。通过 Security 的（Vediamo DiagServer）Properties 实现此类操作；选择 Use User Defined Access Privileges，并且使用 Add 来授予适当用户或者组启动权限。

诊断服务器应当采用自身的用户帐户，而且应当自行定义 Vediamo 用户组。诊断服务器的用户帐户必须授予 Log-In as Batch Job 权限（在用户管理器的 Guidelines / User Privileges 菜单中）。用户管理器位于 Start / Programs / Administration / Computer Administration 中。可以将 Vediamo 的所有用户分配给 Vediamo 用户组。这样做具有一定的优点，无需针对每个新用户采用 DCOMCNFG 授予启动以及访问权限，因为可以通过组来实现此项功能。

1.8.4 多客户端

此项 Vediamo 特性运行顺利实现特定客户端，该客户端提供了典型功能的选项，而不会牺牲便利性以及完整性。除了标准客户端之外，还可以实现特定的客户端。这样可以避免采用过多的客户端并且可以缩短开发周期。

工人客户端用于发动机测试台，其中仅具有常规测试流程的相关功能。这样可以避免未经授权的人员采用过多的工具而造成配置受损。Ecoute 客户端控制并且采用日志方式记载流程，可以在工作或者调试的同时平行运行，而且不会更改设置。此外，还可以通过 Ecoute 直接干预流程，比如，可以执行某个动作来控制计算机流程。这样可以无需等待纠正的测试台软件来分析今后的流程。

其它应用范围报将第二个工人客户端与倒班经理的计算机相连，该人员可以同时监测一台以上的测试台的事件。

1.8.5 多任务（多个 ECU）

Vediamo DiagServer 采用多任务方式同时操作多个 ECU，以便充分利用所有通讯通道（CAESAR 资源）以及 CPU 能力。由于 ECU 通讯多数情况下处于等待 ECU 回复的状态，因此可以对计算机的性格进行优化。Vediamo 只有在采用多任务的情况下才可以采用一台计算机以及一个 CAESAR 硬件同时闪存处理以及校准两个 ECU。

Vediamo 的关注重点是应用程序（即，需要完成的实际任务）而非硬件设置。这意味着，一个输出窗口中可以同时显示不同 ECU 的测量结果，而且可以在不同窗口内调整各种驱动器。

1.8.6 日志

常规操作无需运行时记录的日志文件，但是对于疑难问题的解决来说是必需的。可以采用不同的 INI 参数（详情参见 Vediamo 配置）打开以及关闭各种日志类型。



多个文件夹中存储日志文件，主要取决于 INI 文件。但是无需在 StartCenter 中搜索此类文件，“Logfile administration”按钮打开所有日志文件的清单，可以将其打包为 ZIP 压缩文件或者将其删除。

程序日志记载（BlackBox）

任何程序都可能存在错误。Vediamo 也不例外。必须日志记载所有正在运行的活动，以确定位置以及具体的状况。可以通过软件行程记录仪（BlackBox）记载此类情况。

如果 DiagServer 或者客户端崩溃或者出现其它系统功能问题，那么应当在 INI 编辑器数据激活 BlackBox。

此类日志存在一些例外情况；通常不会写入硬盘中而是记录在 RAM 的环形缓冲中。如果发生严重错误（客户端或者服务器崩溃），或者如果用户申请，那么可以将最近的 500 项（通过 INI 参数来变更这一数值）写入文件中。此类日志包括当时运行的所有 Vediamo 应用程序项。如果发生问题，那么不必在打开日志记载的情况下重现当时状况（无法保证重现问题）。问题发生之后存储日志，而在正常工作期间不会额外增加系统负载。

CAESAR 日志

CAESAR 日志用于分析 CAESAR 硬件以及软件的各种问题。也会记载内部 CAESAR 流程。

可以采用各种 INI 参数来设置此类输出的详细程度。CAESAR 日志输出采用文本格式存储在 CAESAR.log 文件中。此外，还可以在运行时的 BlackBox 查看器窗口中显示日志输出内容。还可以采用 INI 参数控制程序的表现（详情参见 Vediamo 配置）。

采用通讯通道与 ECU 进行通讯。生成通道相关的日志以便分析与各个 ECU 的通讯情况：

- Ecouste 跟踪窗口中采用数据块形式或者带有时间戳的单个字节监测 ECU 的查询以及回复。通过 Ecouste Options / Protocol 配置此类内容。
- 在服务器端采用日志文件方式记载通道相关的 CAESAR 输出以及 ECU 通讯情况。针对每个通讯通道生成文件，文件名称中包括 ECU ID 而且结尾采用 kanal.log。通过 Ecouste Options / Protocol 或者直接采用 INI 参数进行此类配置。

Status.log

此类日志包括 Ecouste 状态窗口中显示的所有文本。

PSR Logs

PSR 适配器采用字节级别记录与测试台控制器的全部数据交换。一旦在 PSR 与 Vediamo 之间的通讯发生问题，那么就可以使用此类文件。一般应当关闭此类日志文件，因为此类文件非常复杂，而且在运行工作过程中无法中断或者删除。

借助工人客户端可以存储不同类型的日志信息。此类日志包括 PSR 与 Vediamo 之间交换的更易读格式的消息。

系统配置日志

系统配置会生成各种流程（比如，系统说明的升级以及一致性检查）的日志。由于系统配置采用了 CAESAR，因此采用系统配置的情况下，还会导致 CAESAR 生成的输出中日志记载内部流程。运行时可以在日志窗口中监测此类输出，但是此类输出还会存储到文件 VediamoSysConfLog.txt 中，以便进行后续分析。

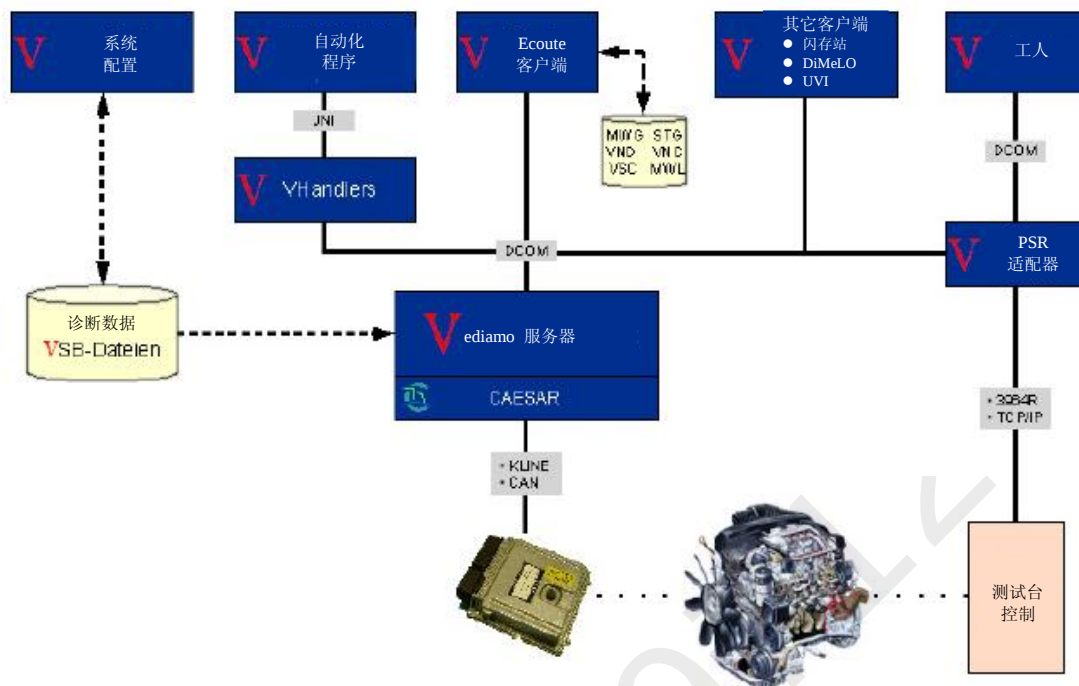
1.8.7 Vediamo 配置

采用文件 Vediamo.ini 配置 Vediamo 系统。该文件分为多个部分，每个部分分配给了一个系统模块（DiagServer、CAESAR、客户端、Java 连接，或者通用参数的“通用”部分）。在全新安装的过程中，将参数设定为允许采用最佳工作方式的数值。应用程序会在运行时更改某些数值（比如，窗口位置）。建议仅在理解并且明确相关结果的情况下对此类参数进行变更。此外，配置的时候还可以使用 INI 编辑器，后者包括了所有参数的详细说明。

以下 Vediamo 系统模块的相关小节中详细介绍了各个模块的配置方式。

2、Vediamo 模块

Vediamo 包括多个应用程序（图中采用蓝色表示）：



DiagServer:

该应用程序封装了诊断硬件以及软件，为客户端应用程序（本地以及 LAN 分布式）提供各种功能，并且可以协调客户端访问系统。DiagServer 管理 ECU 数据，建立与 ECU 之间的通讯，执行服务，将结果或者接收的信息传送给客户端。复杂的客户端管理权限允许客户端几乎采用无限方式访问所有必要的功能：服务器协调访问方式，以便确保

- 请求的信息只会送达需要的客户端
- 客户端不会彼此堵塞，最坏情况下只会拖慢彼此的速度（使用同一个通讯通道）
- 可以一次性多个客户端的相同信息，并且分发给各个客户端。

客户端:

用户可以看到并且操作的程序：Ecoute、工人客户端、DGR 等。设计用于完成特定任务：通过 DiagServer 与 ECU 交换数据，处理或者提交此类数据等。同时，客户端自身也可以作为其它应用程序的服务器，比如，PSR 适配器可以为 2 个客户端提供服务：测试台控制器以及工人客户端。

系统配置:

独立程序，用于生成以及编辑系统说明。

BlackBox:

一种看门狗程序。在 RAM 中存储所有 Vediamo 模块(服务器以及客户端)的日志输出，以便在崩溃或者其它严重问题的时候写入日志文件中。这种情况下，会在发生问题之前存储所有最终事件。这种连续工作方式会影响运行时或者硬盘空间。

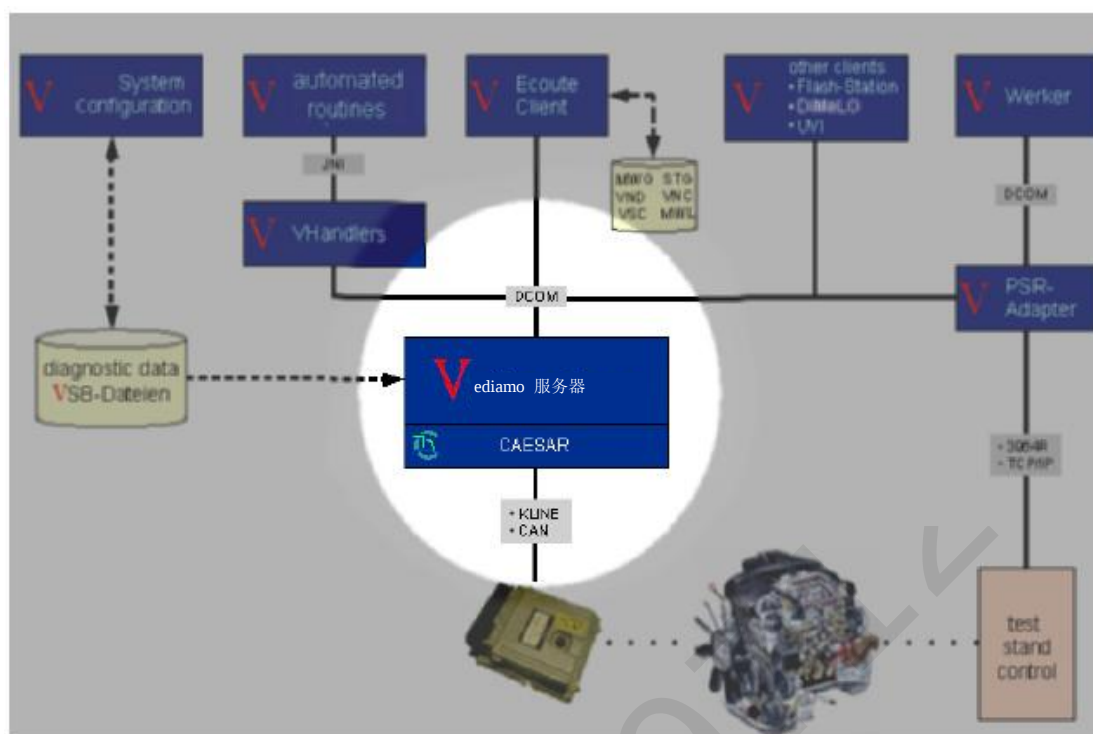
StartCenter:

采用综合方式显示所有 Vediamo 重要元素的用户界面。

INI 编辑器:

配置 Vediamo 设置项的一种程序。其中包括了所有参数的详细说明。

2.1 DiagServer



2.1.1 引言

服务器是整个系统的核心模块。主要工作是采用可用的DCDI硬件控制与ECU的通讯，管理数据，处理以及协调与客户端的数据交换。

除了菜单或者信息窗口之外，DiagServer自身没有其它的用户界面。可以通过任务栏中的 **V**（鼠标右键）访问菜单，其中包括了 Info on...以及 Exit。

DiagServer 拥有 DCOM 接口，客户端可以通过该接口访问相关功能。可以采用平行方式激活多个客户端，服务器主要负责协调并且优化数据交换问题。

2.1.2 配置（INI 参数）

可以采用 INI 编辑器来配置 Vediamo DiagServer。

DiagServer 的 Vediamo.ini 中包括了以下各部分内容。

COMMON:

适用于所有 Vediamo 模块的设置，比如，使用的语言等。

SERVER:

服务器专用设置，比如系统说明路径等。

CAESAR:

CAESAR 性能配置的相关内容。

2.1.3 诊断参数化

DiagServer 需要针对与 ECU 的数据交换采用各种文件：

- CAESAR 文件，包 GBF、FRM 以及 INI 文件。此类文件随 Vediamo 一并安装到 Caesar 子文件夹中，而且必须与使用的 CAESAR 软件相匹配。
- ECU 文件：Cx F（x 表示不同的字母）。定义了 ECU 通讯参数以及诊断服务。

- 系统文件 VSB。定义了配有一个或者多个 ECU 的系统，为 ECU 指定连接编号，定义需要显示给用户的服务列表以及其它各种的规格。详情参见系统配置。
- 仿真文件 SIM：可以在不与 ECU 建立联系的情况下尝试针对 Vediamo 组件使用此类文件。
- Java 程序 CLASS、JAR：可以整合进系统说明 VSB 中的 Java 程序，用于扩展诊断功能。
关于 JAR 的相关说明：
如果启动 JAR 程序，则 Java 必须指定主要类。服务器从 manifest.mf 中获取此类信息，后者必须包括在 JAR 之中。

2.1.4 示例：操作示范

2.1.4.1 关闭服务器 (“Kill”)

最后一个客户端关闭的时候即可自动关闭服务器，或者其它方式也可以关闭服务器（崩溃、通过任务管理器关闭）。如果某个客户端崩溃，则这一过程最长可能需要 20 秒。

但是，对于某些问题来说，可能必须自行结束服务器进程。有两种方法：

- 如果仍然可以控制服务器，那么可以打开服务器菜单（右键点击任务栏中的  图标）然后选择 “Exit”。
- 如果尝试各种方法，甚至是服务器崩溃状态下也无法释放所有 PC 资源，那么可以从任务管理器（“进程” 标签）中终止 “DiagServer” 进程。

2.1.4.2 更换 CAESAR 硬件

Vediamo 支持下列 CAESAR 硬件，可以在后续的 INI 项中激活此类硬件：

- 部件 A (USE_SIPCMCIADriver)
- 部件 C (USE_SIPartCDriver)
- 部件 D (USE_SISerialDriver)
- 部件 E (USE_SIPartEDriver)
- 部件 J (USE_SIPartJDriver)
- 部件 P - eCOM 盒 (USE_SIPartPDriver)
- 部件 X (USE_SIPartXDriver)
- 部件 Y (USE_SIPartYDriver)
- 部件 W (USE_SIPartWDriver)

更换或者安装 CAESAR 硬件的操作步骤如下所示：

- 根据 StartCenter 的硬件规格（正确的街头、系统需求、必要的硬件驱动程序）确定必须遵循的操作要求。
- 如果生产制造提供了检查硬件功能的相关（软件）工具，那么应当使用此类工具来验证功能。
- 采用 INI 编辑器在 “CAESAR” 的 “Vediamo.ini” 文件中激活相关项。
- 如果今后不会使用，那么应当去活所有硬件项。可以保留所有硬件项处于激活状态，但是这样会增加 Vediamo 启动时 CAESAR 确定实际连接硬件时所需时间。
- 重启 Vediamo。识别的 CAESAR 硬件会显示在 Ecoute 应用程序状态窗口内。可以采用 StartCenter 中的 “update Vediamo Server / CAESAR” 功能来测试连接的硬件功能。

2.1.4.3 采用其它固件对 CAESAR 硬件进行闪存处理

CAESAR 软件包括多个不同结构的部分。

比如，CAESAR “master” 与 CAESAR “slave” 是不同的。

“master”是指直接在计算机上运行的 CAESAR 软件部分，可以实现可用的 DIOGENES 数据，或者解释 ECU 发出的回复。

“slave”是直接负责与 ECU 通讯的 CAESAR 硬件/软件。“slave”要求操作所需的“固件”，即，直接存储在 CAESAR 硬件的非易失性存储器中的软件（因此成为“固件”）。

CAESAR 提供了各种不同的固件，主要取决于具体的应用程序。

Vediamo 支持操作下列 CAESAR 固件：

CaesarGo:

该固件通常直接运行在 CAESAR 硬件的非易失性存储器之中，比如，部件 A 或者部件 Y。

此外还支持原有版本的通讯协议（比如，KWFB）。不支持新的 UDS 协议。

TLSlave:

该固件运行在计算机中。计算机速度越快，完成各种通讯任务的速度也越快（比如，闪存处理）。此外还允许同时打开最多 50 个 CAN 通道。该固件还支持“更新的”协议，比如，UDS 等。

BusSim:

一种特殊的固件，用于支持 CAESAR 的 CAN 总线仿真功能。

可以在 StartCenter 或者采用 INI 编辑器（在 CAESAR 中）确定固件可以使用的设置。为了确保变更生效，必须重启诊断服务器。

2.1.4.4 不具备硬件以及许可证条件下尝试 DiagServer（仿真模式）

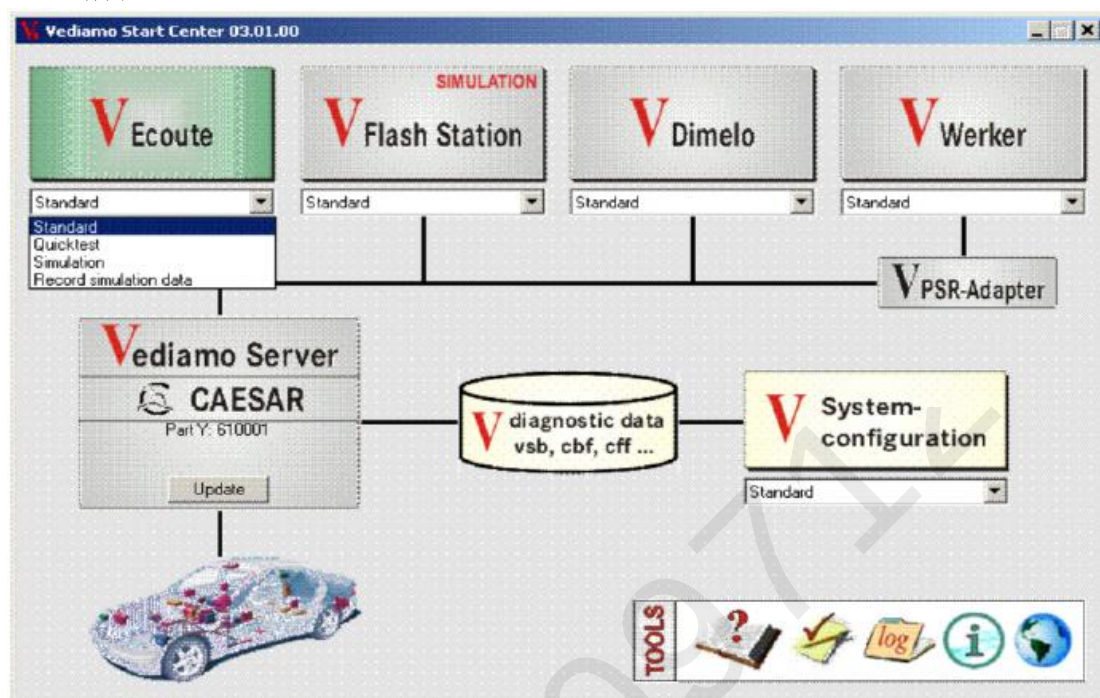
Vediamo 中提供了仿真模式，在不存在实际 ECU 的情况下，与抽象级别上仿真的 ECU 进行通讯。采用 INI 编辑器在“SERVER”部分中通过编辑 INI 项的方式来永久性激活仿真模式。

如果没有有效的硬件或者许可证，那么本软件仅支持仿真模式。这种情况下，Ecout 会在启动的时候显示许可证不可用并且询问是否采用仿真模式操作 Vediamo。

2.2 StartCenter

Vediamo StartCenter 是一种简单的应用程序，可以启动并且管理所有 Vediamo 程序。程序界面包括一个图形显示区域，其中显示了各个 Vediamo 模块以及其它工具。

2.2.1 结构



2.2.2 StartCenter 的功能

启动 Vediamo

点击相应的图标或者按钮即可启动某个程序。对于可以采用命令行参数启动的程序，可以在该程序图标下方通过选项来选择预设的启动参数。

可以对 Vediamo 模块单独授予许可证。如果某个模块未获得许可证，那么会在相应的程序图标上显示“simulation”标签，这表明该模块仅可在仿真模式下工作。对于正在运行的模块，相应的图标会显示绿色背景。

模块启动选项

对于可以采用命令行参数（比如，Ecoute - 快速测试）模块，在模块图标下方会显示可以使用的启动选项列表。

采用文本文件（StartProfile.txt）控制 StartCenter 的行为，该文件包括了需要启动的模块以及命令行参数的所有信息。可以采用文本编辑器添加/扩展该文本文件。如果并未向某个模块提供“Standard”项，那么该模块在 StartCenter 中相关的按钮会处于禁用状态。

该文件的结构如下所示（示例）：

```

Example:
;Vediamo StartProfile file
;Filename can be given as command line parameter of StartCenter.exe
; e.g., "StartCenter.exe MyProfiles.txt"
;
; File format:
; Ecoutex||Profile description||Ecoutex.exe||Command line parameter
; System Configuration||Profile description||SystemConfiguration.exe||Command line parameter
; Worker||Profile description||Worker_Client.exe||Command line parameter

Ecoutex||Standard||Ecoutex.exe
Ecoutex||Quick test||Ecoutex.exe||-K
Ecoutex||Simulation||Ecoutex.exe||/vi "[Server] Simulation 1"
Ecoutex||Record simulation data||Ecoutex.exe||/vi "[Server] Simulation 2"
System Configuration||Standard||SystemConfiguration.exe
System Configuration||With data import/export in text format||SystemConfiguration.exe||/VI
"[SYSTEMCONFIGURATION] EnableImportExport 1"
Worker||Standard||Worker_Client.exe

Lines in the file beginning with semicolons are interpreted as comment lines.

```

配置 Vediamo 程序或者模块

通过 StartCenter 的“Options”图标可以调用 INI 编辑器，该编辑器用于管理以及编辑 INI 参数。编辑并且接受 Vediamo 选项之后，会显示提示信息，说明仅当重启 Vediamo 之后变更才会生效。

更新诊断硬件

通过“update Caesar hardware”功能可以检查选中的硬件是否可以正常工作。所有已检测的 CAESAR 部件状态均在 Vediamo.ini 进行激活，并且可以在适当情况下确定其序列号。可访问部件的编号显示在列表框“Vediamo server/Caesar”字段的左侧区域内。

使用的 CAESAR 硬件（部件...）可以激活的状态下会弹出对话框。

此外，还可以激活以下各项：

- 使用部件 E
- 使用 Mux 模式（Pinmapping）

该模式控制在诊断过程中是否使用 CAESAR 部件 E，以及是否激活 pinmapping。会在使用部件 E 的情况下自动激活 Pinmapping。

- 使用 TL-Slave 或者 CaesarGo 固件

这种情况下会控制已连接 CAESAR slave 使用的固件。详情参见 CAESAR 文档。

对话框底部显示了当前检测到的 CAESAR 硬件的可用信息，比如，线缆 ID 以及安全级别。

对于诊断服务器来说，针对此类设置的变更会立即生效。按下“Update”按钮即可退出对话框，诊断服务器会采用变更的设置重新启动。

编辑诊断数据

“diagnostic data”按钮会在当前设置对话框数据路径中打开资源管理器窗口。

Logfile Administration

选择对话框右下角的“Logfile Administration”图标，则会显示“Trace Analysis”，其中包了所有相关的 Vediamo 日志文件的列表。可以从对话框中选择任意日志文件。采用上下文菜单（鼠标右键）可以针对选中的文件使用下列功能：

- 采用标准编辑器打开文件
- 删除文件
- 将选中的文件路径复制到剪贴板中
- 在资源管理器窗口中到了选中文件的路径
- 将选中文件复制到.zip 压缩文件中

打开 Vediamo 帮助

User Handbook 按钮可以打开并且显示用户文档。

版本信息

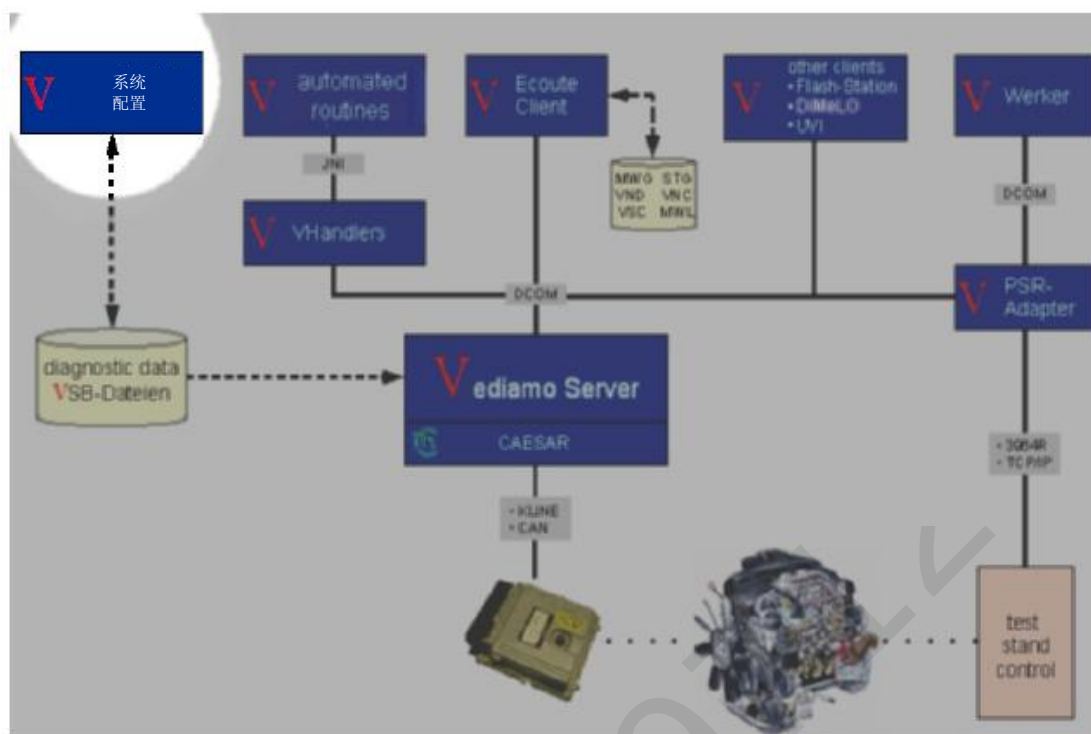
点击 Version / Info 按钮可以确定并且显示 Vediamo 以及 CAESAR 的当前版本。

Vediamo 主页

点击 Vediamo Homepage 按钮可以在默认浏览器中打开 Vediamo 主页。

2Q 1850971v

2.3 系统配置



2.3.1 引言

Vediamo 系统配置主要用于配置 ECU 系统。通过 Vediamo 系统配置，可以实现以下功能：

- 生成系统说明（VSB 文件）
- 过滤掉发动机测试台无法使用的测定结果、功能、驱动器等。过滤掉由于缺少完整车辆系统而导致的错误以及错误环境数据。采用下列符号显示某项的过滤器状况：

☒ 展开过滤项。如果特定项包括无法展开过滤的子项，那么该符号显示为灰色。

☐ 收起过滤项。

☐ 错误项。

☐ 新的、未过滤项。

- 引用 Java 程序
- 创建标准对象
- 搜索各项

除了此类基本功能之外，还可以提供其它功能以确保 ECU 系统运行时的数据一致性：

- 更新系统说明中使用的 CAESAR 数据文件以及 Java 程序文件。
- 系统说明中包含数据的一致性检查。

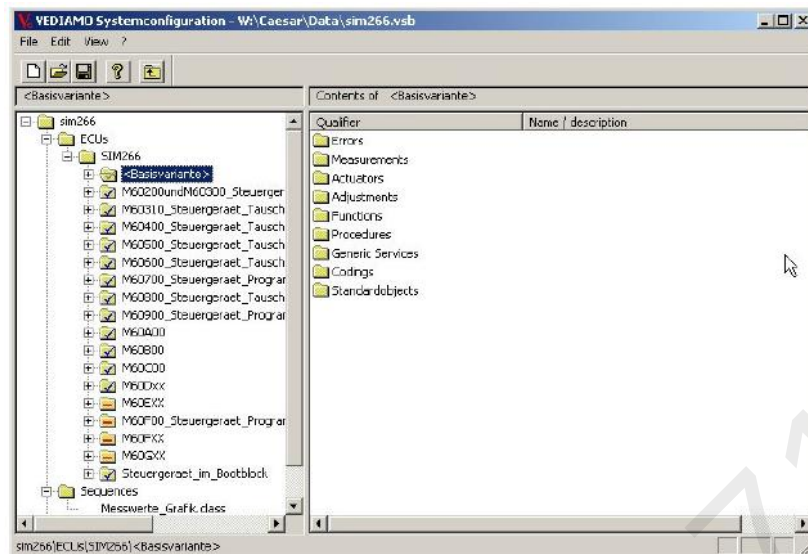
系统配置用户界面支持多种语言。配置数据 Vediamo.ini 的[COMMON]中的 Language 项定义了使用的语言。为了采用系统配置进行工作，必须满足下列条件：

- 正确安装 Vediamo 系统配置部分（尤其是 CAESAR 数据文件）。
无需 CAESAR 硬件。

2.3.2 结构

系统配置属于独立的应用程序。操作此类应用程序无需 CAESAR 硬件以及 Vediamo 诊断服务器。

系统配置用户界面如下图所示：



主窗口的标题栏中显示了正在编辑的当前系统说明的名称。通过名称下方的菜单栏可以选择各种程序功能。菜单栏下方是符号标志栏。对于可以通过菜单栏访问经常使用的某些功能，还可以点击符号执行此类功能。主窗口分为左右两个部分。设置方式类似于 Windows 资源管理器应用程序。左半部系统设置显示为树形结构，右侧则采用列表方式显示了当前选中的元素内容，以及（可用情况下）包括的对应项的说明。点击右半部栏目表头，则会根据当前栏目内容对显示内容采用升序/降序方式进行排序。

根据上下文内容，会在主窗口底部显示包括各种信息的状态行。

菜单栏

提供了下列菜单功能：

File 菜单

- New（新建）
- Open（打开）
- Save（保存）
- Save as（另存为）
- Names of last opened files（最近使用文件的名称）
- Close（关闭）
- Import（导入）
- Export（导出）

Edit 菜单

- Settings（设置）
- Refresh（刷新）
- Check consistency（检查一致性）
- Filter all elements positive（展开所有元素）
- Filter all elements negative（收起所有元素）
- Reverse filter settings（反向过滤器设置）
- Automatically change error environment data（自动变更错误环境数据）

View 菜单

- Symbol bar (符号标志栏)
- Status bar (状态栏)
- Log window (日志窗口)

?菜单

- Help topics (帮助主题)
- System information (系统信息)

树形结构中的每一项均包括上下文菜单, 通过各项列出了可访问的相关功能。将鼠标指针至于该项的上方, 或者点击右键即可显示此类菜单。

符号标志栏

符号标志栏具有以下功能:

 新建文件

 打开文件

 保存文件

 系统配置的相关信息

 回退: 显示主窗口右侧列表中上一级选项的内容。

树形结构

系统说明的树形结构中包括以下各项:

系统

- ECU
 - ECU 采用
 - 采用下列服务的过滤器信息的 ECU:
 - 错误
 - 环境数据
 - 测量结果
 - 驱动器
 - 调整
 - 功能
 - 操作步骤
 - 通用服务
 - 编码
 - 标准对象
 - 标准对象类编码
 - 类编码 1-n 的标准对象
 - 标准对象类闪存处理
 - 分类闪存处理 1-n 的标准对象
 - 标准对象类服务
 - 类服务 1-n 的标准对象
 - 标准对象用户定义类 1
 - 用户定义类 1 的标准对象 1-n
 - ...
 - 标准对象用户定义类 m
 - 用于定义类 m 的标准对象 1-n
 - 程序

○ Java 程序 1-n

- 标准对象
- 标准对象类编码
 - 类编码 1-n 的标准对象
- 标准对象类闪存处理
 - 类闪存处理 1-n 的标准对象
- 标准对象类服务
 - 类服务 1-n 的标准对象
- 标准对象用户定义类 1
 - 用户定义类 1 的标准对象 1-n
- ...
- 标准对象用户定义类 m
 - 用户定义类 m 的标准对象 1-n

诊断服务的分类法

ECU 的基本变体以及变体包括了多种类型服务的列表。一般来说，此类通用服务（即，除特殊服务之外的读取 DTC、变体编码、闪存处理等）采用下列方式进行了分类：

- 测量结果
- 驱动器
- 调整
- 功能
- 操作步骤
- 通用服务

由于 ECU 数据的复杂程度不断增加，因而可能采用其它方式对服务进行分类。从 4.0 版以来，我们根据内部 DIOGENES 使用类采用了替代性分类体系：

- 驱动器
- 调整
- BINARY_ACTUATOR
- BINARY_ADJUSTMENT
- 数据
- DATA_COLLECTION
- 下载
- 功能
- 静态
- 系统
- CYCLIC_DATA
- CYCLIC_DATACOLLECTION
- MEMORY_BLOCK
- 全局
- DIAGJOB
- 安全性
- 会话
- STORED_DATA
- 程序
- IO_CONTROL

在创建新的系统说明的时候，必须确定如何对服务进行分类。这主要取决于文件 Vediamo.ini 中的下列参数：

[SYSTEMCONFIGURATION]

ServiceTypes = AUTO | ASK | VEDIAMO | DIOGENES

ini 参数还可以采用下列数值（通过 StartCenter 或者文本编辑器进行更改）：

- AUTO – 程序自行决定可以采用的类别：
 - 如果是原有 GRIF 系统创建的 CBF 文件，那么采用传统的 Vediamo 分类
 - 如果采用新的 CANDELA 系统创建的数据，那么采用 DIOGENES 分类。
- ASK – 每次向系统中插入新的 ECU 的时候，由用户来决定
- VEDIAMO – 所有服务均包括在 Vediamo 类中
- DIOGENES – 所有服务均包括在 DIOGENES 类中

默认数值为“DIOGENES”。

Ecoute 系统窗口中的树形视图也反映了相关的分类体系。

2.3.3 功能

2.3.3.1 File 菜单

New File（新建文件）

本功能创建新的、空的系统说明。树形结构左侧显示了 New System 项。

Open File（打开文件）

本功能打开现有的系统说明。可以从文件选择窗口中选择需要打开的文件。然后就会读取文件。在读取之前，会提示用户是否采用已使用的 CAESAR 文件方式将文件内容自动对齐。窗口底部状态栏中显示了这一动作的进度（ECU 名称、变体名称、服务名称）。如果 CAESAR CBF 或者 GBF 文件的版本从上次文件编辑之后发生了变化，就会询问用户是否由系统更新各项。此后即可对系统说明进行编辑。

Save File（保存文件）

本功能保存系统说明。如果尚未命名说明，那么在文件选择窗口中必须输入文件名以及存储位置。文件名应当与系统 ID 相同，否则 Vediamo 诊断服务器会发出警告表示文件名与系统 ID 不匹配。

Save As（另存为）

本功能采用另外一个名称存储之前已命名的文件。必须在文件选择窗口中输入文件名以及存储位置。这样即可创建现有系统说明的副本，后续可以开发同一个系统说明的多个变体。

Names of Most Recently Used（MRU）files（最近使用文件的名称）【译者注：与上文不一致】

数据菜单显示最近使用的 4 个文件。点击选中名称即可打开此类文件。

Close（关闭）

本功能关闭系统配置。如果对当前文件数据进行了更改，则用户可以在程序关闭的时候保存数据。

Import/Export（导入/导出）

File 菜单提供了采用文本格式 Import 以及 Export 系统说明的功能。必须在文件选择窗口中输入适当的文件名，然后会采用文本格式读取或者写入文件。文本格式的系统说明的扩展名为.vst（Vediamo System description in Text format）。数据显示的次序与二进制数据版本（.vsb）相同。针对每个对象（即，ECU、变体、驱动器等）创建标题。然后逐行显示相关各项。每一项之前另起一行显示相关说明文本。

Example:

```
ECU-----
ObjectType:6
BasisVersion:3
Filter:0
Qualifier:CR2
Description:Common-Rail 2.x

Version:4
...
DiagPin:0
DeviceNumber:0
DriverTypeCode:KLINE
TrailRequired:0
...
```

注意：

采用文本格式的系统说明进行工作存在较高的出错风险，因为必须严格遵循数据次序。而且在文本文件中很难定位相关错误。

2.3.3.2 Edit 菜单

Settings（设置）

通过选择本菜单，可以进行基本设置：

输出级别：

本程序创建一个名称为 VediamoSysConfLog.txt 的文件，其中会日志记载过程并且记录所有潜在的错误，此外还采用 CAESAR API 网管访问 CAESAR 数据文件。通过选择输出级别可以确定日志信息的详细程度。可选择的选项如下所示：

模块相关：

最小输出

函数相关：

日志记载 CAESAR API 函数的执行情况。

内部函数详情：

日志记载 CAESAR API 函数执行过程中的动作。

最大：

日志记载 CAESAR 提供的所有日志信息。无论采用何种输出级别设置，均会记录发生的所有错误。

CAESAR 文件夹：

这里指定了包括应用的 CAESAR CBF-、GBF-以及驱动程序数据文件的文件夹。可以手动输入或者从菜单中选择文件夹名称。

显示：

在 DIOGENES 中，可以采用唯一 ID（称为修饰符）输入诊断服务。修饰符用于唯一标识某个服务。Vediamo 进程总是使用修饰符。此外还为每个诊断服务提供了名称。该名称不唯一，但是可以有助于用户理解服务的含义。用户可以选择显示修饰符或者描述性名称。为此仅需选择相关设置即可。

Update（刷新）【译者注：与上文不一致】

针对已使用的每个 CAESAR CBF-、GBF-数据文件以及系统描述文件中引用的每个 Java 程序存储器校验和。在诊断会话过程中，Vediamo 诊断服务器借助校验和来验证应用的数据是否与当前配置相匹配，进而可以在适当情况下排除某些系统说明。

本功能会在 CAESAR 文件或者引用的 Java 程序发生变更的时候进行刷新。如果刷新过程中未找到引用的文件，那么用户需要通过文件选择窗口指定相关文件。日志文件以及日志窗口中会记载整个交互过程。

Check consistency（检查一致性）

本功能检查系统说明内容的似然性。在此过程中会验证所有引用的 ECU 变体、服务以及分配给标准对象的服务以及 Java 程序。使用本功能可以确认变更的参数化过程是否导致引用的服务已不存在。日志文件以及日志窗口中会记载整个交互过程。

Filter all elements positive（展开所有元素）

本功能总是引用主窗口右侧的列表内容。在过滤器设置允许的情况下可以将列表中的所有项的过滤器设置设定为展开。

在实现本功能的时候，会弹出 Include subfiles?消息框，其中包括 Yes、No 以及 Cancel 按钮。根据选中的按钮，会决定是否扩展至子文件，或者取消动作。

Filter all elements negative（收起所有元素）

本功能总是引用主窗口右侧的列表内容。在过滤器设置允许的情况下可以将列表中的所有项的过滤器设置设定为收起。

在实现本功能的时候，会弹出 Include subfiles?消息框，其中包括 Yes、No 以及 Cancel 按钮。根据选中的按钮，会决定是否扩展至子文件，或者取消动作。

Reverse filter settings（反向过滤器设置）

本功能总是引用主窗口右侧的列表内容。所有项的过滤器设置均可采用本功能进行反向处理，即，之前展开的过滤或者未过滤项会采用收起过滤，或者之前收起的过滤项会采用展开过滤方式。

Automatically change error environment data（自动变更错误环境数据）

根据参数化的情况，错误环境可能成倍地增加冗余。本功能允许用户在过滤错误环境数据的同时控制程序的表现。如果激活了自动变化，那么会在其它错误、变体中搜索相同的错误环境数据，并且相应地进行更改。激活自动变化选项会在菜单行开始位置显示一个选中符号标志。点击该标志即可打开或者关闭该选项。

2.3.3.3 菜单视图

符号标志栏

可以在窗口上部打开或者关闭符号标志栏。

状态栏

可以在窗口上部打开或者关闭符号标志栏。

日志窗口

可以打开或者关闭显示日志信息的窗口。

2.3.3.4 菜单帮助 “?”

Help topics（帮助主题）

打开 Vediamo 在线用户手册。按 F1 键也可以调用用户手册。

System information（系统信息）

打开某个窗口，其中包括了当前程序版本以及版权信息。

系统配置项

系统：

- 该项代表了系统说明。

上下文菜单的功能：

属性：调用一个窗口，其中包括了下列各项的通用部分：

- 系统说明的名称或者 ID。名称还会以文本方式出现在树形结构项中。
- 说明：主观性描述文字。
- 本系统的 Java 初始化程序名称。可以手动输入名称或者从数据选择窗口中选择名称。Vediamo Ecouste 客户端在载入系统的时候会使用此类信息：一旦激活了“Run initialization routine automatically - after system selection”选项，那么会在每次载入系统的时候执行列出的 Java 程序。如果未使用初始化程序，那么该字段将为空。
- ECU 描述性数据（CBF）以及 ECU 协议数据（GBF）会显示在窗口的 CAESAR 区域内。程序管理此类项，用户无法手动更改。

系统中的其它项：

- ECU

本项下方列出了系统中诊断的各个 ECU。

上下文菜单功能：

- 新 ECU：本功能调用选择窗口，其中包括了已使用 CAESAR 数据文件所涉及的各种可用 ECU。选中某个设备，则会创建该 ECU 的一条新项。借助 CBF 数据文件内容，可以自动生成相应的变体以及服务项。屏幕下方状态栏中记载了此类动作的进程。选中新的 ECU，那么会出现包括下列各项的属性窗口：

- 说明：主观性描述文字。
- 驱动程序类型：从 CAESAR 2.6 开始，可以对 ECU 的多个 GPD 引用进行参数化。该设置在载入系的时候（比如，KLINE、CANLS 等）选择诊断服务器使用的相关项。
- 连接编号：与该 ECU 通讯的 CAESAR 资源编号。在诊断服务器运行时载入系统，那么会采用该编号打开与 ECU 的通道。如果没有可访问的资源编号，那么必须从客户端应用程序打开 ECU 的通道。例外情况：如果输入 0 作为访问编号（=默认设置），则采用第一个可用资源来打开一个通道。
- 设备编号：仅可在网关成员 ECU 中提供该字段。根据说明的相关内容，指定了成员设备编号。对于网关成员 ECU 来说，将网关设备编号作为连接编号。
- Ecoute 的诊断会话过程中，在清空错误的时候，会使用“Shut-down cycle for error deletion”选项。如果激活了选项，用户会调用关闭启动装置，然后在清空错误过程之后再次打开。默认情况下，会激活该选项。
- Ecoute 的诊断会话过程中，会使用“Shut-down cycle for flashing”选项进行闪存处理。如果激活了选项，用户会调用关闭启动装置，然后在闪存处理之后再次打开。默认情况下，会激活该选项。
- Ecoute 的诊断会话过程中，会使用“Shut-down cycle for coding”选项进行编码。如果激活了选项，用户会调用关闭启动装置，然后在编码之后再次打开。默认情况下，会激活该选项。
- “Maximum number of areas to flash simultaneously”字段控制 Ecoute 中多少个闪存区域，用户可以同时变更然后进行闪存处理。默认数值为 1。
- DDLID 字段规定了需要动态定义的服务 ID。该规格仅当 ECU 协议支持 DDLID 的情况下才具有相关性。默认数值为“none”，即，诊断服务器并未针对该 ECU 采用任何 DDLID。
- 该 ECU 的初始化服务 ID。可以针对对话框分配各种服务。Vediamo Ecoute 客户端在与相应的 ECU 建立联系的时候使用此类信息：如果激活了“conduct initialization services automatically – upon contact”选项，则在与相应 ECU 建立每个更新联系的过程中，执行输入的各种服务。还可以在对话框内将 Java 程序指定为初始化服务。
- 可以在“user defined qualifier”字段中指定单独的修饰符，此类修饰符用于识别 ECU（比如，在 Ecoute 中）。修饰符必须唯一，而且在诊断数据中不得使用已有的 ECU 修饰符。

ECU 中的其它项：

- ECU

该项代表了一个 ECU。该项下方列出了该 ECU 的各个变体。

上下文菜单功能：

- 删除：从系统说明中完全删除相应的 ECU 项。
- 属性：访问与添加新 ECU 相同的属性窗口。
- 复制 ECU：复制相应的 ECU 项，创建第二个相同项。采用运行编号(1-n)提供 ECU 修饰符，以便区分相同项。

ECU 中的其它项：

变体：

该项代表了 ECU 的单个变体。在添加新的 ECU 的时候，程序会根据 CBF 文件的内容自动生成变体项。变体属于可过滤项。点击该项的符号标志即可更改过滤器设置。如果从诊断中排除了某个变体，则应当将过滤器设置设定为收起。添加新的 ECU 时候，所有变体的默认过滤器设置均为展开。

上下文菜单功能：

- 属性：
调用属性窗口，采用变体的下列设置：
- 说明：主观性描述文字。
- 此类 ECU 变体的初始化服务标识符。可以将服务分配给某个对话框。Vediamo Ecoute 客户端会在与相应 ECU 建立联系的时候使用此类信息，并且指定变体：如果激活了“conduct initialization services automatically - upon contact”选项，那么在完成 ECU 项指定的所有服务之后，在与相应 ECU 建立更新联系过程中执行指定的服务。如果未识别特定变体，那么会采用基本变体项。还可以在对话框中将 Java 程序分配为初始化服务。
- ID 块信息：本功能提供了一个对话框，其中可以分配显示其它 ECU 属性的服务。此类属性会在读取 ID 块时显示在 Ecoute 中，或者显示在闪存对话框中。对话框左侧列出了可使用的诊断服务。一旦识别了某个服务，那么可以采用“Add”按钮分配该服务。点击按钮之后该服务会添加到右侧显示窗口的列表中。用户必须编辑右侧列表中的“ID”栏。此类说明会连同指定服务的结果显示在 Ecoute 中。

变体中的其它项：

Filter entries for errors, services and variant codings:

这些项属于可过滤项。点击该项符号标志，则可以设定或者清除过滤器设置。如果从诊断中排除了某个服务，那么应当收起过滤器设置。新的应用程序中所有服务的默认过滤器设置均为展开。

可以采用上下文菜单针对每项打开一个窗口。默认使用 CAESAR 提供的说明。可以针对所有驱动器服务以及所有通用服务（可以用作 Ecoute 中的驱动器）指定或者选择“actuation mode off”服务。各种 DIOGENES 参数化诊断服务指定为“actuation mode off”服务。驱动器以及受影响的通用服务上会显示切换符号标志，以便将此类服务与系统说明中的其它服务区分开来。输入为“actuation mode off”的服务，该符号标志显示为黄色，其它显示为蓝色。属性结构中显示了所有 CAESAR/DIOGENES 参数化服务（比如，准备/展示的相关信息）的其它信息。

如果对下列各项进行了参数化，则可以对其进行过滤：

- 环境数据错误

上下文菜单功能：

在“error”项中，可以采用“delete preconditions for error memory”来输入清空错误的前置条件。执行本功能之后，可以在对话框中为服务指定前置条件。Ecoute 在每次删除错误内存之前会执行此类前置条件：基本上，会首先执行输入到基本变体的前置条件。如果识别出特定的变体，那么在满足基本变体的前置条件之后会执行该变体指定的特定前置条件。

- 测量结果
- 驱动器
- 调整
- 功能

- 操作步骤
- 通用服务
- 变体编码

本项中列出了可过滤编码服务。每种编码服务均包括额外的子项，其中列出了相关的代码片段及其可能选取的数值。外部数据文件（结尾：.ccf）参数化的片段采用后缀名（ex）进行标识。此类子项仅供参考，不可对其进行过滤。

上下文菜单功能：

此处采用 ECU 服务形式规定了变体编码的前置条件以及后置条件。无论归属于哪个 Vediamo 类，每个条件均可指定一个或者多个服务。还可以指定 Java 程序。一旦在 Ecoute 中打开编码对话框，则可以执行指定给前置条件的服务或者 Java 程序。一旦在 Ecoute 中关闭编码对话框，则可以执行指定给后置条件的服务或者 Java 程序。

如果指定了前置条件或者后置条件，那么树形控件的“Coding”项中就会出现两个新的子文件夹，即，“Preconditions”以及“Postconditions”。这两个子文件夹依次包括相应的服务。仅当收集到实际前置条件以及后置条件的情况下才会出现这两种子文件夹。

- 标准对象

本项中列出了该变体执行的变体特定标准对象。标准对象被分配给以下各类：

- 服务文件夹：
可以存储“Services”类的标准对象。
- 闪存处理文件夹：
可以存储“Flashing”类的标准对象。
- 编码文件夹：
可以存储“Coding”类的标准对象。

采用“Standard objects”文件夹的上下文菜单可以创建用户定义的标准对象类。

新的标准对象类：可以在此创建其它用户定义标准对象类。针对每个用户定义标准对象类，在“Standard objects”中创建新的文件夹。

“Services”、“Flashing”、“Coding”文件夹以及用户定义标准对象类拥有具备下列属性的上下文菜单：

- 新的标准对象：

本功能访问一个窗口，必须在该窗口内输入新的标准对象的名称。或者，还可以从现有名称列表中进行选择。这样就创建了采用该名称的新项。然后会显示具有下列各项的属性窗口：

说明：主观性描述文字。

服务：必须在此处指定标准对象中隐藏的服务。此类服务可以是 Java 程序、操作步骤或者函数。还可以指定通用服务，前提是对此类服务的默认准备方式进行参数化。可以手动输入服务，还可以从相关服务中进行选择。

- 删除：删除标准对象。

- 属性：此处可以输入标准对象类的说明。

- 对于标准对象以及标准对象类来说，在相应的上下文菜单中提供了复制+粘贴功能。

ECU、变体以及各种服务过滤项的上下文菜单项中包括了“Import”以及“Export”两种功能。可用的功能具有下列特性：

- 在选择窗口中导出/导入单个选定的服务

- 在 ECU、变体以及函数级别的树形视图中可以多重选择
- 导出到面向表的文本文件中。

导出功能：

在上下文菜单中选择导出功能之后，会打开一个选择窗口，其中可以根据过滤器信息来选择需要导出的服务。选择服务之后，会调用文件保存对话框，必须在其中指定文件名称。此外，还必须显示文件名的结尾部分。推荐的文件名默认结尾有助于在树形视图的选择级别区分各种文件：

- ECU 级别结尾为.vds
- 变体级别结尾为.vdv (Vediamo 数据变体)
- 函数级别结尾为.vdf (Vediamo 数据函数)

然后创建文本文件，其中以导出的每个服务项各占一行。一行由多栏组成，其中包括：

- ECU
- 变体
- 函数
- 修饰符
- 说明
- 过滤器

采用制表符分隔各栏，这样就可以采用电子表格程序等读取以及处理此类文件。

导入功能：

通过上下文菜单的“import”功能，可以将具有三种文件结尾的 ECU、变体以及上述文本格式功能说明：

- .vds
- .vdv (Vediamo 数据变体)
- .vdf (Vediamo 数据功能)

导入到系统说明中。选择导入文件之后，会弹出“import”对话框。用户可以在此处选择需要导入的服务类型，主要取决于过滤器信息 (“all”、“positively filtered”、“negatively filtered”、“new/not filtered”)。此外还可以实现下列功能：

1) 无需导入即可创建某项活动：

点击“activity log”按钮，即可完成的整个系统的变更检查。文本窗口中列出了导入导致的变更。此外，用户还可以存储文本窗口的内容以便将其日志记载到一个文件中。默认情况下，文件存储在...\ DATA \ VediamoData \ [current system name] \ yyyy-mm-tt_hh-mm_Import_[current system name] .log 中。

2) 选中服务的实际导入：

点击“import”按钮，就会将数据实际导入及/或写入现有 vsb 中。

- 程序

该项中列出了可以随本系统一同执行的系统特定 Java 程序。

上下文功能：

- 新程序

本功能调用数据选择窗口，其中可以选择 Java 程序。如果选择了 Java 程序，则会创建新项。选择之后，会显示窗口，在其中可以指定程序属性。该窗口还可用于系统、ECU 以及 ECU 变体的初始化程序。可以提供下列各项：

- 说明：主观性描述文字。
- 命令行：执行 Java 程序是需要传送给该程序的命令行参数。

- 原始路径：此处显示了 Java 程序的完整的原始路径。可以使用按钮来显示数据交换对话框，在其中更改此类路径。该项仅用于显示信息。注册的路径不必与运行时 Vediamo 诊断服务器的各种条件保持一致。

- 在 Ecouste 中同步执行 Java 程序：该项控制 Ecouste 应用程序是否在前台（同步）或者后台执行。设置还会影响标准对象，其中可以输入相应的 Java 程序。默认情况下，Ecouste 中同步执行了初始化程序，采用异步方式执行其它 Java 程序。

每个 Java 程序在“routines”下设有一项。此类项的上下文菜单包括下列项：

- 删除：从系统中删除 Java 程序。
- 属性：在此处输入 Java 程序的描述性文本。

- 标准对象

本项中列出了系统可以执行的系统特定标准对象。

- 将下列分类指定给标准对象：

- 服务文件夹：
可以存储“Services”类的标准对象。
- 闪存处理文件夹：
可以存储“Flashing”类的标准对象。
- 编码文件夹：
可以存储“Coding”类的标准对象。
- 采用“Standard objects”文件夹上下文菜单创建用户定义标准对象类。新的标准对象类：可以在此处创建其它用户定义标准对象类。在“Standard objects”中针对每个用户定义标准对象创建新的文件夹。
- “Services”、“Flashing”、“Coding”以及用户定义标准对象类具有下列属性的上下文菜单：
- 新的标准对象：

本功能访问一个窗口，必须在该窗口内输入新的标准对象的名称。或者，还可以从现有名称列表中进行选择。这样就创建了采用该名称的新项。然后会显示具有下列各项的属性窗口：

- 说明：主观性描述文字。
- 服务：必须在此处指定标准对象中隐藏的服务。此类服务可以是 Java 程序、操作步骤或者函数。还可以指定通用服务，前提是对此类服务的默认准备方式进行参数化。可以手动输入服务，还可以从相关服务中进行选择。
- 删除：删除标准对象。
- 属性：此处可以输入标准对象类的说明。
- 对于标准对象以及标准对象类来说，在相应的上下文菜单中提供了复制+粘贴功能。

- 在窗口左半部以及右半部均可编辑所有项。

2.3.4 采用系统配置进行工作

- 创建新的系统说明；
- 编辑现有的系统说明；
- 更新系统说明的内容；
- 对系统说明内容进行一致性检查。

2.3.4.1 创建新的系统说明

采用菜单项 **New** 或者符号标志栏中的符号标志  生成新的、空的系统说明。然后指定新系统的属性：将鼠标放置在屏幕左半边“**New system**”项上方，点击右键呼出菜单。这样会弹出属性窗口显示名称、说明以及（适当情况）可以输入的 Java 初始化程序。

然后即可以指定系统中需要诊断的 ECU。从上下方菜单中的 ECU 项选择 **New ECU** 功能。选中 ECU 之后，程序会生成相关的变体以及服务项。此后，即可指定新的 ECU 的属性。默认情况下，会设置所有变体的过滤器属性以及新的 ECU 项的服务。点击该项左侧的符号标志即可更改过滤器信息。这一过程中应当使用 **Select all** 功能、**Delete all** 选项、**Reverse** 选项、**Automatic erroneous data modification** 以及符号标志栏中的  符号标志。必要的情况下还可以输入 Java 程序以及标准对象。

完成后，通过菜单项或者符号标志栏中的  符号标志来保存系统说明。

2.3.4.2 采用预存系统说明进行工作

首先，采用菜单项 **Open** 或者符号标志栏中的  符号标志打开系统说明文件，或者点击菜单中最近打开文件名称。然后可以按照上述方式操作该文件。

完成变更之后，通过菜单项或者符号标志栏中的  符号标志来保存系统说明。

2.3.4.3 更新系统说明内容

系统说明文件中针对使用的每个 CAESAR CBF-、GBF 数据文件以及引用的每个 Java 程序存储相应的校验和。在诊断会话过程中，Vediamo 诊断服务器借助校验和来验证应用的数据是否与当前配置相匹配，进而可以在适当情况下排除某些系统说明。如果引用的数据发生了变更，那么必须更新系统说明。操作步骤如下所示：打开文件，执行菜单项 **Refresh**，然后保存文件。

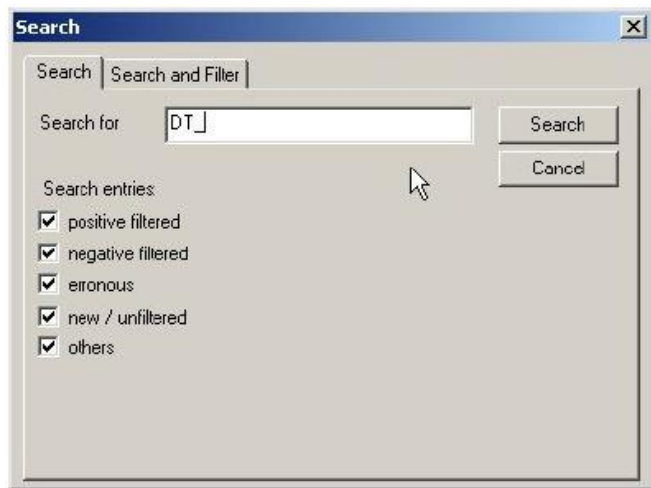
2.3.4.4 检查系统说明的内容一致性

如果参数化过程中的变化导致去除了某项服务，或者更改了此类服务的 ID，那么系统说明就会出现不一致的情况。比如，可能会引用不存在的服务。一致性检查操作步骤如下所示：打开文件，执行菜单功能 **Verify consistency**。日志文件以及日志窗口内记载了一致性审计过程中发现的各种错误。过滤器设置会去除不存在的原有变体以及服务，并且采用叹号表示受影响项。此外还会日志记载发现引用了不存在服务的标准对象实例。在日志窗口中可以根据记录来审查以及纠正各种问题。此后应当再次保存系统说明文件。

2.3.4.5 搜索功能

配置工具具有搜索功能，可以在相应的窗口内查找名称或者部分名称。通过 **Edit** 菜单中的 **Search** 菜单项即可进行文本搜索，详情参见下列 Ecouste-客户端实例。还可以采用快捷键 **Ctrl+F** 进行搜索。

选择“**Search**”功能之后，在对话框中输入需要搜索的文本：



对话框会保持打开状态（非模态对话框），点击“Cancel”即可关闭对话框。输入搜索条件之后，点击“Continue Search”按钮开始搜索或者继续搜索。

搜索系统配置工具中的系统树形控件。由上至下对窗口内容进行搜索。搜索文本的时候会忽略大小写问题。如果搜索达到结尾，则会从窗口顶部继续进行搜索。如果未找到搜索项，则会弹出具有相应通知的信息框。如果找到搜索项，则

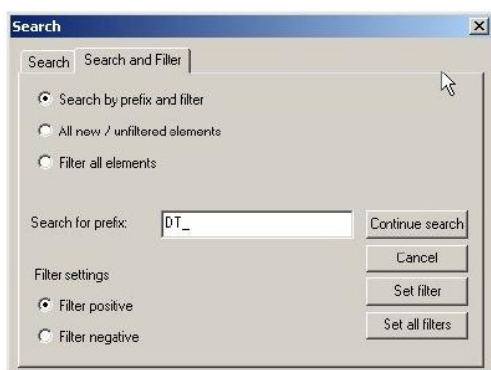
会激活相应的窗口行。点击“Search”对话框中的“Continue Search”按钮即可继续搜索。

采用“Search entries”下方的五个选项，可以将文本搜索限定为具有特定状态的各项。上述各项分为五组：

- Positive filtered（展开过滤）
- Negative filtered（收起过滤）
- Erroneous（错误）
- New/unfiltered（新/未过滤）
- Others（其它）：不具有过滤器状态的所有项，比如，ECU、标准对象分类、Java 程序等。

为了搜索所有项，必须标记所有选项。默认功能是发起 Search 功能。

对话框上部的注册器：用于切换至 Search and Filter 功能。



如果激活了 Search by prefix and filter 选项按钮，则会搜索特定的前缀，并且对找到的各项进行适当过滤。

如果激活了 All new / unfiltered elements 或者 Filter all elements 按钮，则会去活 Search for prefix and filter 特别需要的选项。通过激活 Filter settings，如果激活了 All new / unfiltered elements，那么会依照过滤器设置过滤掉文件中所有新的或者未过滤元素。

左下部的两个单选按钮用于选择是否展开或者收起找到的过滤项。

Set filter 按钮会激活当前服务的预期过滤器设置，然后自动继续搜索。

采用 Set all filters 按钮的情况下，会扫描整个系统说明的修饰符中包含指定前缀的服务。一旦找到此类服务，则会自动设定预期过滤器设置。

Search 以及 Search and filter 之间的搜索算法差异在于：

- Search 会采取任意方式检查包含特定字符串的服务修饰符。
- 比如，如果搜索字符串为 DNU，那么会找到修饰符为 “Test_DNU_1” 与 “DNU_ABC” 的服务。
- Search and filter 仅会搜索服务修饰符前缀符合要求的字符串。
- 比如，如果搜索字符串为 DNU，那么会忽略修饰符为 “Test_DNU_1” 的服务，而找出修饰符为 “DNU_ABC” 的服务。

2.3.4.6 批量模式

可以采用批量模式打开系统配置，以便自动更新以及检查系统说明文件。这种情况下可以采用命令行参数来引导执行程序。

可以使用下列命令行参数（[]方括号中的是可选参数）：

文件名

关注的文件的名称。最初会在实际工作文件夹中搜索文件，如果未找到，就会在 “[SERVER] SystemPfad” 中的 VEDIAMO.ini 文件中输入的文件夹中进行搜索。本参数规范属于强制规定。

[-a]

刷新：

如果给定了本参数，则会重新计算并且重新输入 StartCenter 引用 CBF、GBF 以及 Java 程序文件中所有校验和。如果发生错误，那么会在日志文件中输入错误消息。

不使用本参数则无法进行更新。

[-k]

一致性检查：

如果给定了本参数，会检查 DIOGENES 文件以及 VEDIAMO 服务类中包含的所有变体以及服务。如果发生错误，那么会在日志文件中输入错误消息。

不使用本参数则无法进行一致性检查。

处理新的变体以及服务：

[-n+]

向系统说明以及过滤展开项中添加了 DIOGENES 参数化的新变体以及服务，即，在服务器运行过程中向客户端提供此类变体以及服务。

[-n-]

向系统说明以及过滤收起项中添加了 DIOGENES 参数化的新变体以及服务，因此不会在服务器运行过程中向客户端提供此类变体以及服务。

如果并未指定参数 **n+** 或者 **n-**，则会忽略新的变体或者服务。

如果同时指定了两个参数，那么会在日志文件中输入错误消息，并且放弃程序。

[-t]

测试：

如果指定了本参数，那么在编辑之后不会保存系统说明，仅会创建日志文件。

[-l Log filename [+]]

记录结果以及错误消息所在日志文件的名称。如果给定了参数“+”，那么不会覆盖现有的文件，而是会在此类文件中附加所有新的日志信息。

如果并未给定日志文件名，那么会将日志信息处理为交互性操作，并且写入到 **VediamoSysConfLog.txt** 文件之中。

[-s+ Prefix]

定义：过滤具有给定前缀展开项所有新的服务。

[-s- Prefix]

定义：过滤具有给定前缀收起项所有新的服务。

追随过滤方式仅会影响新添加的服务，不会改变之前提供的相同前缀名的服务。可以多次指定 **-s** 命令行参数，以便过滤各种前缀，比如

示例：

-s- DNU -s- WVC

-s 参数的优先级最高。比如，如果给定了 **-n+** (= 新服务过滤器展开项) 以及 **-s- DNU**，那么会具有过滤收起项的新服务 **DNU_xxx**，但是依照 **-n+**，则应当过滤展开项。

批量操作过程中无需用户交互。这种情况下会禁用显示交互操作错误的对话框（唯一的例外：发生日志文件相关的严重错误）。错误会记载到日志文件中。

一旦完成系统说明的编辑，则会结束程序。为此，会针对调用进程生成返回代码。可以采用下列返回代码：

0=程序结束而且无错误

1=发生严重错误，比如：

- 无法读取指定的系统说明数据。
- 无法写入指定的系统说明数据。
- 未知/遗失/错误的命令行参数。
- 其它错误，比如，CAESAR 初始化过程中发生的错误、未找到 ECU 参数化等等。详情从日志文件中获取。
- 日志文件打开/创建/写入过程中发生错误。

2.3.4.7 设定命令行参数之外的选项

系统配置启动的时候，可以在命令行中指定 **Vediamo.ini** 文件的所有键值。指定键值在程序会话期间有效。

命令行语法如下所示：

System Configuration.exe /VI "[Section] key value ... [Section] key value ..."

选项 **/VI** 或者 **-VI** 用于区分 **Vediamo.ini** 键值与其它（批量模式）参数。需要覆盖的键值列表必须包括在引号之中。

包含 **cbf** 文件的文件夹

一旦启动配置工具，则会向诊断服务器发送模拟信号，后首先应用 “[CAESAR] CBFPFAD” 中列出的 **Vediamo.ini** 文件所在目录中输入的 **.cbf** 文件。

如果需要打开当前系统文件夹中并未包含的系统说明文件，那么就需要检查.cbf 文件是否存储在这个文件夹中。如果是，那么在采用诊断服务器情况下，由于没有载入当前的.cbf 文件，则会在新的文件夹中载入.cbf 文件。

菜单项 Edit / Return to the standard system path 会将.cbf 路径重置为默认数值，比如，创建新的系统说明。

2.3.5 特别特性

2.3.5.1 服务过滤器

可以采用多种方式过滤各种服务。通过 Vediamo.ini 中的设置来过滤各种服务。

[Ecoute]Global Filter

修饰符以过滤的收起项开头的服务（前提是并未采用[CAESAR]UseServiceTypes 对其进行过滤）。

从菜单中选择适当命令可以取消收起项过滤。

[CAESAR]UseServiceTypes

如果本项数值为“STANDARD”，VSB 中不会接受以下服务：

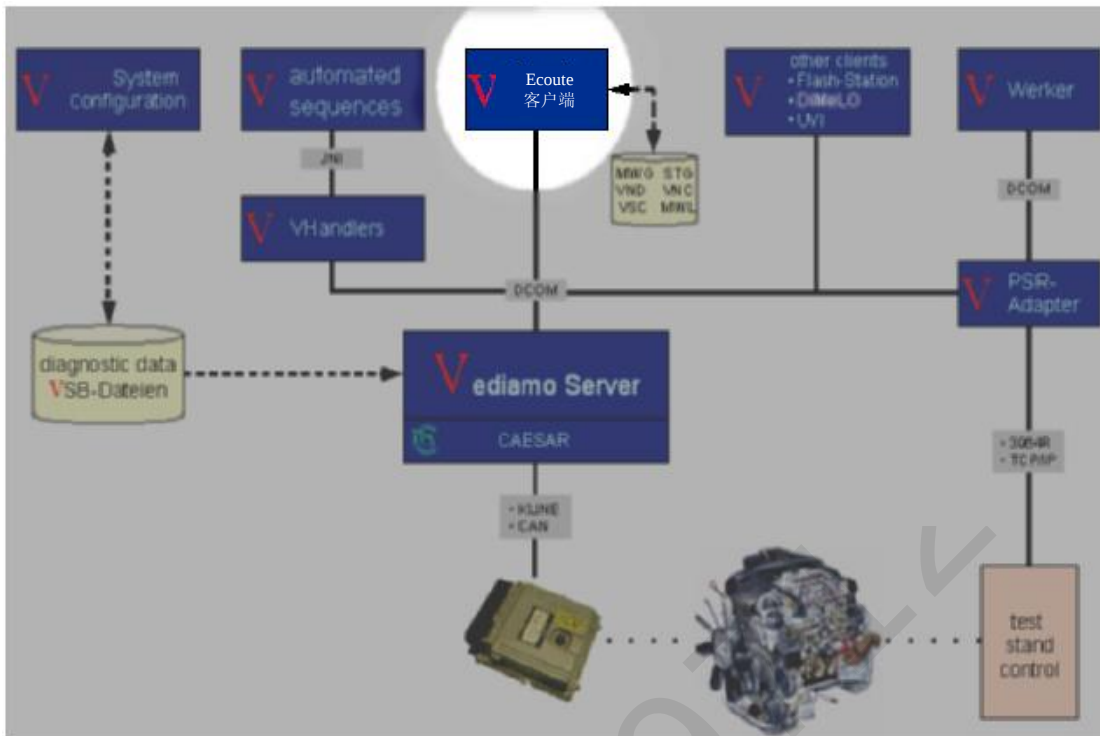
- DST_SYSTEM
- DST_ENVIRONMENT_DATA
- DST_GLOBAL, DST_NEGRESP
- DST_BINARY_ACTUATOR_INP
- DST_BINARY_ADJUSTMENT_INP

上述服务按照不存在的方式来处理。

2.3.6 配置（INI 参数）

INI 文件中的系统配置仅具有 4 个参数，均与日志存储有关。详情参见 INI 编辑器。

2.4 Ecouté



2.4.1 引言

Ecouté 应用程序通过综合交互界面为用户提供所有可用的诊断服务器功能。可以按照预期方式配置、采纳以及安排的每个任务均拥有特定的窗口。通过 Java 程序可以实现特殊、非标准任务定制解决方案，并且将其整合进程序之中。

Ecouté 在其功能范围内，主要用于在发动机测试台中操作 ECU 系统，在实验室装置中测试 ECU，或者用于车辆诊断（比如，快速检测）。

接口结构章节中简要介绍了接口。Ecouté 文件章节中概要总结了 Ecouté 所需的各种文件。Ecouté 功能中详细介绍了各种功能以及动作。最后还介绍了 Ecouté 菜单以及快捷键。

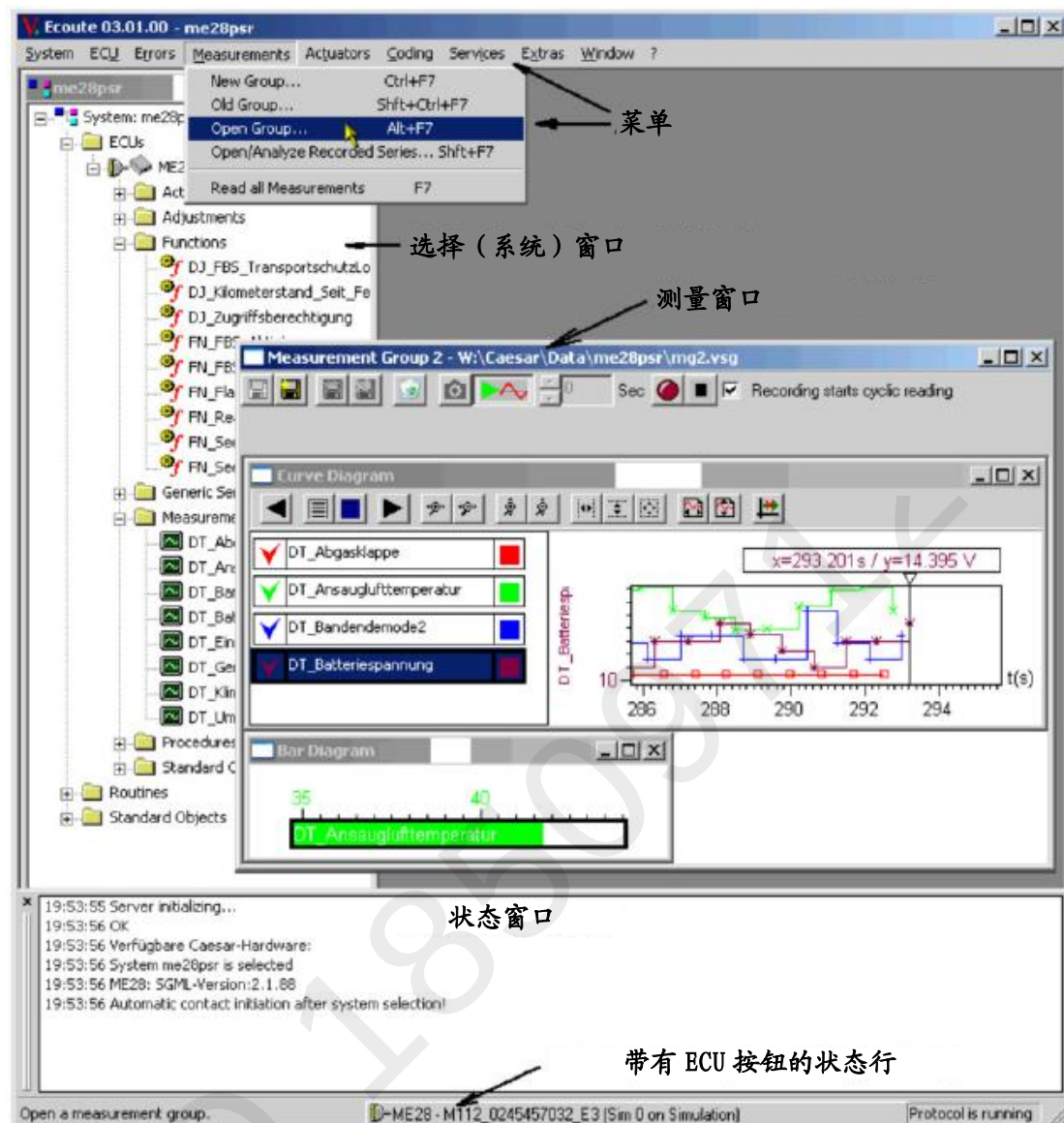
Ecouté 用户界面支持多种语言。Vediamo.ini 配置文件的[COMMON]中 Language 项中规定了所用的语言。3.1 版支持德语（DE）以及英语（EN）。

Ecouté 工作的相关要求

为了使用 Ecouté 建立与 ECU 的通讯，必须满足下列需求：

- 正确安装 Vediamo 以及 CAESAR 硬件驱动程序。
- 安装以及注册的服务器必须与使用的 Ecouté 版本相同。仅需正确执行安装程序即可满足此项需求。
- ECU 所需的 CAESAR 参数化。
- 测试仪（蔡园镇 CAESAR 硬件的诊断服务器）以及 ECU 之间建立正确的物理连接。
- CAESAR 硬件电源（CAESAR 部件 D 以及部件 P-eCOM 必须采用外部供电）。
- ECU 电源。某些 ECU 除了工作电源之外还需要点火线路电源。
- 提供 Vediamo 系统说明，其中包括需要诊断的 ECU。采用 Vediamo 系统配置生成系统说明。如果尚未生成系统说明，那么可以对具有 Vediamo 系统说明（VSB）的 ECU 进行诊断。对于 CAESAR CBF 文件中包括的每个 ECU 来说，Ecouté 会提供与诊断同名的系统。
- 至少一个已实现 CAESAR 硬件组件的许可证。

2.4.2 GUI 结构



Ecout 用户界面

下表列出了最重要的 Ecout 窗口概况信息：

Ecout 窗口

选择窗口	● 显示选定的 ECU 系统 ● 显示 ECU 名称/ECU 变体名称 ● 显示 ECU 与 CAESAR 硬件的连接 ● 显示标准对象（具有解锁、ECU 复位等特殊功能的选定服务）
输出窗口	● 显示状态消息 ● 显示函数/Java 程序等产生的结果
状态行	● 说明当前选定菜单项 ● 显示以及变更 ECU 联系状态的按钮 ● 显示日志文件
测量窗口（服务组）	● 显示以及说明具有下列功能的测量： ○ 文本（表格）以及模拟（条形图）数值显示 ○ 时间依赖性（曲线图） ○ 一系列测量的记录 ○ 一系列测量的分析记录
控制器组（*）	● 控制器的显示以及操作/调整 ○ 采用控制器/调整的 ECU 名称 ○ 名称 ○ 实际数值 ○ 控制器单位 ○ 差异控制器/调整。采用（*）指定调整 ○ 改变控制器/调整状态的控制装置
错误窗口（*）	● 显示包括以下信息的 ECU 错误： ○ 错误代码（比如，B1470） ○ 错误文本 ○ 是否为最新错误的信息 ○ 是否存储了错误的信息 ○ 是否打开 MIL 的信息 ○ 采用下列信息心室错误环境： ○ 错误环境数值名称 ○ 说明（比如，Geberbruch） ○ 错误环境数值的单位
变体编码（*）	● 显示以及选择变体编码的服务名称 ● 显示以及选择编码片段 ● 显示以及选择编码片段数值 ● 编码字符串（十六进制以及十进制）
闪存处理窗口（*）	● 显示以及选择闪存（区域、含义、闪存键） ● 开始 ECU 闪存进程
跟踪窗口（*）	● 显示 PC 与 ECU 之间的通讯数据（字节或者块）
	（*）未包括在图中

系统或者选择窗口

系统或者选择窗口具有一项特殊的功能。可以采用分级树形结构显示所有 ECU 及其诊断服务,此外还显示了标准对象以及 Java 程序。双击(或者<Enter>)可以激活每个元素(ECU: 建立或者终止联系, 服务以及 Java 程序: 开始执行)。

对于树形结构中的每个对象(具备相关说明), 此类说明均可在树形窗口中(右键-上下文菜单-树形)显示。

对于 ECU 来说, 此类内容显示在树形窗口中。对于服务以及 Java 程序来说, 与服务/Java 程序最具相关性的信息会显示在一个简单窗口中。

服务分类

ECU 的诊断服务分为若干组。分组方式一般如下所示:

- 测量 – 从 ECU 中读取数值的服务。
- 调整 – 允许针对特定 ECU 参数采用固定设置。
- 控制器 – 与调整类似, 但是不会将设置数值永久性存储在 ECU 中。
- 功能 – 针对 ECU 执行动作, 提供结果。
- 操作步骤 – 与功能类似, 但是不提供任何结果。
- 通用服务 – 其它所有服务。

或者, 还可以根据 DIOGENES 服务类型对服务进行分组。服务的分类方式主要取决于系统说明文件。在使用较新的 ECU 数据的时候, 这种方式可能更好, 因为 ECU 数据趋向于越来越复杂。

注意:

通过调整服务设定的参数会永久性改变 ECU。

关闭点火器不会将数值重置为默认数值。

服务专用的过滤器

可以采用不同方式过滤各种服务(即, 在系统窗口显示中进行掩码处理)。通过设置 Vediamo.ini 即/或系统配置来过滤各种服务:

- [Ecoute]UseFilters
- [Ecoute]GlobalFilter
- [CAESAR]UseServiceTypes

以下列出了不同配置以及预期结果：

[CAESAR] UseServiceTypes	[ECOUTE] UseFilters	VSB/DIOGENES 系统	预期结果
(无关)	0	VSB 系统	无论服务具有何种过滤器特性（根据系统配置中的设置），VSB 中包括的各种服务均采用无过滤方式显示，
(无关)	1	VSB 系统	VSB 中仅包括此类服务，而且显示了展开过滤项。
全部	0	DIOGENES 系统	显示 DIOGENES 系统 (CBF) 中包括的所有服务。
标准	0	DIOGENES 系统	不显示的服务类型： ● DST_SYSTEM ● DST_ENVIRONMENT_DATA ● DST_GLOBAL, DST_NEGRESP ● DST_BINARY_ACTUATOR_INP ● DST_BINARY_ADJUSTMENT_INP
全部	1	DIOGENES 系统	不会显示具有以[ECOUTE]GlobalFilter 中包括的前缀开头的修饰符的服务。
标准	1	DIOGENES 系统	不会显示下列服务： ● 服务类型： ○ DST_SYSTEM ○ DST_ENVIRONMENT_DATA ○ DST_GLOBAL, DST_NEGRESP ○ DST_BINARY_ACTUATOR_INP ○ DST_BINARY_ADJUSTMENT_INP ● 具有以[ECOUTE]GlobalFilter 中包括的前缀开头的修饰符的服务

尽管存在通用的相关性，但是仍然可以在没有选择窗口的情况下工作。如果通常仅采用预先准备的测量以及控制器窗口，那么可以关闭选择窗口（菜单 **Window / System Window** 或者 **CTRL-S**）。这种状态会存储到配置中（Vediamo.ini）。下次启动 Ecouté 的时候，窗口会处于关闭状态。采用相同的菜单或者快捷键可以随时再次打开该窗口。

状态窗口或者输出窗口

本窗口用于输出所有已执行动作的结果，比如系统选择、服务执行等。本窗口可以保持关闭状态，以便给其它窗口留出更多的空间。

其它窗口

以下相关各节介绍了其余几个窗口。

2.4.3 Ecoute 文件

DiagServer 负责管理诊断数据以及系统数据。但是 Ecoute 还会使用其它文件，这些文件如下所示：

名称	后缀	说明
会话	VSC	本文件中存储了 ECU 状态，并且可以在后续启动的时候恢复。其中包括了主窗口与其它窗口（MWG、STG、Errors 等）的大小、位置，选定系统以及与 ECU 的联系状态。
服务组	VSG	可以针对每个系统定义测量窗口中显示的测量分组（XML 格式）。文件包括了窗口位置与大小、图表以及服务选择与参数化（最小/最大、色彩等）。
测量记录	VSR 以及 BIN	可以采用相同名称在两个文件中记录并且保存测量窗口中的测量结果：VSR 文件包括了服务组的信息，BIN 文件包括一系列数字数值。用户不可编辑文件。
手动命令输入	VND	在此保存消息以及通讯参数，以便通过 CAESAR API-I 的通讯。
快照	TXT HTM(L)	仅可通过 Ecoute 写入此类文件。其中包括了数据窗口的内容，可以按照预期方式存储此类内容并且用于文档记录。格式为纯文本或者 HTML。
日志文件	LOG	必要时可以写入各种日志文件。

2.4.4 Ecoute 功能

本节介绍了 Ecoute 客户端的一些最重要的功能。

选择以及关闭系统

联系 ECU

执行服务

读取错误

快速测试

检查测量结果以及设定控制器/调整

变体编码

闪存处理

OBID2

配置 Ecoute 以及服务器选项

宏

Java 程序

程序生成器

标准对象

显示跟踪以及监测数据

手动命令输入

CAN 总线仿真

快照文件存储

ECU 通讯仿真

可通过菜单访问上述功能。还可以采用工具栏中的对应按钮来访问某些常用功能。可选择显示或者隐藏工具栏。

2.4.4.1 选择以及关闭系统

从 System / Select... 菜单或者使用 Alt-A 选择需要诊断的系统。这时会弹出对话框“Select system”并且列出了可用的 ECU 系统。

系统选择列表控件显示了每一项是否基于 VSB 或者 CBF。列表的第二列显示了相关信息（CBF 或者 VSB）。通过两个按钮指定选择列表中显示仅基于一个系统的各种系统，或者显示基于 DIOGENES 参数化的各种系统，或者显示两种类型的系统。Change Directory 按钮可更改诊断服务器查找系统说明以及相应的 DIOGENES 数据所在的文件夹。点击按钮之后，可以从弹出的对话框中选择新的文件夹。诊断服务器随后更新受影响的 DIOGENES 数据。状态窗口中会显示这一动作的进度。接下来会更新系统选择窗口。选择需要诊断的系统，然后点击 OK。状态窗口中显示的消息表示选择了新的系统。选择窗口中采用树形结构显示选中的系统。

比如，变更系统文件夹之后，可以采用菜单项“set standard system path”，可以设定当前系统说明的默认数值以及-.cbf 路径。

还可以通过系统选择对话框手动选择系统，在启动 Ecouste 的情况下，可以作为命令行参数输入需要载入的系统名称。服务器初始化之后会立即载入该系统。可以在 StartProfile.txt 文件中生成需要经常分析的系统项，以便 StartCenter 采用此类系统名称作为参数来启动 Ecouste。

通过“Auto system selection”按钮可以选择包含预定义 ECU 的 ini 文件。此后，应用程序会尝试与选中文件中定义的所有 ECU 建立联系。如果检测到具有有效诊断变体的 ECU，而存在包括此类 ECU 的系统说明，那么会在系统选择列表框中高亮显示该系统说明。

警告：

打开以及关闭系统会影响 DiagServer。如果同时激活了其它客户端，则系统变化还会影响此类客户端。

2.4.4.2 关闭系统

使用菜单选项 System / Close 来关闭系统。这样会关闭显示 ECU 相关数据（系统、错误、测量、跟踪）的所有窗口。

在以下情况下会自动关闭系统：

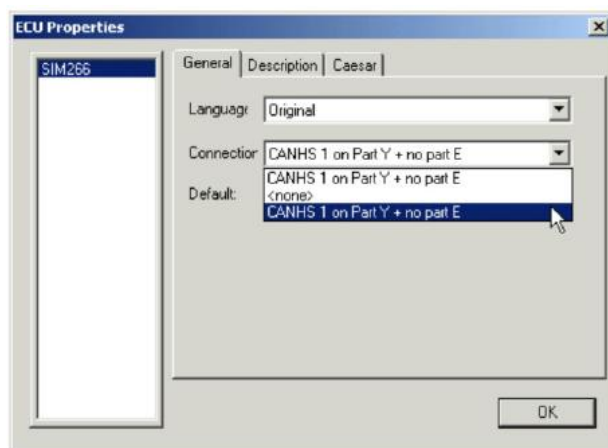
- Ecouste 结束，没有客户端使用服务器。
- 选择了新的系统。
- DiagServer 初始化（服务器复位）。

2.4.4.3 联系 ECU

建立联系之前必须为诊断硬件分配资源（连接）。

ECU 提供了组合的 ECU 选择/属性对话框，这样可以将 CAESAR 硬件的连接分配给 ECU。

ECU 属性窗口



通过 ECU/ Properties 菜单选项或者右键点击选择窗口中的某个 ECU 并且选择上下文菜单中的 Properties，这样即可调用上图中的对话框。

对话框左侧包括了系统当前载入的所有可用 ECU。对话框右侧显示了连接的选择列表。其中显示了当前分配给 ECU 列表中选中 ECU 的连接。上图中，ECU “SIM266” 被分配给 CAESAR 部件 Y 上的 1 号 CANHS 连接。下拉菜单中包括了当前可用的所有联系（并未分配给其它 ECU 的）以及 <none> 项。如果选择 <none>，那么就会释放当前分配的连接，这样可以将该连接分配给其它 ECU。

比如，如果当前选中系统包括两个 ECU，即，ECU1 与 ECU2，其中 ECU1 处于 K-line 4 而 ECU2 处于 K-line 5，如果需要交换这两个连接，则应当采用以下操作步骤：

- 选择 ECU1 以及连接 <none>，以便释放连接 K-line 4。
- 选择 ECU2。之前释放的 K-line 4 会出现在连接列表中。
- 针对 ECU2 选择 K-line 4。这样就会释放之前选择的 K-line 5。
- 选择 ECU1，为其分配连接 K-line 5。

所做的变更会在每个步骤之后立即生效，无需点击 OK 退出对话框。但是变更可能需要数秒的时间方可生效。

使用多工器

CAESAR 硬件会提供超过通讯通道的连接。采用多工器（部件 C 或者 E）可以在不同连接之间切换通道。假设存在以下场景：计算机包括 CAESAR 部件 A。部件 B2 以及部件 E 与该部件相连。硬件配置提供了两种通道。如需将两个 ECU（ECU1 以及 ECU2）与 K-line 1 以及 K-line 2 相连，则操作步骤如下所示：

- 打开连接的下拉菜单列表。比如，可以为两个通道指定 7 个 K-lines。
- 比如，将 ECU1 指定给 K-line 1。
- 那么 K-Line 1 就不会出现在下拉菜单列表中。当前显示的是 6 个 K-lines。
- 比如，可以将 ECU2 指定给 K-line 9。

如果需要将 ECU1 的 K-line 1 切换为 K-line 7，则操作步骤如下所示：

- 打开连接的下拉菜单列表，选择 ECU1。由于已经使用了两个可用的通道，因此列表中不会出现其余的 5 个 K-lines！针对 ECU1 选择连接 <none>。这样就释放了一个通道。
- 打开连接的下拉菜单列表，选择 ECU1。现在会显示 6 个可用的 K-lines。
- 为 ECU1 选择 K-line 7。

如果选择了所有通道，而另外一个人希望采用多工方式将一个通道分配给另外一个针，那么首先必须释放通道（<none> 连接）。然后就可以分配新的针了。

网关成员 ECU

对于网关成员 ECU 来说，会显示“GW member x on ECUy”。其中 x 表示设备编号，ECUy 表示网关 ECU 的名称。

“Description” 标签

ECU Properties 对话框中的该标签中会显示选中 ECU 的基本变体以及当前变体的参数化说明。如果进行了参数化，还会显示部件编号。

“CAESAR” 标签

ECU Properties 对话框中的该标签中会显示选中 ECU 的 CBF 文件版本。

建立与 ECU 的通讯

采用下列命令可以建立或者终止与 ECU 的联系：

- 功能键 F3（建立）以及 F4（终止）
- 菜单项 ECU / Contact...
- 点击状态行中的 ECU 按钮
- 选择窗口：双击 ECU 符号标志或者右键点击并且选择上下文菜单项

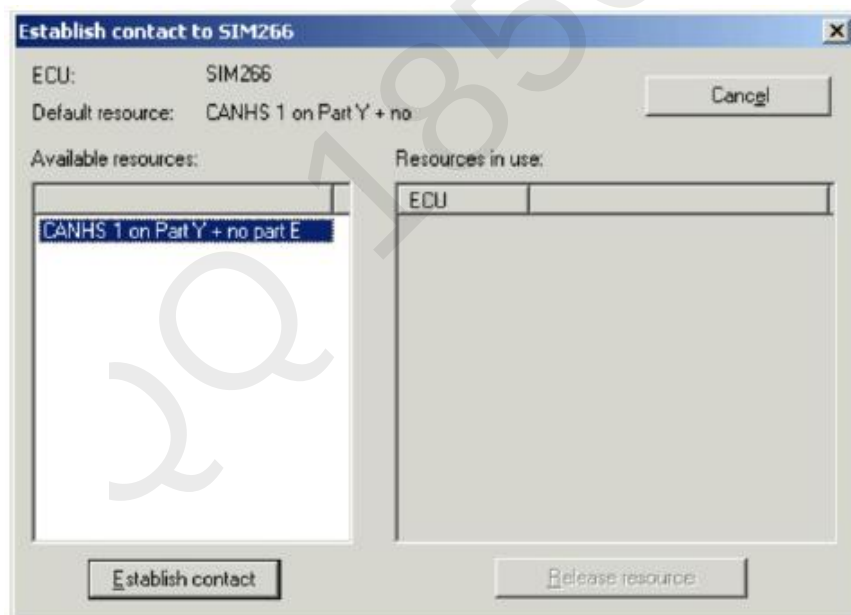
观察选择窗口中的 ECU 符号标志以及状态行中的按钮：建立连接之前的符号标志为：

。一旦建立通讯，则连接器显示为：.

如果识别出了 ECU 变体，那么建立通讯之后，会在选择窗口中的 ECU 名称之后显示变体名称。如果未识别出变体，那么可以自动检查是否针对另外一个 ECU 对变体进行了参数化。

采用手动命令输入方式也可以与 ECU 建立连接。

如果需要与当前未分配资源（通讯通道）的 ECU 建立联系，那么会弹出下列对话框：



左侧列表中显示了可与 ECU 建立联系的当前可用资源。如果 VSB 的默认资源已经包括在列表中，那么就会处于选中状态。如果已经指定了 ECU 所有可用资源，那么该列表为空。

在右侧的列表中，会显示可用的资源，其中包括使用此类资源的 ECU。如果 VSB 的默认资源已经包括在列表中，那么也会处于选中状态。

现在，用户可以（如果没有可用的空闲资源的话）从右侧列表中选择希望释放资源的相关 ECU。如果已经释放了某个资源，那么就会立即更新第一个列表中的可用资源的当前

状态。如需建立联系，用户可以从左侧列表选择一个可用资源，然后点击 Establish contact 按钮。

读取 ID 块

从 ECU 菜单中选择菜单项 Read ID Block，以便读取 ECU ID 块（详情参见键盘操作）。
在一个单独窗口中会显示下列内容：

- MB 编号
- 硬件版本
- 软件版本
- 供应商
- 诊断版本
- ECU ID
- （诊断）状态，比如 “Series production” 等
- 生产日期

此外，相应的系统说明中还会显示其它信息。系统配置中固定的信息如下所示：

对于每个 ECU 变体来说，可以采集为用户提供 ECU 其它信息的所有服务，并且可以在其属性对话框（系统配置、ID block Information 功能）中进行指定。如需显示此类信息，ECU 会向 ECU 请求系统配置中输入的服务 ID，并且执行此类服务。

然后会在 ECU ID 块窗口内的左侧列中显示服务 ID（比如，“Software-Stand”、“Code”、“Boot-Block”）。右侧列则会显示相应的诊断服务结果。

采用 Store 按钮可以将 ID 块窗口的内容存储在快照文件中。

采用 ESC 或者 Close 按钮关闭窗口。

2.4.4.4 执行服务

Eroute 可以从选择窗口中直接执行服务交互。必须在系统树形结构（选择窗口）中选择服务，然后双击或者按回车键即可启动服务。结果会显示在状态窗口（输出窗口）内。

如果选中的服务具有输入参数（DIOGENES 中被称为 “Preparations”），那么在执行之前会弹出对话框提示用户针对每个参数输入相应的数值。

Qualifier	Name
ADJ_Generator_Test	Ergerstrombegrenzung
ADJ_Generator_Test	Spannungsbegrenzung
ADJ_Generator_Test	Ramperzeit
ADJ_Generator_Test	Drehzahlschwelle

Min: -1638.4 Max: 1638.35

Description: Not found.

0.000000

Cancel OK

注意：

执行服务可能永久性改变 ECU 参数。

可以永久性改变的服务包括调整、变体编码、闪存处理等等。

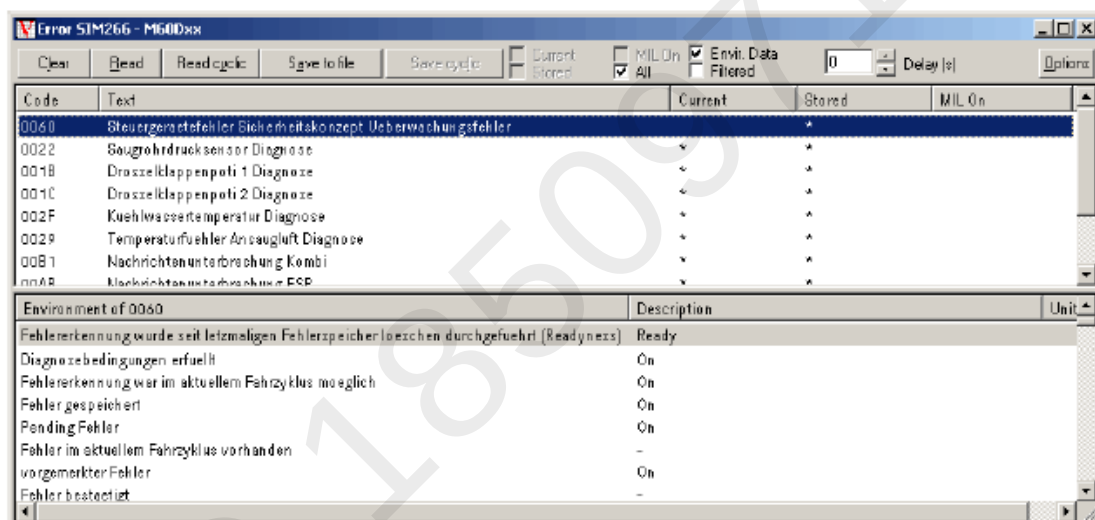
与所有诊断服务相同，在系统窗口中双击也可以执行 Java 程序。使用上下文菜单（右键单击）中的 **Execute with command line params** 项，即可采用命令行中的任何参数执行 Java 程序。可以在系统说明中针对每个 Java 程序设置默认的参数。

2.4.4.5 读取错误

与 ECU 建立通讯之后，即可读取相关错误。可以采用一次性或者循环方式读取 ECU 错误。如果系统文件夹中存在文件名为 **Fehler_Mit_Protokoll.mwg** 的测量组，那么在打开错误窗口的时候会自动打开该测量组，并且与错误窗口相连（**Read**、**Read cyclic**、**Save** 以及 **Close** 动作适用于两个窗口）。

- 从菜单中选择 **Error / Read errors**（**F6** 键）。如果系统具有多个 ECU，这时会弹出对话框，请求选择需要读取错误的 ECU。
- 选中 ECU 之后，会出现错误窗口。窗口的标签为“**Errors**”、ECU 变体名称（或者，如果未识别变体，则为 **<Basic variant>**）。
- 打开窗口之后，从 ECU 读取错误。如果错误数据过多，那么这一过程可能需要一定的时间。然后会显示错误的代码、文本、当前信息、存储的信息以及 **MIL**（故障指示灯）。

调用 **Read errors** 的时候，会打开两部分错误窗口：



错误窗口的上半部分显示了按钮以及选项。按钮以及选项下方是错误与错误代码（“**Code**”）、错误文本（“**Text**”）以及相关标记“**Current**”、“**Stored**”或者“**MIL On**”。如果针对某项设定了其中一种错误标记，则会在相应的错误标记列中显示为星号（“*”）。

错误窗口的下半部分显示了错误环境数据，其中包括：

- 环境名称值（“**Name**”），
- 说明（“**Description**”），
- 以及单位（“**Unit**”）

窗口上半部显示了选定错误的错误环境数据。

日志记载其它信息 – 链接的测试组

您可以建立测量组，在必要的情况下自动显示错误并进行日志记载。对于测试驱动程序特别有用，在此期间按下某个键即可令测试驱动程序确定所有重要信息，无需打开多个窗口并点击相关按钮。

生成常规的原有方式测量组，并且将其存储到文件 **Fehler_Mit_Protokoll.mwg** 中。一旦将该文件提供给选定的系统，就会在错误读取过程中自动读取该文件，并且在存储快照的时候自动存储该文件。测量组不会显示在进程窗口内。

错误窗口按钮与选项如下所示：

“Clear” 按钮：

本按钮用于删除 ECU 错误。必要的情况下（VSB 中要求的项目关机循环），提示用户在关闭循环之后关闭再次打开点火器。

此后会按照需求刷新错误窗口内容（带问号的对话框）。

“Read” 按钮：

可以一次性读取 ECU 错误，并且连同本按钮一并显示在错误窗口中。按下之后，该按钮无法锁定。或者，按“F6”键也可以一次性读取 ECU 错误。

“Read cyclic” 按钮：

Read cyclic 按钮具有两种状态。在<pressed>状态中（显示为按下的按钮），从 ECU 循环读取错误。

打开错误窗口的情况下，按钮处于<not pressed>状态（显示为未按下的按钮）。这意味着不会采用循环方式读取错误。

“Save to file” 按钮：

Save to file 按钮在文件中保存当前错误信息。如果打开了链接的测量窗口，那么会同时保存其中内容。

“Save cyclic” 按钮：

Save cyclic 按钮在文件中循环保存 ECU 错误。仅当循环读取错误的情况下方可激活本按钮。

除按钮之外，可以在错误窗口上部的 Current, Stored and MIL On 选项。

“Current” 选项：

本选项确定是否显示“Current”标志的错误。

“Stored” 选项：

本选项确定是否显示“Stored”标志的错误。

“MIL On” 选项：

本选项确定是否显示“MIL On”标志的错误。

“All” 选项：

本选项确定显示 ECU 的所有错误，与错误标志“Current”、“Stored”、“MIL On”无关。默认情况下会选择“All”选项，这样可以显示所有错误。

“Envir. Data” 选项：

如果激活了 Envir. Data，那么会显示带有环境数据的错误。如果去活了本选项，那么就不会显示环境数据。示例中没有显示窗口的下半部分。如果去活了 Envir. Data 选项，则会从 ECU 中读取不带环境数据的错误，用于通讯协议（比如，KW2000），这样可以读取没有环境数据的错误。

“Filtered” 选项:

可以在 Vediamo 系统配置中过滤掉错误或者错误环境数据，即，过滤掉错误或者错误环境数据不会显示在 Ecouté 应用程序中。Filtered 选项允许用户确定是否应当在 Ecouté 中应用显示错误以及环境数据的过滤器机制。

“Delay” 参数:

本参数用于暂缓循环错误读取。设定时间（秒）意味着读取循环之间存在额外的暂停。因此整个循环等于设定时间加上一次性数据传输所需时间。

删除 ECU 错误

Delete error 菜单项可以用于直接从菜单中删除错误而无需打开错误窗口。如果针对相应的 ECU 变体在系统说明中给出了删除错误的前置条件，则应当删除错误存储器之前执行此类操作。

选择 “Delete error with log” 会在一行内执行三项任务:

- 读取错误（打开错误窗口）
- 在快照文件中存储错误
- 删除错误

“Options” 按钮:

采用通讯协议 UDS 之后，还可以配置用于其它协议的 Read errors 流程。

采用 Read errors 窗口中的 Options 按钮进行配置。按下本按钮即可在显示的对话框中配置 Read errors 功能。

可以再次设定:

- 循环读取过程中的延迟时间
- 错误读取过程中使用的标志

此外:

- 如果以组为单位读取或者删除错误，那么可以在此指定组。

最上端显示了当前使用的协议。

点击 Advanced 按钮就会弹出对话框，其中可以选择错误读取专用的单个协议特定参数。

为了方便进行配置，将五个输入框合并为一个“组合列表框”（combobox）。

采用 CAESAR 中预定义常量（比如 “Group” 字段）载入各种列表:

UDS 中:

示例:

EEG_UDS_EMISSION
EEG_UDS_POWERTRAIN
EEG_UDS_CHASSIS
等

KW2000 中:

示例:

EEG_KW2000_POWERTRAIN
EEG_KW2000_CHASSIS
EEG_KW2000_BODY
等

用户可以选择列表中输入的配置数值, 或者手动输入数值。点击“Apply”按钮, 则此类数值被用作当前错误会话的默认数值。点击“Save”按钮, 则会将当前设置作为标准设置永久保存到文件 Vediamo.ini 中。点击“Standard”, 则会选择内部存储的默认设置, 可以采用下列方式应用或者保存。

“Show ROE” 选项:

本选项按钮位于错误窗口右侧。

ROE (事件响应) 指示灯的需求是, 一旦错误状态发生变更, 那么 ECU 就会将此类信息以及 DTC 编号放入定义的 CAN ID 之中。

如果激活本选项, 并且 ECU 与相关诊断数据均支持 ROE 指示灯, 那么“ROE events”弹出窗口内会显示关于 DTC 状态变更情况的消息。首次发生 DTC 事件则会打开窗口, 前提是该窗口之前已经关闭了。其它事件会显示在同一窗口的下行内。一旦发生新的事件, 则会重新打开窗口。

2.4.4.6 读取永久性错误

从 4.1 版开始不再支持本功能。本功能包括在了 OBD2 功能之中。

2.4.4.7 状态读取的错误

KW2000 特定函数 REQUEST SUPPORTED 2BYTE HEX DTC AND STATUS (\$03) 执行, 成为与 Vediamo 的连接的“\$18 \$03”, 或者“Error reading by Status”。

仅 KW2000 协议 (类似于 UDS 协议) 支持本功能。尽管应用程序可以通过 CAESAR 获取当前使用的协议的名称, 但是无法根据这一名称 (比如, KW2C2PE、KW2C3PE 等) 确定协议是否源自 KW2000 或者 UDS。因此, 用户应当合理地使用本功能。

激活菜单项之后会显示专用窗口。

针对每个错误在窗口中显示具有错误编号的按钮。项目颜色主要取决于错误状态。窗口显示的图例解释了颜色编码。选中某一项, 则会在窗口的下部显示对应的错误文本。如果窗口中无法同时显示所有窗口项, 则可以采取滚动窗口的方式查看相关项。

分别点击 Read 或者 Read cyclic 按钮, 则状态会采取一次性或者循环方式读取所有支持的错误代码, 并且同时会更新窗口。

点击 Store in file 按钮, 则会弹出对话框, 其中应当输入需要保存的文件名。日志格式与同名 Java 程序生成的之前格式相同。

CAESAR 或者 DIOGENES 参数化目前无法采用令人满意的方式支持采用 KW2000 功能 \$18 \$03 实际读取错误。

因此通过调用 CAESAR-API1 的方式在 Vediamo 服务器中针对 KW2000 实现本功能。为此，Vediamo 对于使用的 ECU 设置了某些最低需求：

- 获取错误代码总数，其中必须支持\$18 \$E0。
- 在逐块依次读取具有 \$18 \$03 的错误代码过程中，每个读取块均应当包括块中所含错误代码数量。

“Permanent DTCs”按钮开始读取永久性 DTC。永久性 DTC 的列表中的所有代码均在代码旁边显示为 “P”。

为了在测量以及后续评估中更好地获得总体概况，可以指定 ECU 执行的其它服务（测量）。

与 ID 块扩展类似，可以在系统配置中定义测量。

读取错误代码之前执行服务，然后显示在 “read errors by status window” 以及协议文件中。

2.4.4.8 系统快速测试

Quick test 功能读取及/或删除所有当前可寻址 ECU 的错误（比如，对于整车）。

如果在 Ecout 启动的时候输入命令行参数 -K，在启动程序之后自动调用该功能。对于快速测试而言，必须在“quick test directory”中提供必要的 DIOGENES 数据（.cbf）：可以在以下项中的 Vediamo.ini 文件中指定包含 DIOGENES 数据的快速测试文件夹：

[ECOUTE]

ShortTestDataDir=

标准信息文件用于指定需要诊断的型号以及 ECU。本文件包括了可以提供给用户的所有型号的相关信息。各种响应的发动机 ECU 信息文件中包括了各种型号。文件的默认名称为 KurztestMercedes.ini，内容的构建示例方式如下所示：

示例：

[Models]

NumberOfModels=3

Model1=BR211

Model2=BR220

Model3=BR203

[BR211]

NumberOfECUs=5

;these are the (engine) ECUs from which the user

;must select one for the quick test

OptionalECU=1,3

;ECUx = <ECU name>, <Connection>,

; <Pin with Part E/Pinmapping>,

; <Pin without PartE/Pinmapping>

ECU1=ME28, CANHS, 1, 1

ECU2=ZGW211, CANHS, 1, 1

ECU3=CR3, CANHS, 1, 1

ECU4=ARMADA, KLINE, 6, 1

ECU5=SAMH, KLINE, 11, 1

[BR220]

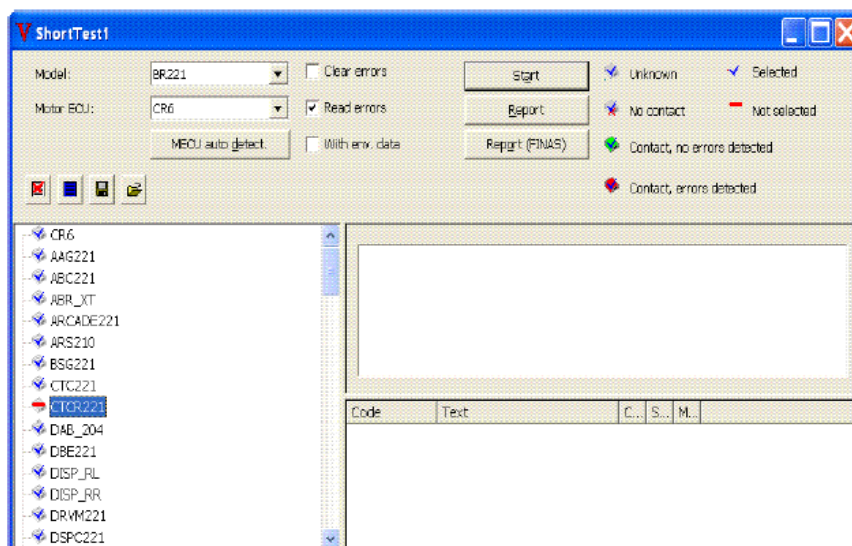
...

[BR203]

...

快速测试开始的时候，用户可以选择需要使用的信息文件（*.ini）。

选中文件之后，会显示下列窗口：



用户可以从中选择型号，然后选择对应于该型号的一种发动机 ECU。

点击 **MECU auto detect** 按钮，则会弹出对话框以便自动获取针对该型号注册的所有发动机 ECU。如果检测到一种 ECU（具有不等于基础变体的诊断变体），那么可以选择并且进一步使用本发动机 ECU。

选择型号/发动机 ECU 之后，窗口左侧会显示对应于该型号配置的各种 ECU。

还可以在窗口中进行下列设置：

- 读取错误
- 采用环境数据
- 清空错误

点击 **Start** 按钮开始测试。

对于需要在快速测试中考虑的 ECU 各种规范来说，还存在下列其它选项：单个选择一个或者多个 ECU。针对选中的 ECU，忽略掉未选中的 ECU。

点击鼠标即可选择/取消选择所有 ECU，还可以保存或者载入最近一次的选择配置。选择某个型号的 ECU 的具体方法是，在结果窗口内选择每个 ECU 的“单选框”（直接点击树形结构中的 ECU 符号标志也可以选中），而且四个按钮的名称分别为“Deselect all ECUs”、“Select all ECUs”、“Save filter configuration”、“Open filter configurations”。

“Deselect all ECUs”按钮的功能：

点击本按钮会取消针对快速测试选中的所有 ECU。

“Select all ECUs”按钮的功能：

点击本按钮会针对快速测试选择所有 ECU。

“Save filter configuration”按钮的功能：

允许保存 ECU 的选择状态。点击本按钮，则会弹出文件选择对话框，指定需要采用文本格式保存选择状态信息的文件名称。文件默认扩展名为.Vkk。采用下列格式存储此类信息（示例）：

[General]

Version=1.0

Model=BR204

MotorECU=CR6

SelectedECUs=5

ECU1=CR6

ECU2=CGW_204
ECU3=CLAMP15
ECU4=CTRLC_204
ECU5=DAB_204

“Open filter configurations” 按钮的功能:

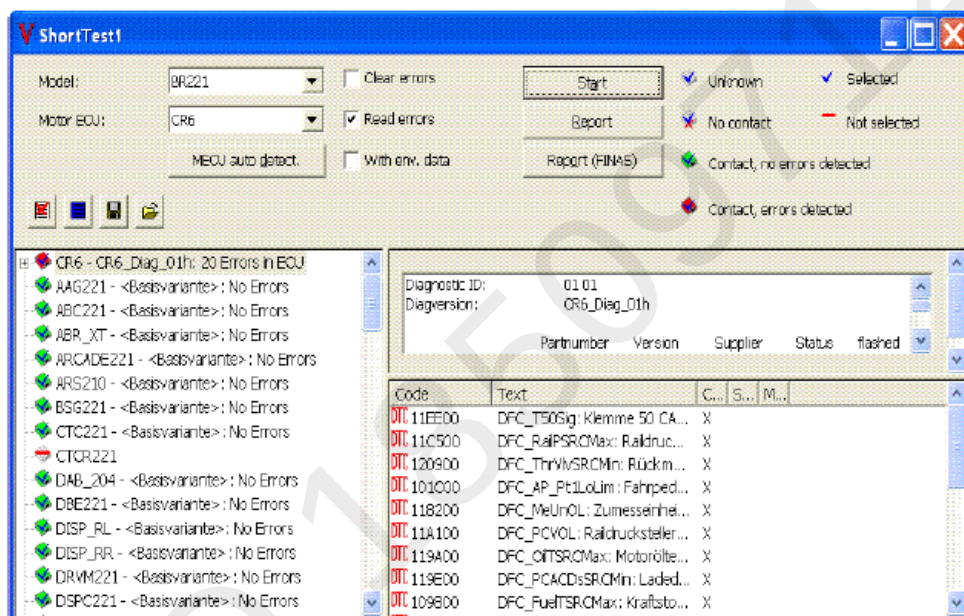
点击本按钮会提示您选择包括快速测试配置的文件 (.vkk)。适当情况下, 采用包括的过滤器信息选择/取消选择当前选中系列的所有 ECU。

运行检查以便确定是否具有可用的 DIOGENES 参数化。如果出现不一致的情况, 那么就会放弃快速测试并且提供错误消息。

否则启动实际测试流程。

尝试确立与型号配置中包括的每个 ECU 建立联系, 在此期间采用给出的 CAESAR 资源。如果建立了联系, 根据相应的设置, 读取及/或清空错误。快速测试窗口内进度指示符显示了测试进度, 状态窗口内显示了其它信息。

完成快速检测之后, 会在快速检测窗口内显示相关结果:



左侧结果树包括了快速测试中涉及的所有 ECU 的第一级别。如果选择了型号以及发动机 ECU, 那么填充该级别。采用相关图标显示当前状态。功能四种状态:

- 未知 (灰色符号标志): 尚未激活 ECU, 比如, 由于没有可用 CAESAR 资源导致的。
- 无联系 (带有红叉的灰色符号标志): 尝试与 ECU 建立联系失败, 或者未识别任何变体, 或者错误读取失败。
- 无 DTC (绿色符号标志): DTC/事件数量 = 0
- 采用 DTC (红色符号标志): DTC/事件数量 > 0

ECU 的最初状态为<unknown>。

如果 ECU 具有 DTC/事件数量 > 0 的情况, 那么会显示类具有子列表的结果树中。如果存在 DTC, 则会在树中显示其数量。

测试过程中, 状态从未知变更为其它任何一种状态。但是, 这种情况下不会自动展开子列表。

右键点击某个 ECU 项会打开一个窗口, 其中显示了相应的 ECU 的其它信息。

结果树后半部分：

本部分显示了结果树中选中错误的环境数据。如果并未选中任何错误，或者选中了 ECU，那么会在结果树后侧显示相应的 ECU 信息，并且下方显示错误窗口。这种情况下不会显示环境数据。

如果在结果树中选中了某个错误，那么环境数据上方右侧部分会显示错误数量、错误文本以及状态（MIL、current、stored）。

点击 Report 按钮可以将快速测试结果日志记载到文件中。

如果 Vediamo 快速测试数据中存在名称为 ECU-Name-Mapping.txt 的文件，那么会采用纯文本名称补充快速测试中的 CBF 名称（在状态窗口中，具有快速测试结果的窗口，以及快速测试日志文件）。

此类列表内容示例如下所示：

示例：

[ECU-Name-Mapping]

EWM211=Electronic transmission selector lever module for BR211

TSG_V_L211=Front left door control unit for BR211

Presentation in quick test:

EWM211 - Electronic transmission selector lever module for BR211

TSG_V_L211 - Front left door control unit for BR21

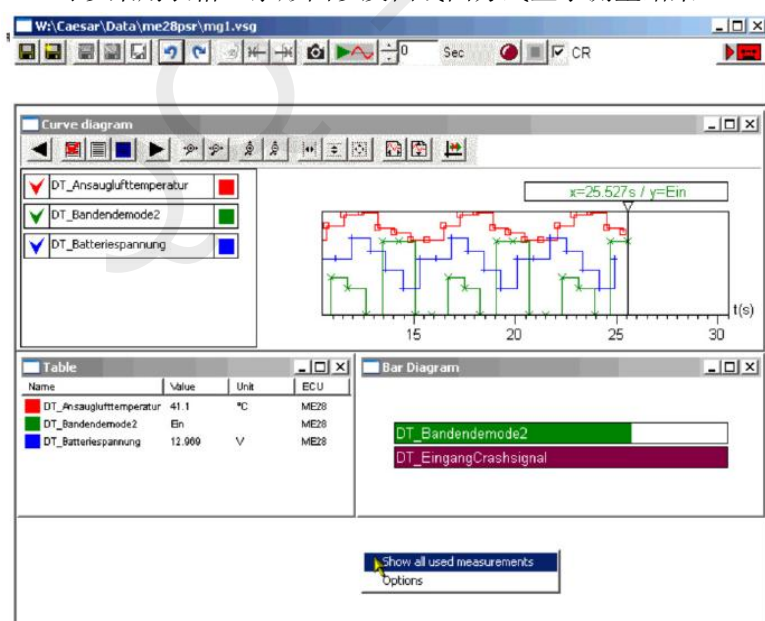
2.4.4.9 服务组

双击系统窗口即可执行测量以及其它服务。但是对于更高级的试验来说，可以采用选中的服务来定义一个服务组，以便采用多种图表方式显示和控制此类服务：表（文字）、测量数值的条形图与曲线，以及其它服务的驱动器窗口。定义窗口属性与内容的服务组保存在 VSG 文件中。

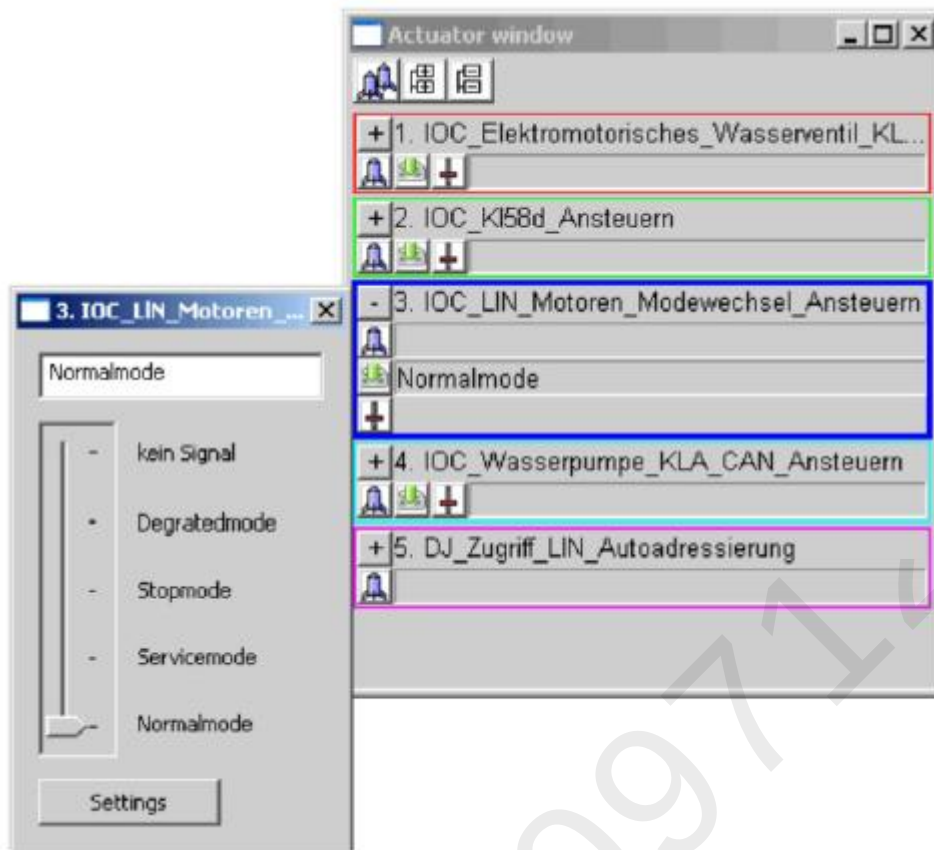
如需创建新的服务窗口，请选择主菜单中的 Service Groups / New Group。如果针对当前系统提供了 VSG 文件，那么可以选择 Service Groups / Open Group 载入此类文件。

一旦将服务插入窗口中，那么所有服务均会具备更强目标指向性的颜色。所有图表中均采用该颜色来代表此类服务。

可以采用表格、条形图以及曲线图方式显示测量结果：



为了执行除测量数值之外的其它类型服务，可以使用驱动器窗口：



上图中，每一种服务均获得了控制此类服务所必需的各种控件。采用颜色边框包围起来的是一种服务的各种控件。适当的情况下，使用此类控件改变输入参数，执行服务并且读取 OutputRef 测量结果（测量服务传送驱动器的实际状态）。

各种图表中不同的服务处理方式

系统窗口插入测量图表（表、条形图或者曲线）中的服务被视为测量服务，即，会在每个单一快照或者读取循环中执行测量，而且结果会显示在图表中。

将文字格式返回值映射到连续数字之中，以便将其显示为一条曲线。比如，将“On”映射为 1，“Off”映射为 0。如果无法通过自动映射满足您的需求，那么还可以在服务选项中变更映射表。

插入到驱动器窗口中的服务会被自动采用其它方式予以处理。此类服务被视为改变系统状态的要素。仅可通过驱动器窗口中的“Execute”按钮来执行此类服务。

不过，您可以从驱动器窗口中将某个服务拖拽并放置在测量窗口中（表、曲线、条形图）。这种情况下，测量窗口会出现若干新项：

- 一个或者多个输入参数，标记为“<P>”
- 复制自行执行，标记为“<S>”
- 存在的话，还包括驱动器的 OutputRef 服务

如果通过驱动器窗口控件执行了服务（这种情况下是驱动器），那么测量窗口会显示下列内容：

- 每个输入参数一个测量数值
- 执行结果
- OutputRef 服务结果，在服务自身之后直接执行。

如果无需图表中的此类信息，那么可以将其单独删除。

注意：反之，不可能将服务插入测量窗口然后再插入驱动器窗口。

测量重口的控件

测量窗口的控件位于工具栏中。某些控件可能不可见或者处于去活状态，主要取决于窗口的当前状态。



控件的功能（从左至右）：

- 保存服务组文件（VSG）。文件中保存了图表、服务、位置等所有变更情况。
- 将服务组另存为新的文件。
- 保存测量系列。如果记录了测量结果，那么可以采用成对文件（VSR 与 BIN）方式保存此类内容。如果没有可用的记录数值，则该按钮处于去活状态。
- 将记录的数值输出到 CSV 文件中（逗号分隔数值），此后可以将其导入到 Excel 或者数据库中。如果没有可用数值，则该按钮处于去活状态。
- 将所有测量的最新数值保存到快照文件中。
- 撤销变更（最多可达 10 步）。这样可以恢复重要变更之前的最近状态，比如添加/删除图标或者服务，删除记录数据等。改变窗口大小或者位置等简单操作并未被视为单独的变更，因此无法撤销。
- 重复变更。撤销的变更可以再次重复。
- 从存储器以及图表中删除记录的数值。如果正在进行记录，那么在用户确认后可以停止此类记录动作。
- 删除最多可达时间滑块当前位置的记录数值。
- 删除从时间滑块当前位置开始的记录数值。
- 单一快照。一次性读取所有测量结果。
- 循环读取开始/停止。激活循环读取的情况下，会循环执行所有服务。暂停循环之后，当前图表的时间轴持续移动，并且可以去活循环或者可以进行单一快照。
- 设定循环读取的延迟（秒）。可以尽快执行循环（延迟 0）或者每隔 X 秒执行一次，详情参见控件中的输入结果。正在读取的过程中无法变更延迟。
- 记录。从这一刻开始，会在存储器内记录所有数值，并且此后可以保存到文件中。如果选中了 **Recording starts cyclic reading** 按钮，那么会自动开始循环读取。
- 停止。本按钮停止记录。如果激活了循环读取，那么也会同时停止此类服务。窗口模式变更为分析模式：停止读取测量结果，可以分析记录的数值，变更显示选项并且将测量数值保存到 VSR/BIN 文件对中。
- 单选按钮 CR=Cyclic Recording (Recording starts cyclic reading)。如果希望在开始记录的时候自动开始循环读取，那么请选中本单选框。否则，按下记录按钮之后，必须点击 **Cyclic reading** 或者 **Single Shot** 按钮。
- 窗口状态。如果模式为 **Recording** 或者 **Analysis**，那么可以看到一个录音带以及眼睛的符号标志。

（第 2 行，从左至右）

- 时间滑块的当前位置。仅当存储器内存在测量数值的时候才可见。如果输入了浮点数值，那么显示内容会将测定点移动至最近的指定时间（从 0 秒开始）。
- 时间滑块。您可以使用该滑块方便地选择显示点。还可以使用箭头来改变时间位置。

键盘的使用

可以通过键盘在测量期间使用多数重要功能：

- 空格键- 开始/停止循环读取
- 回车键 – 开始/停止记录。如果并未启用自动保存（详情参见下文配置），那么停止记录之后会提示您输入保存测量系列所需的文件名。如果第三次按下回车键，那么会采用默认名称（窗口名称+日期+时间）保存这一系列。
- Delete – 清空存储器。从存储器中擦除所有记录数值。如果目前尚未保存记录数据，那么会询问您是否应当马上保存。点击 ESC 即删除而不会保存，或者点击回车键予以确认，再次点击回车键则接受默认的文件名称。
- Shift-Del 以及 Ctrl-Del – 删除时间滑块当前位置左侧或者右侧的记录数值。

测量窗口的模式

窗口共有 4 种模式/状态：

- 配置模式（自由运行模式）

您可以在本模式中随意配置窗口：其中包括添加以及删除测量专用图表。您还可以采用当前配置读取并且显示测量结果，但是不可将其保存在文件中（VSR/BIN 或者 CSV）。

- 记录模式（右上角的红色录音带符号标志）

在本模式中，所有读取数值均临时性记录在存储器中。不允许对数值进行不一致的变化（即，不可添加新的测量数值，但是可以添加窗口中已经使用的测量数值）。如果从图表中删除了某个测量数值，那么该数值即使不会出现在任何图表中，但是仍然应当包括在内部测量列表中。为了查看所有用过的测量数值，从弹出菜单中选择 **Show all used measurements**（右键点击窗口空白区域）。

如需停止记录模式，请点击 **Stop** 按钮或者 **Delete** 按钮。

- 分析模式（右上角的眼睛符号标志）

记录测量数值之后即进入本模式。不可变更测量数值列表（即，添加新的测量数值），也不可从 ECU 中读取数值。只要本模式处于激活状态，记录的数值就实中保持一致性。可以改变影响数值的任何内容：最小值以及最大值、颜色、图表及其位置等。记录的数值可以保存到 VSR/BIN 文件中，此类文件还包括了图表、参数、颜色等，或者也可以保存到 CSV 文件中，此类文件可以用于其它工具，比如 Excel 或者数据库。如需离开本模式，则必须显式删除记录的数值或者启动新的记录。在这两种情况，如果尚未保存记录的数值，则系统均会提示您进行确认。

- 仅分析（右上角的眼睛符号标志）

本模式与上一模式的差异在于，在仅分析（Analysing only）模式中仅可在如之前保存的测量记录（VSR/BIN 文件对）。由于您获取数值的文件并未真正与 ECU 相连，因此不可能切换至 configuration 或者 recording 模式。您可以变更显示选项（颜色、线粗细等）并且在新的 VSR/BIN 文件对中保存一系列测量数值，输出到 CSV 文件中，或者提取服务分组定义文件（VSG）。

配置服务分组窗口

开始试验或者测量会话之前，必须配置适当的服务分组。需要注意的是，需要读取的测量数值列表以及确定读取服务频率的延迟因子。

您可以向一个测量窗口内插入多个图表，采用相同或者不同的服务对其进行填充。窗口中使用的所有服务均可在一个循环内执行，其结果会发送给所有图表以便对其进行显示。

如需采用服务填充图表，仅需将服务从系统窗口或者包含测量数值的另外一个窗口中拖放出来。或者还可以选中服务，然后对其进行复制/粘贴。

注意：

如果并未与 ECU 建立联系，那么在系统窗口内可以显示所有变体及其服务修饰符。但是，

不可以将当前 ECU 变体中未定义的任何服务插入到服务窗口内。

如需插入来自任何 ECU 变体的服务，请在仿真模式中启动 VEDIAMO，并且选择希望插入服务所在的变体。

注意：

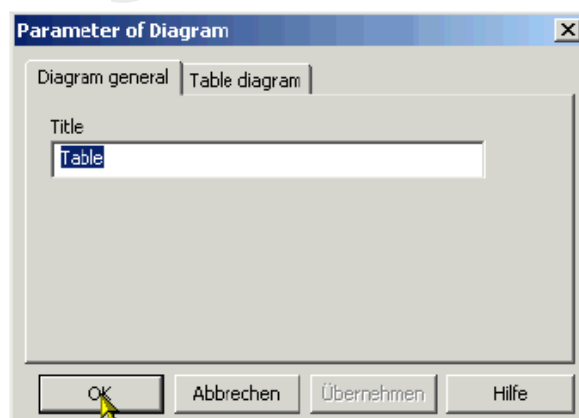
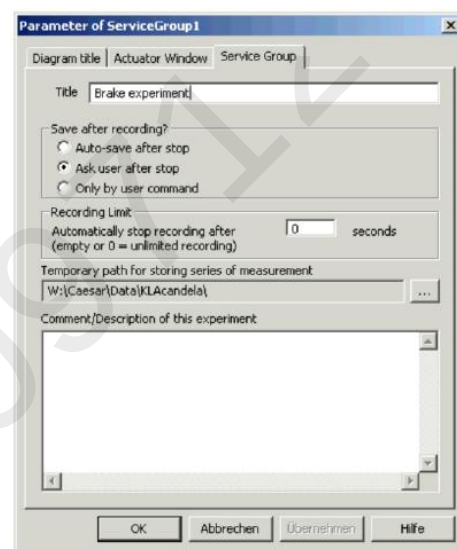
服务窗口可能包括多个不同 ECU 的测量数值。如果使用本窗口的系统并未包含所有 ECU，那么可能无法执行某个测量，因此无法显示此类测量数值。

图表以及添加的服务均是预先配置的，一般情况下无需配置或者仅需少量配置。如需变更设置，仅需右键点击服务或者图表，即可在弹出的服务或者相应的图表配置对话框中进行配置。

服务窗口的参数

点击窗口空白区域，即可弹出该窗口的配置对话框。在此处可以变更相关设置，此类设置会影响测量窗口的功能：

- 分组的标题。
该文本显示在窗口的标题栏中。
- 停止记录后的行为
auto-save 是指，按下停止按钮之后，所有记录的数值会记录保存到特定文件夹的 VSR/BIN 文件对中，并且采用默认的文件名。
Ask user 是指，停止之后会提示您输入文件名称。还可以接受默认名称（点击回车键），或者输入另外一个名称。
Only by user command 是指，停止记录之后，不会提示您保存，而是仅当您启动所有删除记录数值的动作时进行提示（即，删除、停止记录、关闭窗口）。
- 记录限值。如果大于 0，则该时间之后会自动停止记录。
- 记录数值的存储路径（VSR/BIN 以及 CSV 文件）。如果进行了多次试验（即，测试过程中），那么可以针对每一天、每个受试对象或者车辆建立一个不同的文件夹。如果不输入，则文件会存储在当前打开系统的文件夹中。
- 备注。您可以在此处输入文本格式的所有信息。这种情况下会采用服务分组文件 VSG 以及一系列测量文件（VSR/BIN 以及 CSV）进行保存。
- 其它变化，比如添加或者移除图表等，均可直接从弹出菜单中选择指令，或者关闭图表窗口。



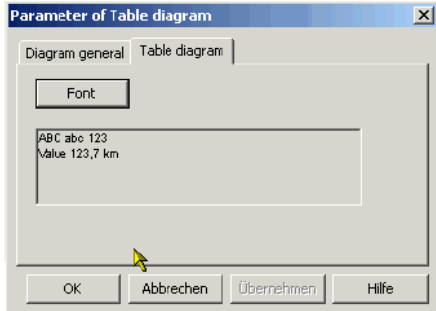
图表的参数

右键点击图表区域，而不要点击服务修饰符，即可弹出对话框，其中包括了图表参数。对话框有两个属性页：一个具有通用参数，一个具有这一类型图表的特定参数。

所有此类参数均存储在 VSG 文件中，并且可以在下次打开文件的时候进行恢复。

通用参数

所有图表的标题均可以编辑。默认标题说明了图表的类型。

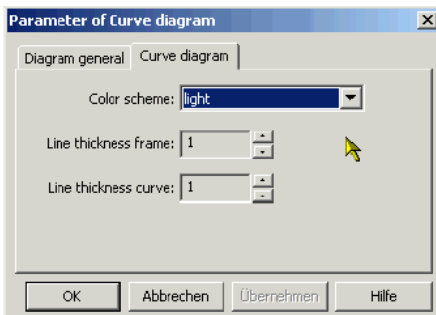


表参数

在表中可以更改字体(字体名称、大小、字号等)。

曲线参数

可以在此类图表中设置线的粗细以及颜色方案。需要注意的是，如果采用深色方案，那么线可能会更浅一些，而在浅色方案中，线会显示的更深一些，这样做是为了强化对比度。



条形图表参数

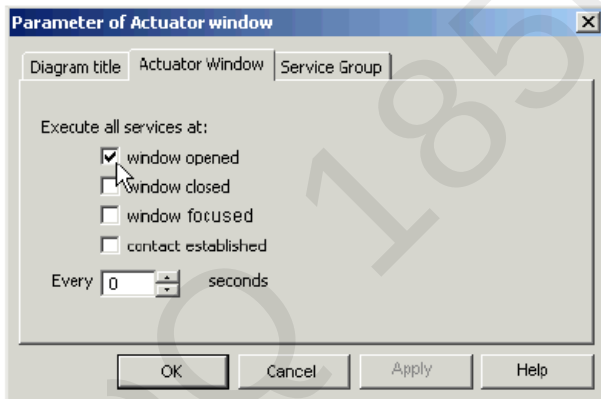
条形图表不存在特定参数。

其它参数

拖拽光标可以更改图表窗口、表栏等的位置以及大小。

驱动器窗口的参数

可以在此处定义事件，此处可以自动执行所有服务。一般情况下会在“window opened”的情况下执行前置条件，或者针对某些 ECU 重复执行设置驱动器“Every x seconds”，其中会在超时之后自动重置所有驱动器。



服务参数

右键点击图表中某个测量服务的修饰符（任何图表），可以在弹出的对话框中设置测量参数。

首先，需要为服务指定替代性名称（别名），这个名称会显示在当前测量窗口的所有图表中，替代实际的修饰符或者名称。

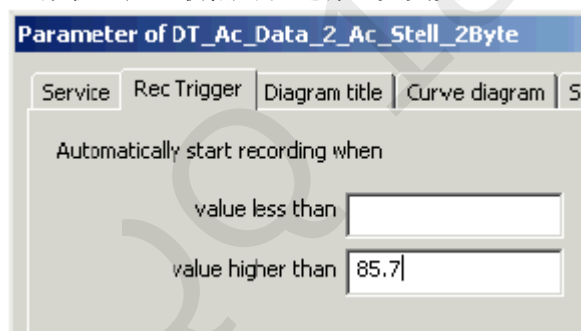
最小值以及最大值决定了图表的缩放比例（条形图以及曲线图）。下次从 ECU 中读取某个数值的时候，会选中该数值。如果超出限值，那么会将限值自动调整为新的数值。此类数值适用于测量包含的所有图表。

测量所属的颜色也同样适用于所有图表：如果采用绿色曲线方式显示数值，则条形图也会显示为绿色条，在表中的名称旁边会显示相同绿色的图标。

标记是一种曲线图中标记测量点的符号标志。

数指标用于将数值附加到文本上，这种情况下服务传送的是文本响应而非数字。通常使用的文本包括“on”、“off”、“yes”、“no”，其它则自动采用 0 以及 1 来表示。从 ECU 中接收的其它所有文本均获得下一个闲置的数字。

如果您改变了自动定义的文本-数字对，那么应当点击 Value table 按钮。这是会显示当前测量服务所使用的所有数值。您可以在此时更改附加于文本上的数值。更改数值之后，采用该文本的所有测量点均会变更为新的数值。这样，在记录甚至在载入已经记录的一系列测量数值之后，仍然可以进行此类变更。



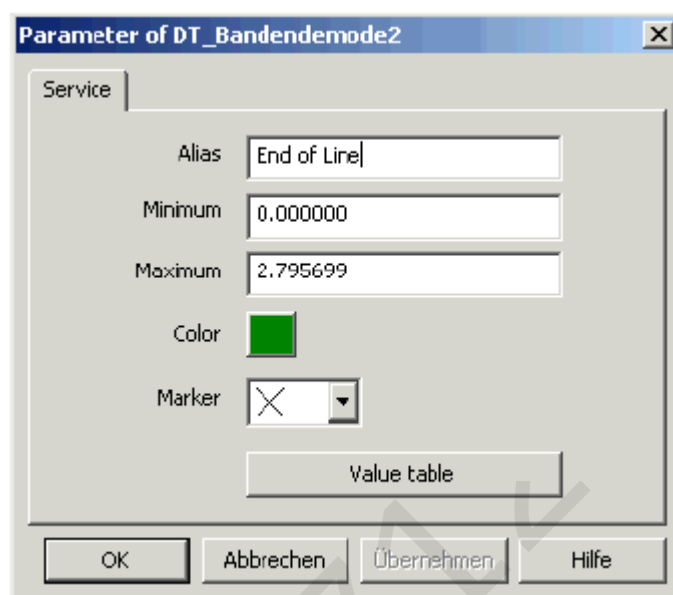
服务窗口选项）。

读取、记录以及分析测量数值

无论拥有多少图表以及拥有何种图表，测量窗口均会管理至少一个图表使用的测量服务列表。读取的时候（单次或者循环），每个服务仅会执行一次，而且会将结果发送给所有受影响的图表。这样还确保了所有进入的数值均处于范围之内。如果任何数值超过最大值或者小于最小值，则会自动调整限值。

在 Configuration（Free Running）以及 Recording 状态中，可以执行所有服务（从一个或者多个 ECU 请求数值）。您可以选择单次执行或者循环执行，同时还可以选择采取或者不采取延迟措施：

- 单次



服务选项第 2 页用于设定触发事件。

设定完毕之后，一旦低于该下限值，或者超过该上限值，则会自动开始记录。

凭借本特性，程序可以等待达到某个数值之后再读取当前状态，然后自动开始记录。

您可以将这一特性与一定时间之后自动停止记录特性组合在一起（详情参见服务窗口选项）。

一次性执行窗口内的所有服务。

- 循环读取而且无延迟

尽快执行所有所有服务。如果输入数值的次序可能发生变化，这种情况主要取决于多个问题。尤其是，如果您拥有不同 ECU 的服务，那么 ECU 会独立执行此类服务。这意味着，您从 ECU1 接收的测量数值数量可能超过从 ECU2 接收的数值数量。

- 循环读取而且有延迟

如果您在更长时间内记录数值，那么可以采用特定频率读取数值。您可以以秒为单位设定延迟。如果整个循环的时间超过规定的延迟，那么会调整重复时间。上一个循环完成之前不会开始新的循环。

存储器的使用：

记录的测量数值会保存到系统存储器中，以便快速访问以及编辑。每个数值均需要占用一定的字节（具体取决于字符串结果的长度）。机器变慢之前运行一次循环的时间（在操作系统开始将交换存储器中的内容向 swap 文件中移动的时候可能会发生此类情况），那么主要取决于空闲 RAM 的数量以及测量读取的速度。

在开始重要测量系列之前，采用任务管理器测试存储器数量以及存储器填充速度。

如果系统资源不足，那么可以增加存储器的储备（停止正在运行的应用程序以及进程），减少存储器的使用（采用更长的延迟进行记录），或者定期保存记录的数据并且开始新的记录。

记录测量数值

在记录过程中，程序会确保整个系列的一致性。这意味着，在本模式中，不可能改变已使用的服务，无法添加测量数值或者删除任何正在使用的测量数值。

但是您可以更改以下内容：

- 添加、关闭、移动或者调整图表大小
- 更改颜色、比例以及数值表项
- 添加、移除以及移动窗口中已经存在的服务

这种情况下可以采用某种特性来优化记录速度：

性能：

绘制图表可以极大地增加 CPU 的负担。需要绘制的内容越多，可以执行的测量越少。如果您希望尽快读取并且记录数值，那么在记录过程中仅可使用一张表格类图表。如需分析数据，您可以在记录之后打开其它图表，记录的所有数值均会显示在其中。

保存以及编辑测量系列

可以将记录的数据保存到一对文件中（详情参见上文）。VSR 文件定义了窗口内容以及布局，其中包括测量的所有参数以及图表（详情参见上文）。BIN 文件包括二进制数值。仅可与 VSR 文件配合使用。

这两种文件必须配合使用，不得在 Ecoute 之外编辑，否则可能丧失一致性。

文件名

保存测量系列的时候，Ecoute 会采用包含的测量窗口名称、当前日期以及时间（开始记录的时间）来创建文件名。如果同时保存多个系列，那么可以在文件名之后加上下划线以及数字。这样可以确保不会意外覆盖记录。当然，用户也可以在文件选择对话框中更改默认名称。

保存测量系列所需的路径

除非另行声明，否则文件均保存在当前系统的默认文件夹中。可以在文件选项对话框（提示保存的时候）中更改文件夹，或者在窗口参数设置对话框中设定路径。新设定的路径应用于当前测量窗口此后所有保存动作，此类动作涉及 VSR/BIN 文件或者 CSV 文本输出文件。

自动保存

在测试过程中，可能需要激活自动保存功能（窗口参数设置对话框中）。在本模式中，用户仅可打开适当的窗口并且在开始之前建立与 ECU 的联系，然后点击回车键开始以及停止记录。记录停止之后，采用默认文件名自动保存记录。

编辑测量系列

记录之后，变更主要涉及下列数据：

- 剪切系列中的部分内容。采用时间滑块或者曲线图中的时间选择线选择剪切点。
Shift+Del 删除选中点之前的所有点，Ctrl+Del 删除选中点之后的所有点。
- 更改显示选项。可以添加、删除、移动或者调整所有图表的大小，将测量数值放入其中或者从中移除测量数值，更改颜色，缩放以及标记测量数值。
- 还可以更改测量参数。即，最小值以及最大值。如果测量服务采用文本而非数字数值进行响应，那么程序会将数字数值附加到每个文本上（数值表），以便采用图形方式显示数值。可以更改附加到文本上的数值，程序会针对整个系列重新记录记录的数值。
记录后或者甚至在载入之后可以针对已经保存的系列进行此类更改。

图表处理

表格类图表

在表格视图中，会显示当前时刻（或者任何指定时刻）的测量数值。在读取/记录过程中，表中包含了最近读取数值。

如果移动时间滑块或者在时间编辑区内输入时间数值，那么会显示指定时刻之前读取的最新数值。

名称旁边（修饰符或者别名）显示了分配给该项服务的颜色。此类颜色用于在窗口的任何图表中显示此类测量数值。

按照原有方式在表中显示测量数值：文本保持原样。

为了显示得更为明了，可以对测量数值进行分组并且采用逗号分隔。

可以更改各栏的字体以及大小。如果名称过长或者含义不清楚，除了调整名称栏宽度之外，还可以为服务指定别名。此类别名会显示在所有图表中。

将鼠标指针悬停在名称或者别名上方，则弹出的工具提示中会显示全名、修饰符以及 ECU 修饰符。

条形图

条形图采用水平彩色条显示测量数值的变化情况。方便用户快速查看数值变化情况。整个图表窗口的宽度用于显示测量上限以及下限之间的所有数值。

图表中进现实当前（上次读取）或者选定（采用时间滑块）的数值。

点击测量名称即可选中。条的上方显示了选中测量数值范围。

此类图表中不存在其它设置。

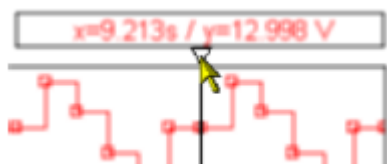
曲线图

曲线图显示了数值的按照时间顺序的发展变化情况。



左侧区域包含图表中显示的测量数值列表。与其它图表相同，可以控制测量数值的设置（右键点击鼠标->弹出菜单->选项->参数对话框）。左键点击鼠标选中某个测量数值。影响测量的窗口控件的动作总是应用于选中的测量。如果未选中，则应用于所有测量。

点击名称旁边的选中标记，可以切换显示此类特定曲线。关闭测量的情况下，无论控件处于何种选择状态，均会忽略测量控件的动作。



右侧区域包括曲线以及焦点线。拖拽三角形即可移动焦点线。如果按住鼠标左键拖拽线，那么会移动曲线的数值。松开之后，会跳至最近的数值点。其它所有图表均变更至选定显示点。

如果按住鼠标右键托转，那么线会随着曲线移动。不会更改选中的显示点。

点击  按钮可以将曲线图中的 x 轴从窗口时间中“分离”出来。这种情况下，改变曲线图中的选中显示点不会影响其它图表。

这种方式适用于在仍然进行的记录过程中分析曲线的早期点。

测量读取过程中，新的数值总是显示在焦点线上，前提是图表不处于“分离”状态。随着时间的推移，所有曲线均会向左侧移动。即使并未接收到新的数值也会按照这种方式移动，直至停止读取或者点击 stop 按钮停止记录为止。

曲线图中的控件



点击本按钮将曲线移动至选中测量之前的那个点上。



点击本按钮取消选中所有测量。



点击本按钮选中所有测量。不会影响显示状态：不会显示未选中测量。



点击本按钮将曲线移动至选中测量的下一个点上。



点击本按钮放大时间轴 (x)。



点击本按钮缩小时间轴 (x)。



点击本按钮放大选中测量的 y 轴。



点击本按钮缩小选中测量的 y 轴。



点击本按钮选中应当扩展至全长的 x 轴部分。



点击本按钮选中应当扩展至全高的 y 轴部分（影响选中的测量）。



点击本按钮选中应当扩展至整个图表大小的图表区域 (x 以及 y 方向)。还会影响 y 方向上的选中测量。



点击本按钮在 x 轴上显示整个曲线。如果正在读取，那么从 0 到当前点的所有数值均会扩展至整个轴的长度。

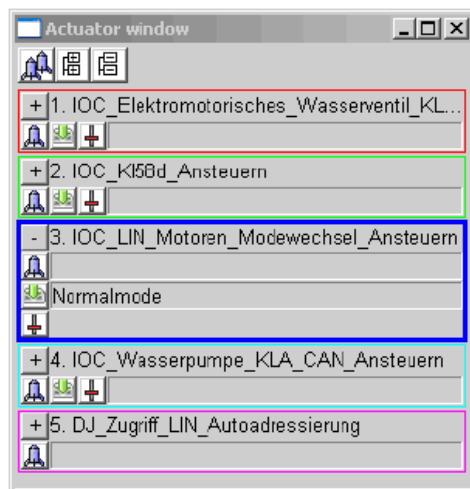


点击本按钮将选中测量的曲线最大化至整个图表高度。



点击本按钮从整个窗口的时间中分离 x 轴。分离的时候，更改焦点线的位置不会影响其它图表。时间滑块的位置也不会影响分离的曲线图。

驱动器窗口



彩色框包括了使用服务所需的所有控件。

 展开框

 收起框

 执行服务。采用之前输入（或者从文件中载入的）的参数值。

 打开窗口键入输入参数值。设置参数之后，不会执行任何服务。

 打开常规设置的滑块控件。更改数值会变更参数值并且执行服务。保存最后一次使用的参数值以便今后使用。

此外还有三个按钮：

 执行所有服务

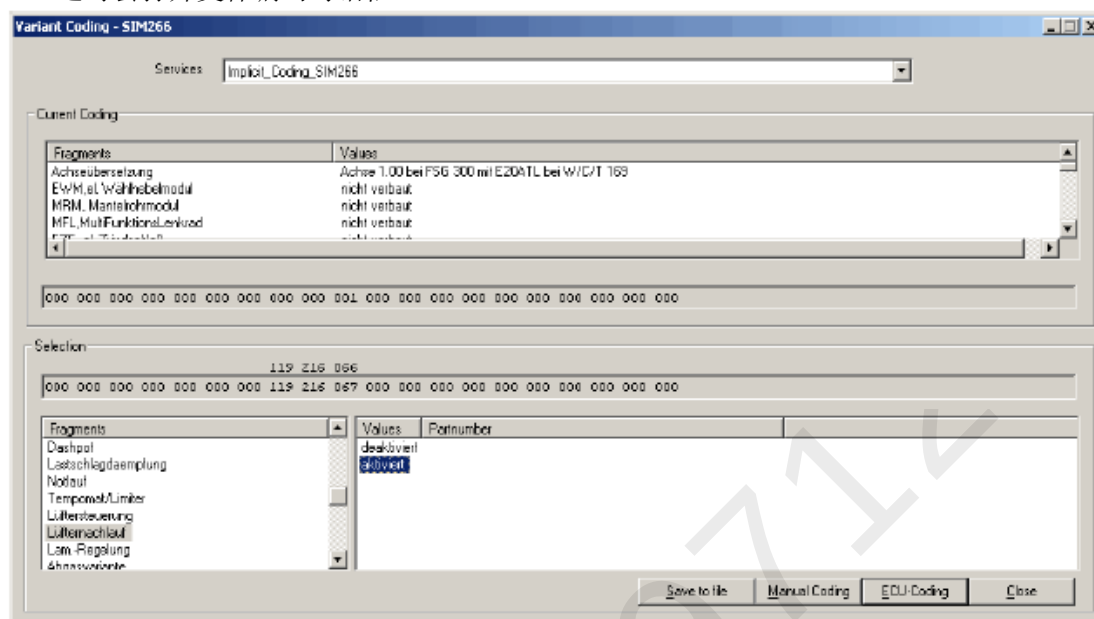
 展开所有框

 收起所有框

2.4.4.10 变体编码

可以采用 Ecouté 读取并更改变体编码的当前状态。操作步骤如下所示：

- 从 Ecouté 主菜单中选择 Coding / Variant coding。
- 这时会打开变体编码对话框。



如果针对当前 ECU 变体在系统说明中规定了变体编码的前置条件，那么会在打开窗口的时候执行此类前置条件。如果将带有输入参数的服务指定为前置条件，那么会显示适当的对话框，用户在其中可以输入各种参数。

服务

这一对话框区域内会显示所有 ECU 编码服务。如果从选择列表中选中某个服务，那么会在对话框下半部分显示所有片段以及“选中窗口”中的所有对应数值。

当前编码

在对话框打开的情况下，从 ECU 中读取并显示当前编码字符串。列表区域内会显示与编码匹配的片段以及数值（最多 4 行）。如果参数化中不存在任何编码，则在数值下方显示“unknown”。

根据 Vediamo.ini 中的设置，可以采用十进制或者十六进制格式显示编码字符串。在 Vediamo.ini 中[ECOUTE]部分中的“VarCodStringFormat”项指定编码字符串的格式。该项的取值为“Decimal”或者“Hexadecimal”。“Decimal”表示采用十进制格式显示编码字符串。“Hexadecimal”表示采用十六进制格式显示编码字符串。默认采用十进制（decimal）格式显示编码字符串。

选择

对话框的这个区域内会显示选定服务的所有参数化片段以及数值。弹出对话框的时候，Coding string 字段不包含任何数值。一旦从列表中选中某个数值，则会立即显示相应的编码字符串。此外，还评估了当前编码字符串以及选中编码字符串之间的差异，采用差异的数字数值标记两个字符串差异的位置（十进制：两个数值之间差异的绝对值；十六进制：变更比特位的比特位掩码）。

外部参数化编码片段：在对话框中，采用后缀（ex）的方式指明编码片段，此类片段采用外部文件（ccf 后缀）的方式进行参数化。

保存至文件

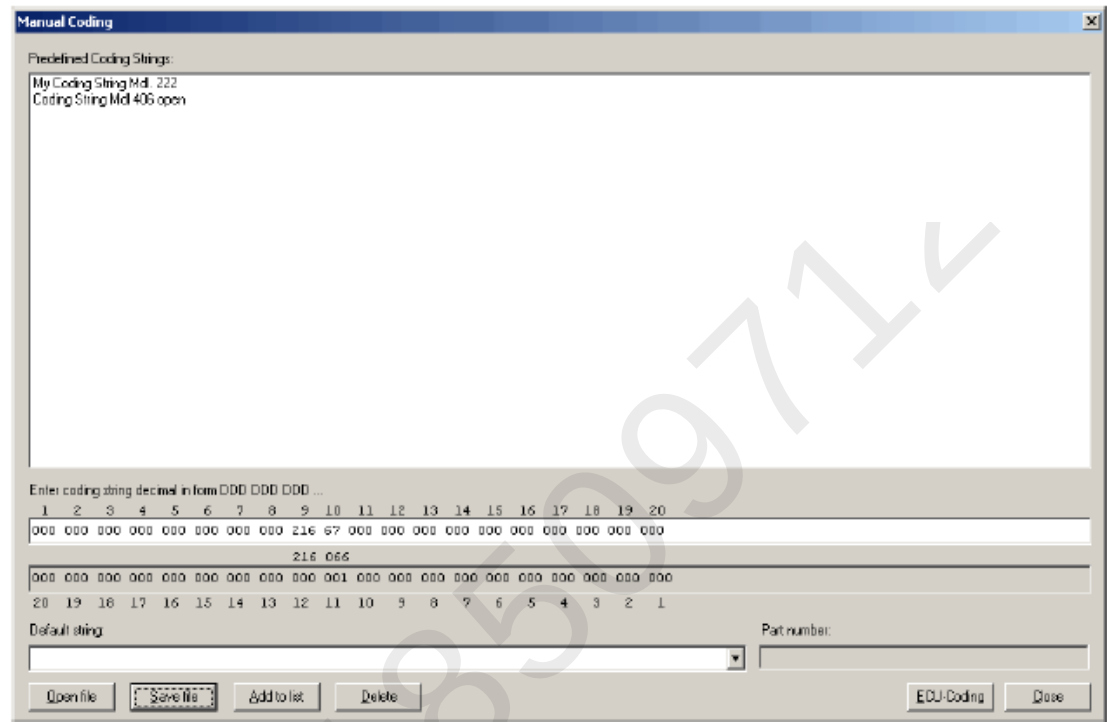
点击该按钮将变体编码保存在快照文件中。

关闭

点击本按钮关闭变体编码对话框。点击按钮之后，会执行系统说明中包括的所有变体编码相关的后置条件，执行方式与上述前置条件相同。如果在 VSB 中激活了 shut down cycle for coding 选项，冰箱在 Ecouste 中执行了编码，那么就会打开关机循环对话框。对话框会提示用户关闭点火器然后再重新打开点火器。

手动编码

点击按钮 Manual coding 会打开对话框，其中可以手动编辑编码字符串。



编辑文本框内的编码字符串是从选择字段中获取的。如果选择字段并不包括字符串，则会从当前编码字符串的字段中选取编码字符串。用户可以编辑编码字符串并且点击按钮 ECU-Coding 执行变体编码。点击 Close 按钮可以关闭对话框而不再进行编码。

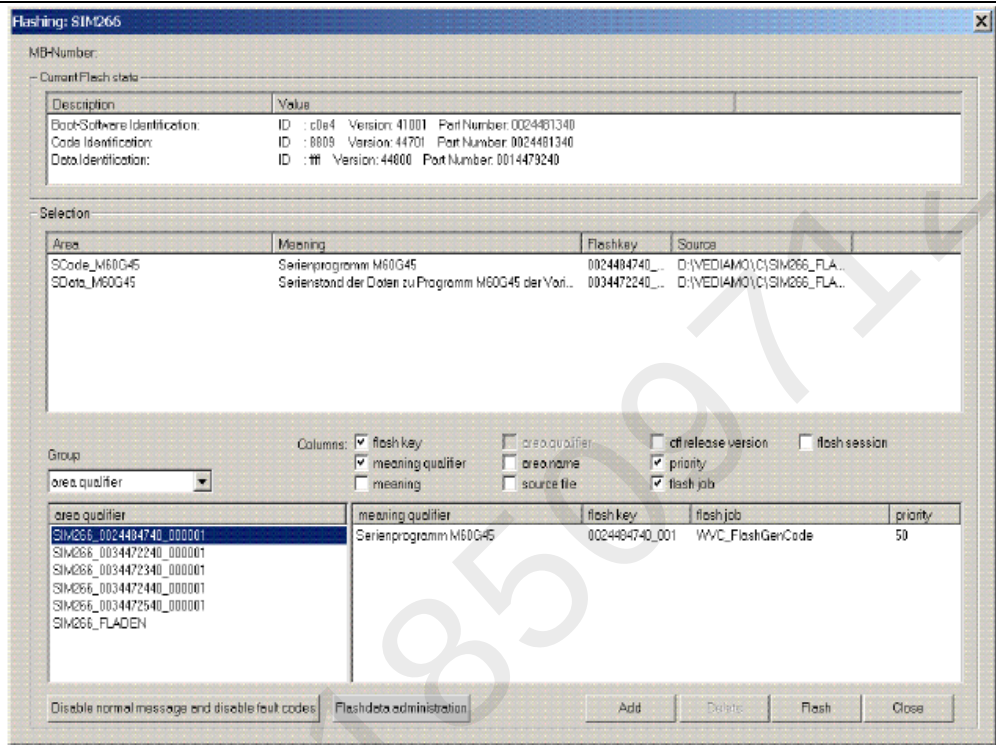
您可以命名自有的编码字符串，并且将其保存到列表中（Add to list 按钮）。还可以删除列表中的字符串（Delete 按钮）。编译的列表可以存储到文件中（Save file），或者可以从文件中载入之前编译的列表（Open file）。如果组合数量非常大，那么这一点就显得非常有用，而且使用自有的选项还有助于更高效地进行工作。

根据 Vediamo.ini 文件中的 VarCodStringFormat 参数，可以采用十进制或者十六进制显示文件中载入的编码字符串。字符串在文件中的存储格式可以是十进制、十六进制或者混合格式（比如，“123 001 5D 7F 111”），对于显示以及输入格式不会造成任何影响。

2.4.4.11 闪存处理

- Ecoute 允许您读取以及更改 ECU 闪存的当前状态。操作步骤如下所示：
- 从 Ecoute 主菜单中选择 Coding / Flash。
 - 这时会弹出闪存对话框。
- 如果没有已联系 ECU 的闪存数据，那么这时就不会弹出对话框。状态栏中显示的消息表明当前 ECU 不存在可用的闪存数据。

注意：
注意，ECU 数据（CBF）中 DIOGENES 名称必须与闪存数据（CFF）中输入的名称保持一致。这样可以防止在 ECU 中处理错误的



Ecoute 允许您在单一闪存进程中对多个区域进行闪存处理。系统说明 ECU 属性中规定了闪存进程中可以同时进行闪存处理的区域数量。默认设置是每个闪存进程一个区域。

当前闪存状态:

窗口的这一选项包括列表字段，其中显示了闪存数据的相关信息。此类信息在系统配置中的指定方式如下所示：

对于每个 ECU 变体来说，均可列出为用户提供 ECU 额外信息的所有服务，并且将其分配到属性对话框中（系统配置、功能 ID-Block Information）。如需显示此类信息，ECU 会针对 ECU 查询系统配置中输入的服务 ID，然后执行此类服务。服务 ID（比如，“Software Status”、“Code”、“Boot Block”等）均显示在左栏 Description 中，执行相关诊断服务获得的结果则显示在右栏 Value 中。

选择:

本窗口选项具有两个列表字段。第一个列表字段显示了用户组装的闪存。第二个列表字段显示了可用的闪存区域列表、闪存含义以及闪存键。用户可以针对显示的数据选择分组标准，具体做法是从 Group 多选框中进行选择。选中的分组会显示在左下方列表框中。

单选框 Columns 可用于选择选中分组应当显示的相关信息。可以通过拖拽的方式改变各栏所在位置。关闭对话框的时候会自动保存这些栏，下次打开的时候会自动恢复这些栏。点击栏标题可以更改列表项的排序方式。

如果在系统说明中针对 maximum number of areas to flash simultaneously 输入“1”，则在一个闪存进程中仅可在 Ecouté 中对一个闪存进行闪存处理。这种情况下，用户可以针对当前选定区域选择一个含义，或者一个闪存键，并且点击 Flash 按钮对 ECU 进行闪存处理。

如果 maximum number of areas to flash simultaneously 数值大于“1”，则点击 add 按钮可以首先在第一个列表字段中添加需要闪存处理的多个含义。然后点击 Flash 按钮，那么会按照排定次序设置该列表中的所有含义。此后，可以采用正确的次序对设定含义进行闪存处理。进程显示中会显示闪存进程的状态。闪存进程结束后，会提示用户关闭点火器然后中关机循环结束后再次打开点火器。

如果针对含义参数化选定为“late binding”，那么双击或者点击<RETURN>键即可弹出 Open File 对话框。用户可以从该对话框中选择适当的文件。可以在 Meaning 栏中的列表字段中输入选中的文件名称。然后点击 Flash 按钮启动闪存进程，这一点与早期绑定是完全一样的。

可以显示每个闪存含义的区段信息。从右侧列表框中选择闪存含义，然后右键点击鼠标。弹出的对话框中会显示当前选中含义的区段。一个区段可以理解为区域的起始地址以及闪存区域的长度。注意，一个区域的区段信息是可以改变的，这主要取决于设置汉语。根据诊断数据，区域/含义组合可以拥有多个区段。

按钮“disable normal message and disable fault codes”：

自从部署了 CAESAR3.0 之后，就不会在针对车辆 ECU 进行闪存处理的同时再出现自动禁用正常消息以及剩余总线参与者故障代码了（通过 OBD 套接字与车辆进行联系/测试仪与网关控制器相连，比如与 Powertrain CAN 相连）。

此类机制目前是通过 CAESAR2.0 提供的（仅支持关键字协议）。

从 CAESAR3.0 开筛查，由于可以针对一台车辆使用多种协议，因此应用程序必须提供“disabling normal message and fault codes”机制。

在采用固定标识符的功能消息的情况下对剩余总线参与者进行禁用的。通过此类标识符必须可以循环发送“tester present”，以便保持禁用处于使用状态。如需停止禁用功能，则必须发送另外一条消息，或者必须终止“tester present”。这种方式同样适用于抑制错误注册（不过，采用的是其它消息以及其它标识符）。

“Disabling normal message and fault codes”与具体协议有关，即，必须针对每种协议单独激活此类机制。

Ecouté 客户端通过按钮“disable normal message and disable fault codes”提供此类机制。

通过采用此类机制，可以获得更好的闪存性能。在针对车辆进行闪存处理的时候，可以实现一种机制，即，在静默方式进行闪存处理的过程中，切换剩余总线参与者，并且抑制错误注册。此项机制目前支持关键字以及 UDS 协议。用户点击相关按钮可以在适当时候采用手动方式激活/去活该机制。通过 API 1 可以激活/去活该机制。在“SpecialFunctions.ini”文件中规定了服务说明，该文件位于 Vediamo 程序文件夹中。用户可以编辑该文件。

按钮“Flashdata administration”：

如果按下按钮“Flashdata administration”，则会显示包括下列功能的窗口：

- 右侧树形控件显示了 CAESAR 中当前注册的.cff 文件。
- “Add”功能打开文件选择对话框，其中可以选择其它.cff 文件或者文件夹。这种情况下会立即在 CAESAR 中注册新选中的文件/选中文件夹中的文件。状态窗口内会显示可能发生的错误。
- “Remove”功能会从 CAESAR 中注销当前选中的文件。

- “Load standard list” 功能会卸载当前载入的所有闪存文件，并且载入文件[\\Flashfiles.cfg](#)中列出的闪存文件。如果该文件不存在，则不会卸载任何文件，而是显示报错消息。如果并未载入系统，则会禁用该按钮（变灰）。
- “Save standard list” 功能将当前载入的所有闪存文件的名称存储在文件[\\Flashfiles.cfg](#)中。如果并未载入系统，则会禁用该按钮（变灰）。
- “Load list” 功能提示.cfg 文件名称，卸载当前载入的所有闪存文件，并且载入指定的.cfg 文件中列出的闪存文件。
- “save list” 功能提示.cfg 文件名称，然后将当前载入的所有闪存文件的名称存储在指定文件中。
- “Close” 会关闭闪存数据管理对话框。
- 在树形控件的每个文件项中会显示相关的闪存文件内容（区域、含义、闪存键）。如果用户在闪存数据管理对话框中更改了任何闪存数据，那么后续会更新上级闪存对话框：会删除选中的含义列表，并且重建可用闪存的列表。
详情参见“闪存处理 ECU” ...

注意：闪存处理会覆盖 ECU 软件。
错误的闪存可能会损毁 ECU。

2.4.4.12 OBD2

OBD2 功能与扫描工具相应的 OBD2 或者 EOBD 规则类似，而且给予 Ecouté 用户直接访问多数重要诊断数据的权限（DTC – 诊断问题代码，当前数据，lambda 传感器专用 OBD 测试，点火，催化剂等），同时无需特定的 ECU 文件（*.cbf 文件）。

如需访问此类信息，则必须定义十个模式：

- 模式 1 – 当前诊断数据（PID）
- 模式 2 – 冻结框架（PID）
- 模式 3 – 排气相关的已存储 DTC
- 模式 4 – 清空排气相关的已存储 DTC
- 模式 5 – 氧气传感器测试结果
- 模式 6 – 诊断功能测试结果（MID）
- 模式 7 – 当前或者最近传动循环中排气相关的 DTC
- 模式 8 – 控制车载系统
- 模式 9 – 车辆信息（信息 ID）
- 模式 10 – 排气相关的永久性 DTC

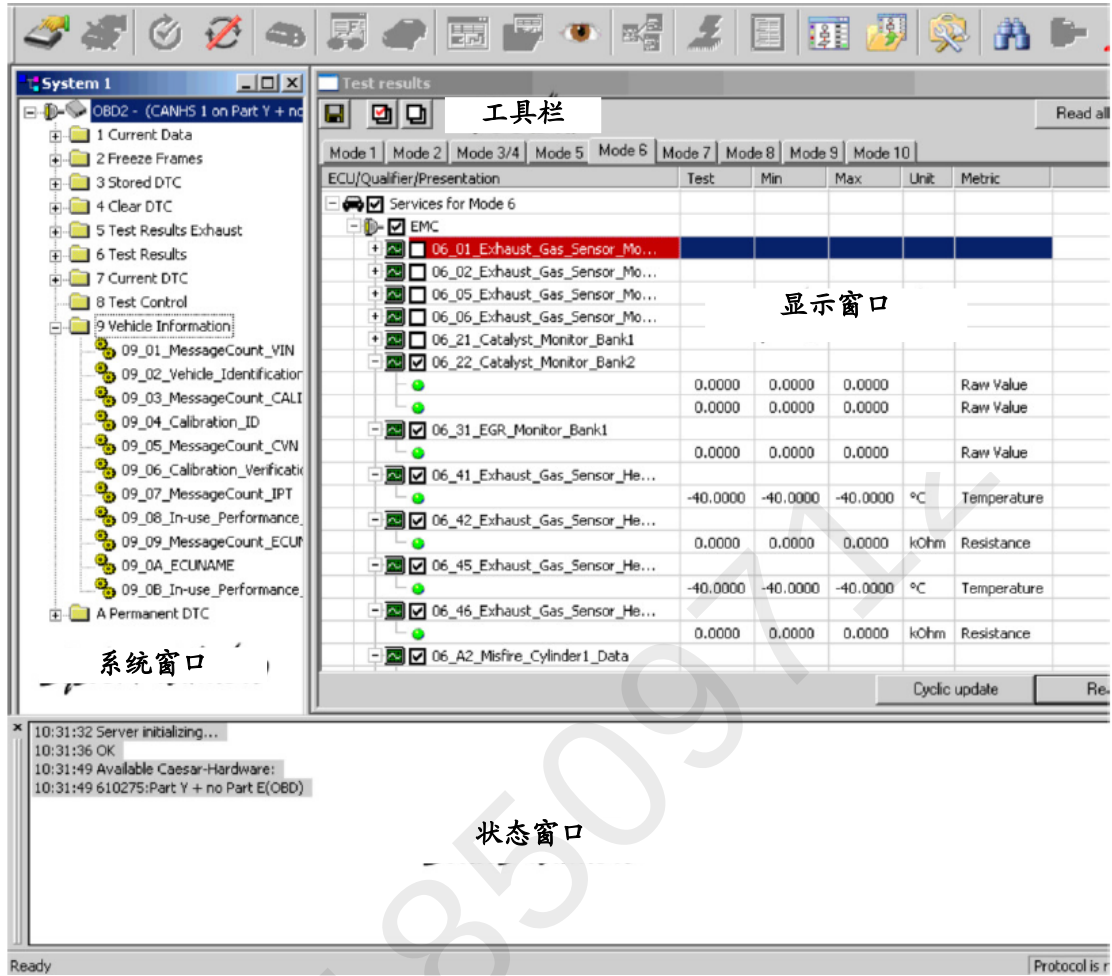
ISO-15765-4 规范中定义了 OBD2 功能标准（公路车辆 – 控制器局域网的诊断（CAN）-第 4 章：排放相关系统的需求）。

功能性

可以从菜单项 System / open OBD2 或者 close OBD2 中打开以及关闭 OBD2 功能，或者还可以点击 Ecouté 工具栏中的 OBD2 按钮。

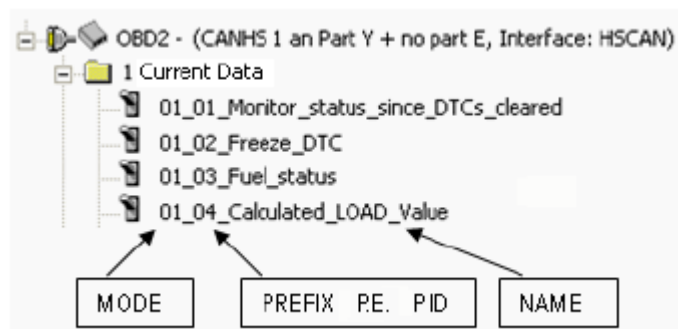
OBD2 窗口

打开 OBD2 功能之后，会打开两个独立的窗口：OBD2 系统窗口以及显示窗口。



OBD2 系统窗口

OBD2 系统窗口与 Ecout 系统窗口结构相同。其中包括所有十个 OBD2 模式的树形分支。每个分支均包括所有定义的服务（PID、TID、MID 以及信息 ID）：

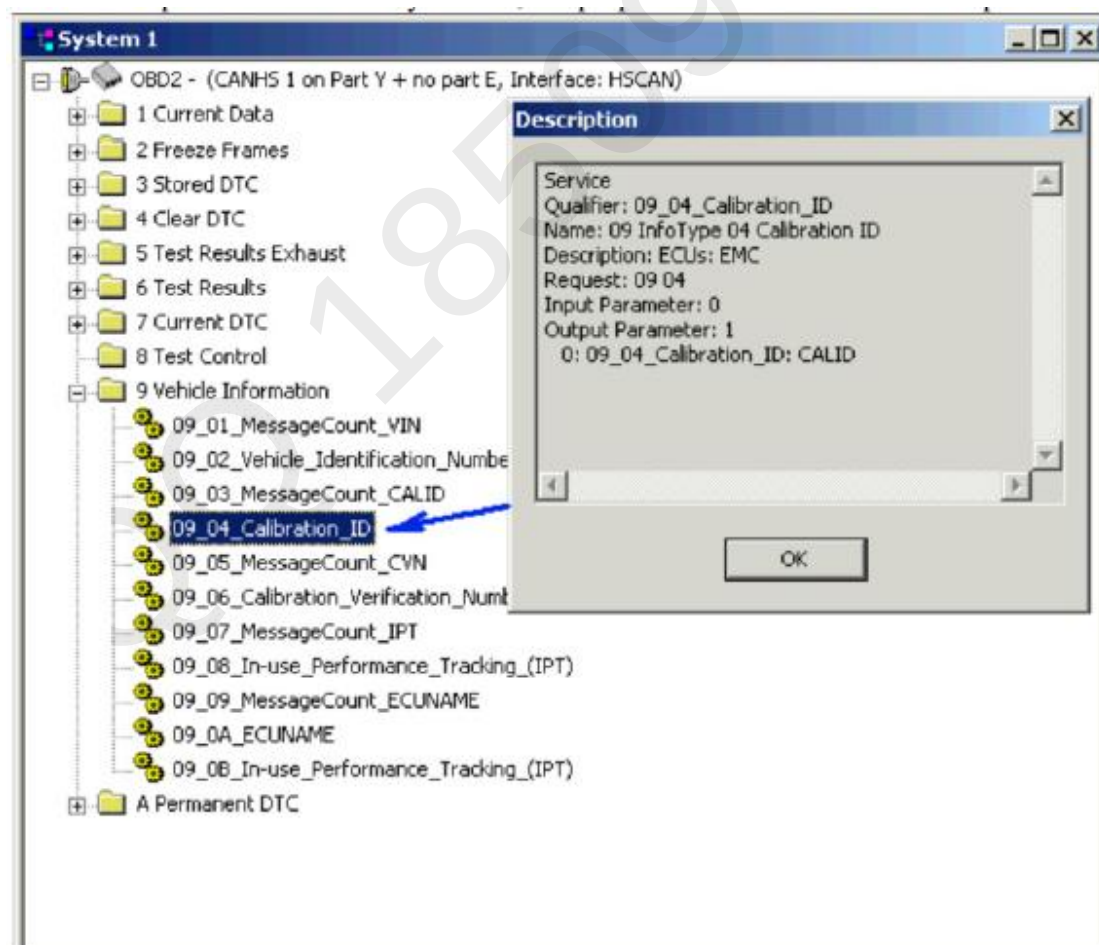


功能：

OBD2 系统窗口会显示所有定义的服务，而无论当前 ECU 是否支持此类服务。双击（或者选择并按下 ENTER）则会执行服务，并且将结果显示在状态窗口中。

从模式分支的上下文菜单中，Execute All 功能会针对每个 ECU 分别调用执行本模式支持的所有服务。这种情况下，每个结果会显示在单独一行中，随后是度量数值（含义的快捷方式）。

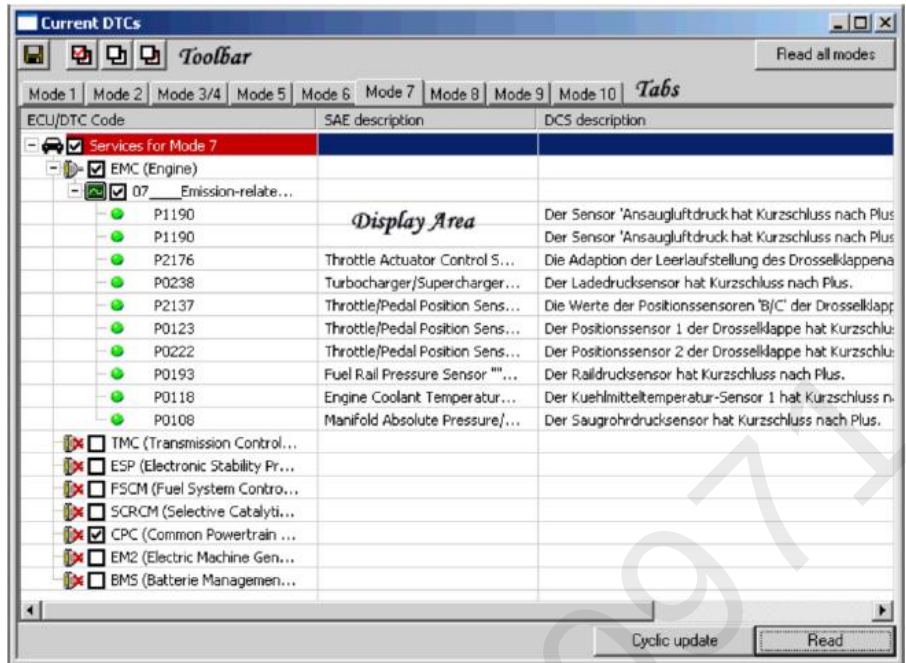
从每个服务的上下文菜单中，可以打开服务的属性。



OBD2 显示窗口

OBD2 显示窗口包括主要显示区域、通用以及特殊工具栏，其中包括按钮以及一个工具栏。

使用标签可以选择需要执行以及显示的模式。改变模式同样会导致激活或者去活下部特殊工具栏中的特殊按钮。



通用工具栏按钮的功能

	保存日志。所有模式页面中包括的所有读取的数值均会保存在 txt 或者 html 日志文件中。
	选择每一个模式页面中的树形控件中的所有元素。
	取消选择树形控件的所有元素。
	打开窗口的时候恢复选中状态。
Read all Modes	执行所有模式页面的所有选中服务。

关闭窗口的时候，所有服务以及 ECU 的选中状态，以及表格行的宽度（可以采用拖拽的方式进行变更）均保存在 OBDWin.ini 文件中。下次打开窗口的时候，会再次恢复此类状态。

特殊工具栏最多包括三个按钮，主要取决于选中的具体模式页面：

Read	执行当前模式页面中选中的所有服务。
Cyclic update	循环重复执行当前模式页面中选中的服务。
Clear errors	删除采用模式 4 存储的 DTC。

显示窗口中模式之间的差异在于：

点击标签可以选中 OBD 模式。每个模式均具有特殊的表结构。左栏包括树形视图，其中包括了所有 ECU 及其服务：

- 模式 1
 - 级别 1: ECU
 - 级别 2: 服务 – 所有支持的服务（Mode_PID_Name）

- 级别 3: 结果 – 一项服务可以传送一个以上的结果数值
- 模式 2
 - 级别 1: ECU
 - 级别 2: Fehlercode 级别 – 所有支持的服务 (Mode_PID_Name)
 - 级别 3: 结果 – 一项服务可以传送一个以上的结果数值
- 模式 3/4
 - 级别 1: ECU
 - 级别 2: 读取存储的 DTC 服务
 - 级别 3: 单一 DTC – 代码、SAE 以及 DCS/DCA 说明
- 模式 6
 - 级别 1: ECU
 - 级别 2: 服务 – 所有支持的服务 (Mode_PID_Name)
 - 级别 3: 结果 – 一项服务可以传送一个以上的结果数值
- 模式 7
 - 级别 1: ECU
 - 级别 2: 读取当前的 DTC 服务
 - 级别 3: 单一 DTC – 代码、SAE 以及 DCS/DCA 说明
- 模式 9
 - 级别 1: ECU
 - 级别 2: 服务 – 所有支持的服务 (Mode_PID_Name)
 - 级别 3: 结果 – 一项服务可以传送一个以上的结果数值
- 模式 10
 - 级别 1: ECU
 - 级别 2: 读取永久性的 DTC 服务
 - 级别 3: 单一 DTC – 代码、SAE 以及 DCS/DCA 说明

每个 ECU 以及服务均具有一个单选框。选中该单选框,则用户可以决定执行哪个服务。取消选中某个 ECU,则不会执行该 ECU 相关的所有服务。

输出文件/OBD 日志

结果可以保存到文本或者 HTML 文件中。点击“Save log”按钮,会提示用户保存其它信息:姓名、公司、车辆信息以及日志文件的名称与格式。

为了确保文件具有更好的可读性,应当在每行的结尾部分加上数值的名称/说明。

时机问题

在采用速度更快的 PC (双核以及四核) 以及 eCOM 诊断硬件的情况下,有时会出现问题。如果无法通过 OBD2 通讯方式访问 ECU,而且通讯工作正常,那么可以通过增大参数 OBD_P2_MAX 的数值 (250 以上) 以及 OBD_REQREPCOUNT 的数值 (1 以上) 解决此类问题。详情参见 INI 参数中的[SERVER]部分相关内容。

注意

上述参数数值越大,初始化过程中 OBD 通讯的速度就越慢。

2.4.4.13 配置 Ecouté 以及服务器选项

Ecouté 界面

使用鼠标即可操纵 Ecouté 窗口。程序结束的时候，会自动将窗口的位置以及大小保存到配置文件中，下次启动程序的时候再进行恢复。服务分组、测量分组以及驱动器分组均存储在 VSG、MWG 以及 STG 文件中。

Ecouté 会话

下列 Ecouté 属性存储在会话文件中（VSC），以便下次启动程序的时候进行恢复：

- 选中系统、ECU 联系状态以及通道分配
- 大小、位置与状态划分、选择与主窗口
- 所有打开的窗口（错误、测量、驱动器以及轨迹）及其位置与大小
- 呈现选项

其它选项

通过主菜单 Extras / Options 即可访问多种选项设置窗口。设置选项存储在 Vediamo.ini 中。可以采用 INI 编辑器或者常规文本编辑器对其进行更改。

选项分为三个组，每组均可在各自的窗口内进行编辑。通过窗口上部边缘位置的标签（CTRL+TAB）即可在不同窗口之间进行切换。

一般选项

名称与修饰符显示切换

通过唯一 ID 调用修饰符即可在 DIOGENES 中进入 DIOGENES 服务。修饰符唯一地标识了服务，Vediamo Java 程序总是使用修饰符。

此外，每个诊断服务均具有一个名称。该名称不具有唯一性，但是用户可以为服务指定具有一定含义的 ID。

在 Ecouté 中，通过选项对话框可以在显示修饰符与名称之间进行切换。仅需选择所需设置即可。名称或者修饰符的显示适用于 Ecouté 应用程序的所有窗口（选择窗口、测量分组、驱动器分组...）。

快照文件

Ecouté 允许存储快照文件。快照文件的路径以及文件名均可在选项对话框中进行设置（Extras /Options）。选择“...”按钮即可直接前往选项对话框中的“Snapshot File”项字段。这种情况下会弹出标准的文件选项对话框。前往目标存储文件夹，输入您选定的文件名。

硬件选项

可以在 StartCenter 应用程序中编辑硬件选项。

启动选项

载入最近会话

如果激活了 Load last session 字段，则会在下次启动 Ecouté 的时候自动载入上次选中的会话文件。

自动建立联系

Ecoute 会在下列三种不同情况下自动与 ECU 建立联系：

- 1、选中某个系统之后（最初联系）
- 2、由于外部影响而失去联系
- 3、用户尝试建立联系失败

可采用选项对话框独立配置各种情况。如果激活了自动联系，那么用户可以在状态窗口中查看相关消息。

在第 2 种情况以及第 3 种情况下，会尝试建立联系，直至已建立联系或者锁定或者切换系统，或者重置选项为止。

选中系统之后自动建立联系

如果设定为 After system selection，会在用户打开系统之后立即尝试与系统中包含的所有 ECU 建立联系。如果系统中的 ECU 超过 CAESAR 空闲通道，那么会针对无法分配给 CAESAR 通道的每个 ECU 显示一条错误消息。如果首次尝试与 Ecoute 系统选择建立联系失败，则提供其它选项（第 3 种情况）以便确定是否应当循环尝试与相应的 ECU 建立联系。

注意：

对于手动指令输入（Ecoute）来说，不会执行自动最初联系，即，在载入 VND 文件的情况下，不会评估自动联系的相关选项。

丢失联系之后自动建立联系

如果由于外部影响（断开物理连接，线路出现干扰，ECU 无法响应等）而导致与 ECU 失去联系，那么在设定 After contact loss 选项之后，会尝试与相应的 ECU 重新建立联系。

注意：

对于手动指令输入来说，尽管此类设置，仍然不会自动建立联系。

尝试联系失败之后自动建立连接

如果设定了 After failed contact attempt 选项，那么在尝试不成功之后会循环尝试与 ECU 建立联系。这种情况同样适用于选择系统之后最初的自动联系。

注意：

这种情况不适用于与 Java 程序建立联系。同样也不适用于采用手动指令输入情况下尝试联系。

自动执行初始化服务

可以在系统载入之后或者在建立通讯之后启动系统说明中包括的初始化服务。无论设定何种相关选项，Ecoute 客户端均会启动服务。系统说明中包含多种初始化程序。如果设定了 After system selection 选项，则会执行系统最初程序，如果设定 After contact 选项，则会执行 ECU 以及 ECU 变体初始化服务。

在以下情况下会启动初始化服务：

- 1、系统初始化程序（仅当激活 After system selection 选项的情况下）：每次启动或者选择新系统，Ecoute 客户端检查是否设定 After system selection 选项。如果 Ecoute 客户端通知诊断服务器，则应当执行系统初始化程序。如果 Java 程序包括在系统说明中，诊断服务器会执行该程序。
- 2、ECU 初始化服务以及 ECU 变体初始化服务（仅当激活 After contact 选项的情况下）：每次与系统中的一个 ECU 建立联系之后，Ecoute 客户端均会检查是否设置了 After contact 选项。如果设置了该选项，则 Ecoute 客户端会通知诊断服务器，即，应当执行相关的 ECU 初始化服务。这种情况下，如果此类服务包括在系统说明之中，诊断服务器会首先执行 ECU 相关的 ECU 初始化服务，并且在此后立即执行识别的变体或者基本变体的变体相关初始化服务。

如果同时存在多个 Ecouté 客户端，那么诊断服务器会确保仅在选择新系统之后，或者与 ECU 建立联系（新联系或者丢失联系之前）执行一次初始化服务。

相关的选项设置存储在 Vediamo.ini 文件的下列键中：

[Ecouté]

ExecuteInitSequence = <Key>

下列键值适用于<Key>：

- 0 未执行初始化服务
- 1 选择系统之后
- 2 联系之后
- 3 选择系统之后以及联系之后

日志记载选项

Ecouté 允许采用不同方式日志记载 ECU 与诊断服务器之间通讯，日志会存储在各种文件中。本窗口中可以指定日志类型。此外，还会显示日志数据存储的具体文件。

DCDI 通道（Ecouté 客户端）

这里指定 ECU 通讯在跟踪窗口内采用何种格式显示，在 Ecouté 中可以采用菜单项 Extras / Trace 调用该窗口。此外还会显示协议文件的完整名称。该名称总是与相关 ECU 的修饰符组合在一起。可能采用下列跟踪格式：

数据块

激活本选项，则会采用十六进制格式显示请求与响应，并且在 Ecouté 中进行日志记载。

字节以及计时

如果激活本选择，则会采用详细方式日志记载 ECU 与测试仪之间的通讯。监测会调用本功能。仅可在 ECU 通讯协议级别将监测用于调试目的。

有个按钮允许调用对话框以便选择文件夹，而 ECU 相关的数据就存储在这个文件夹中。日志记载期间，此类文件存储在子文件夹中名称为<ECU>Trace.log 的文件中，该文件夹采用系统名称。

DCDI 通道（诊断服务器）

必须通过诊断服务器将 Ecouté 中显示的用于日志记载 ECU 通讯的所有数据传送至 Ecouté 应用程序。此类传送过程需要一定的时间。因此，对于时间有严格要求的诊断过程来说，就需要将日志数据直接记录到诊断服务器中。诊断服务器中的日志文件完整名称显示在窗口中。可以采用以下各项控制服务器的行为：

诊断服务以及数据块

诊断服务器记录已执行的诊断服务的修饰符、十六进制格式的请求与响应，以及其它 ECU 相关的 CAESAR 日志记载信息。

字节以及计时

激活本选项之后，会采用日志方式详细记载 ECU 与测试仪之间的通讯，格式与上述格式相同。

不可能同时选择上述两个选项。

有个按钮允许调用对话框以便选择文件夹，而 ECU 相关的数据就日志记载在这个文件夹中。日志记载期间，此类文件存储在子文件夹中名称为<ECU> Kanal.log 的文件中，该文件夹采用系统名称。

在 Ecouté 的跟踪窗口中，如果跟踪 CAN 总线，那么会针对字节以及计时显示过滤器设置：

- 过滤激活/未激活情况
- 激活过滤器

- 当前使用的过滤器文件名称

如果根据 CAN 协议进行通讯，那么下列情况适用于日志记载 DCDI 通道通讯：会过滤累积的 CAN 监测数据。默认情况下，仅会日志记载具有来自当前 CBF 的相应 ECU 的 CAN ID 的 CAN 消息。对于 CAN ID 来说，会在日志中显示 ECU 修饰符。

在 Vediamo.ini 文件中，存在一个键：

[CAESAR]

"MonitoringFilterCANIDs"= 0 | 1 .

如果键值为 0，那么，与之前前述内容相同，不会过滤监测数据。如果该键采用其它键值，或者完全遗失键值，那么会过滤监测数据（=默认设置）。

作为过滤标准，每个 ECU 会使用两个比较符：

CP_REQUEST_CANIDENTIFIER 以及 CP_RESPONSE_CANIDENTIFIER。如果从监测缓冲中读取新的 CAN 消息，那么会检查 CAN ID 是否对应于其中一个比较符。如果不对应，则会过滤消息（在 Vediamo.ini 中激活了过滤的情况下）。

在日志的 CAN ID 位置输入 ECU 修饰符。

此外，可以针对所有需要的 CAN ID 在文本文件中指定过滤器。

文本文件格式如下所示：

示例：

```
;Assignment CAN-IDs ECU identifier
2016,ECU
2024,ECU
;
786,Unknown$312
528,Unknown$210
```

分号开始的行属于注释内容，系统会忽略该行。

对于过滤器指定内容来说，会首先采用十进制格式列出 CAN ID，然后，采用逗号分隔，然后显示标识符。

运行时可以更改文件内容。与 ECU 建立联系之后会更新过滤器信息。

在 Vediamo.ini 项中指定需要实现的过滤器文件名称

[CAESAR]

MonitoringFilterCANIDFile=

示例：

[CAESAR]

MonitoringFilterCANIDs=1

MonitoringFilterCANIDFile=C:\ProgramData\Vediamo\Config\CANIdFilter.txt

状态消息

如果激活了 Generate Status Logfile 选项，则会将 Ecouté 状态窗口中的所有输出写入文件中，文件名称显示在选项窗口内。

采用命令行参数设置选项

启动 Ecouté 的情况下，可以通过命令行输入 Vediamo.ini 文件中的任何键（在需要载入的会话或者系统说明名称旁边）。输入的键在运行的诊断会话持续时间内有效。

命令行语法如下所示：

Ecouté.exe /VI "[Section] Key Value ... [Section] Key Value ..."

选项/VI 或者 -VI 用于区分会话/系统说明名称以及 Vediamo.ini 键。需要上传的键列表必须放入引号之中。

如果采用/VI 选项启动 Ecouté, 则会在引导服务器之前将替代性 Vediamo.ini 键传送到服务器应用程序。在此类应用程序中, 会在 Vediamo.ini 文件的对应数值位置临时使用此类键。

例外

本机制仅会影响启动服务器的 Ecouté 实例。如果服务器已经在运行, 那么会忽略 Ecouté 命令行中指定的键。

无法覆盖在服务器端定义语言设置 ([Common]Language) 的键, 因为在操作系统启动服务器应用程序之后, 该键会直接起作用 (即使 Ecouté 可以为了覆盖而直接传送任何键之前)。

指定的键不仅会在服务器中起作用, 同样也可以在 Ecouté 中起作用, 即, Ecouté 访问 Vediamo.ini 键的时候, 也会使用临时数值。Ecouté 的例外情况属于十六进制编码项, 后者包括了各种 Ecouté 窗口的信息。

如果采用覆盖 Vediamo.ini 数值启动服务器, 那么会在每个 Ecouté 客户端信息窗口内显示下列信息, 此类客户端会与服务器建立联系 (示例):

"Server/Application was started with partially overwritten initialization parameters: [Ecouté] ProgramLogPath c:\Logtest\Client [Server] ProgramLogPath c:\Logtest\Server"

示例:

Ecouté.exe /vi "[CAESAR] USE_SIPartEDriver 1 [CAESAR] PinMapping 1"

采用激活的 CAESAR E 部件以及激活的销子映射启动服务器。

示例:

Ecouté.exe /vi "[CAESAR] USE_SIPartEDriver 1 [CAESAR] PinMapping 1 [Ecouté] UseFilters 0"

关闭过滤器功能的情况下启动 Ecouté, 而且服务器具有激活的 CAESAR E 部件以及激活的销子映射。

如果在诊断会话期间更改并且写入了 Vediamo.ini 项 (比如从 Ecouté Options 对话框中), 那么仅会在会话持续时间内临时性更新更改的键值, 而此类键值是在会话启动的时候上传的。不会在 Vediamo.ini 文件中进行永久性变更。

应用程序中规定文本语言

可以采用各种语言显示 Ecouté 应用程序 (菜单、消息等) 专用文本语言。可以采用 DLL (比如, 英语为 Ecouté_Res_EN.dll) 提供翻译的文本。在 Vediamo.ini 文件中的 [Common] 部分中的 LANGUAGE 中选择语言。

在 Ecouté 中, 还会输出应用程序中并未定义而是来自 DiagServer 的文本。此类文本的语言版本取决于 Vediamo.ini 文件的相同项, 该文件位于服务器上。翻译的服务器文本位于 VCommon_Res_XY.dll 中, 其中 XY 是语言缩写词。

针对诊断数据指定语言

可以采用各种语言显示 ECU 参数化文本以及来自 CAESAR 硬件的消息。必须提供适当的 CTF 文件。采用服务器端 Vediamo.ini 文件中的 [CAESAR] 部分中的 LANGUAGE 项选择语言。

此外, 在运行时还可以采用 ECU properties 对话框变更每个 ECU 的语言。此类变更会影响本 ECU, 并且会影响与同一个 DiagServer 相连的所有客户端应用程序。

2.4.4.14 宏

在 Ecouté 中，宏主要用于记录、存储以及后续传送并且执行某种频繁调用的功能序列。宏存储为文本文件，扩展名为.mak。

可以通过菜单选项 Extras / Macro 使用宏功能。

目前可以提供下列功能：

Record Macro:

调用本功能之后，会记录此后在 Ecouté 中执行的各个动作。“Record Macro”显示在右上角，列出了记录的宏指令。记录的结束标志为“End Macro recording”。然后必须在弹出的对话框内输入需要存储宏的文件名称。

Execute Macro:

调用本功能之后，可以从弹出的文件选择对话框中选择之前记录宏的文件，此后即可执行该文件。同时仅可记录或者执行一个宏。

还可以在 Ecouté 启动的时候，采用命令行参数的方式输入宏文件的名称。然后 Ecouté 会来启动之后执行相应的宏文件。

可以从 Extras / Macro 菜单中之间选择上次执行的三个宏。

在执行以及记录宏的时候，仅可考虑单一功能的调用次序。无法记录或者重现单一功能调用之间的时间间隔。

可以在宏中记录并且重现 Ecouté 的以下动作：

- 选择系统
- ECU：建立/终止联系
- 读取/删除错误
- 在文件中存储错误数据
- 关闭错误窗口
- 打开测量/驱动器窗口，读取数值
- 在文件中存储测量/驱动器窗口
- 关闭测量/驱动器窗口
- 调整驱动器
- 采用图形方式显示、打开、读取 x 秒，并且存储测量数值。在宏执行过程中，至少在记录过程中打开了图形窗口，则会一直循环读取图形窗口中的测量数值。
- 执行所有服务，必要的情况下包括准备情况。可以从 Services（包括 Java 程序）菜单项的树形控件中进行选择。
- 各种编码：针对选定编码服务对单一片段数值进行设置/编码。（非手动编码）
- 闪存处理
- 快速测试
- 延迟：可以将某个延迟间期插入到宏中。第一个延迟项必须采用编辑器以手动方式插入到宏中。比如，等待 5 秒：“Delay||5000”。在回放宏的时候，会在相应的位置根据插入的延迟时间（以毫秒为单位）中断执行过程。
- 结束程序

在回放宏的时候，可以交替采用手动方式指定服务的输入参数。必须采用编辑器编辑宏文件，此后方可查询该文件中的输入参数：

如果需要查询宏中某一行的参数，则必须在参数的位置指定一系列的“??”，然后可以选择，然后可以选择采用某个文本，然后再次可以选择采用“??”然后是默认数值。

文本仅作为用户的注释。比如：宏记录之后某几行宏如下所示：

...

```
SetInputParam||CR3||ADJ_Injektorklassierung_ZYL4||0||Kl. 3  
ExecuteService||CR3||ADJ_Injektorklassierung_ZYL4
```

...

现在,每次执行的时候需要查询服务 ADJ_Injektorklassierung_ZYL4 的第 0 个输入参数。
宏文件可以采取以下方式:

...

```
SetInputParam||CR3||ADJ_Injektorklassierung_ZYL4||0||??Inj.Klassierung Zyl.4:??Kl. 3  
ExecuteService||CR3||ADJ_Injektorklassierung_ZYL4
```

...

可以将宏选定为初始化服务。Ecouste 会将具有<SystemName>_SysInit.mak 以及
<ECUQualifier>_EcuInit.mak 特定名称的宏视为初始化服务。在加载新系统的时候,ECU 会
检查当前系统文件夹中是否存在名称为...SysInit.mak 的宏,如果有,则会执行该宏。

如果已经与 ECU 建立了联系,则 Ecouste 会检查当前系统文件夹中是否存在名称
为...Ecunit.mak 的宏。如果存在,则会执行该宏。系统名称或者 ECU 修饰符必须位于此类
宏的名称之前。

2.4.4.15 Java 程序

如果存在各种程序(Java 程序或者进程,早期版本称之为脚本),则会显示在各自文件
夹的选择窗口内。双击(或者选中后点击回车键)即可像其它服务那样启动此类程序。正确
执行的前置条件是正确安装了 Java 运行时,而且 INI 文件中的所有涉及均正确无误,而且
执行程序的 Java 程序正确无误。

每个 Java 程序均具有预定义命令行参数,此类参数存储在系统说明文件中。双击系统
窗口中的程序,即可采用预定义参数启动该程序。还可以采用不同的参数以多种方式启动同
一个 Java 程序。

如需采用不同于系统说明中存储的参数来启动 Java 程序,需要右键点击程序符号标志
调出上下文菜单。这时会弹出输入窗口,在其中可以覆盖原有的各个参数。

一旦 Java 程序启动之后,会打开窗口显示目前活跃的 Java 程序。必要的情况下,可以
使用该窗口放弃 Java 程序。

输出以及用户交互主要取决于执行的 Java 程序。请阅读句柄功能文档,以便获取关于
Vediamo Java 程序提供的各种功能。通过 Java 程序提供的输出或者用户交互的最基本功能
如下所示:

- Ecouste 状态窗口中的输出
- 接受用户发出的是/否响应
- 接收用户发出的文本输入
- 信号提示音

2.4.4.16 程序生成器

通过程序生成器可以采用 GUI 来创建以及编辑复杂的程序而无需了解任何 Java 知识。程序生成器在 Ecoule 应用程序中以窗口的形式显示。其中包括四个区域（符号标志集合、绘制页、属性对话框、工具条）。可以在其它 Ecoule 窗口旁边打开该窗口，因此可以将系统窗口中的各种服务直接拖拽到动作元素上。

如需采用新的或者已经存在的程序打开程序生成器，可以采用菜单项 System / Routines / New Routine 或者 System / Routines / Open Routine。

还可以直接从系统树形窗口或者从菜单中启动某个程序，前提是此类程序需要定义为标准对象。

使用实例

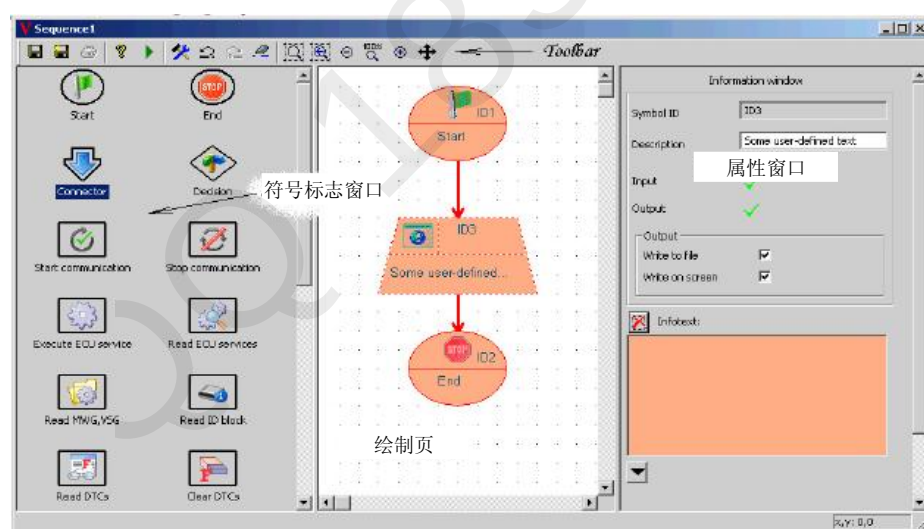
程序生成器允许用户采用图形化设计流程图的方式来定义复杂的程序，具体包括以下功能：

- 一次性或者循环读取数据
- 设置调整以及控制驱动器
- 基于其它动作或者基于用户输入的描述
- 从用户或者从文本文件获取输入
- 延迟以及时间戳
- 采用变量以及赋值
- 正则表达式

窗口元素

程序生成器窗口主要包括一个具有各种控件（工具条）的边框以及三个主要区域：

- 左侧区域 – 符号标志窗口
- 中间区域 – 绘制窗口
- 右侧区域 – 树形窗口



在工具条中对控件元素进行分组。根据其状态，其中的某些控件处于未激活状态，以便防止被误用。下表介绍了程序生成器的所有控件。

程序生成器的工具条

-  保存按钮 – 将程序保存在*.xml 文件中，并会改变文件名称。
-  另存为按钮 – 将该程序保存在到新的*.xml 文件中。
-  打印按钮 – 打印整个程序或者绘制页上的所有内容。
-  确认按钮 – 手动控制/确认基于具体方案的程序。采用图形化方式显示确认结果。

-  执行按钮 – 执行程序。
-  选择按钮 – 打开窗口以便显示并且编辑程序生成器的各个选项。
-  撤销按钮 – 最多可撤销 10 次采用控件执行的各种动作。
-  重做按钮 – 重做撤销的动作。
-  删除按钮 – 删除页面中的动作符号标志。
-  缩放片段按钮 – 放大用户定义的某个片段。
-  缩放按钮 – 显示上一个片段。
-  缩小按钮 – 缩小页面。
-  放大按钮 – 放大页面。
-  切换按钮 – 关闭缩放/显示整个区域。
-  移动片段按钮 – 移动页面中的某个定义的区域。

符号标志窗口

左侧区域包括所有可用的动作符号标志。通过按钮 **Options** 可以单独记录各种符号标志。将鼠标指针悬停在某个符号标志上方则可以显示工具提示。

可以将符号标志插入（通过拖拽的方式）到绘制页中。不可对符号标志进行选中或者缩放。插入之后，右侧区域会显示新的动作元素的属性窗口。

程序生成器的动作符号标志

-  程序的起点。仅可使用一次。
-  程序的终点。仅可使用一次。
-  两个动作之间的连接部分。确定了执行的次序。
-  比较两个数值 – 动作或者用户输入的结果 – 采用预先确定的操作符。比较结果是/否决定了执行图表中的两条路径中的某一条路径。
-  与 ECU 建立联系。
-  关闭与 ECU 的联系。
-  执行诊断服务。服务的结果存储起来以备日后使用。
-  读取多个测量数据。
-  从 VSG 或者 MWG 文件中读取数据。
-  读取 ECU 的 ID 块。
-  读取 ECU 的诊断问题代码。
-  清空 ECU 的 DTC。
-  打开编辑窗口。输入的文本存储在程序中以便日后使用。
-  显示用户定义消息的对话框。
-  打开文本文件，存储器中内容以便日后使用。
-  打开消息框，并且请求用户作出相关决策。存储是/否结果以便日后使用。
-  针对定义的按键以停止执行。

-  创建时间戳。采用毫秒为单位存储数值以便日后使用。
-  将某些文本输出到文件中或者状态窗口中。
-  打开文本文件，读取并且存储器中内容以便日后使用。
-  定义数值为 0 的变量。如需变更该数值，则必须采用赋值方式。存储该数值以便日后使用。
-  将某个数值赋值给某个变量。
-  采用正则表达式过滤文本。存储过滤器结果以便日后使用。
-  执行另外一个程序。

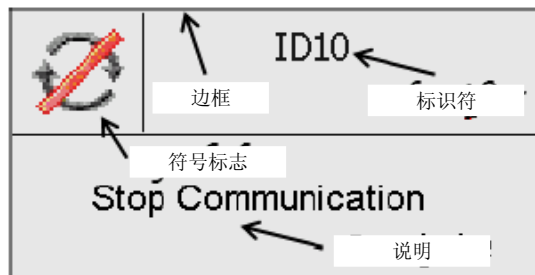
任何动作符号标志的边框均具有特殊的含义：

-  正方形表示需要执行的动作。
-  菱形表示决策（是/否）。
-  梯形表示显示对话框。
-  启动/停止某个程序。

绘制页（中间区域）：

在该区域内可以插入、连接、记录以及编辑动作符号标志。可以切换打开以及关闭光栅。如果内容超过了窗口，则使用滚动条来上下左右移动相关内容。

动作符号标志



边框：矩形、菱形、梯形或者圆形。

标识符：每个动作均具有自动分配的唯一 ID。

符号标志：动作图标显示在区域的上部。

说明：可以在属性窗口内编辑说明文本。

属性窗口（右侧区域）

如果在绘制页选中了某个符号标志，则可以在右侧的属性区域内设定元素属性。此处您可以更改说明、日志记录、输出或者输入等。

如果没有选中任何动作符号标志，则会显示序列的属性。

注意：共有两种程序：保护以及未保护。可以从 **Save As** 窗口中选择类型。

未保护：

在普通 XLM 文档文件中保存程序。任何用户均可读取、编辑以及存储程序。

保护：

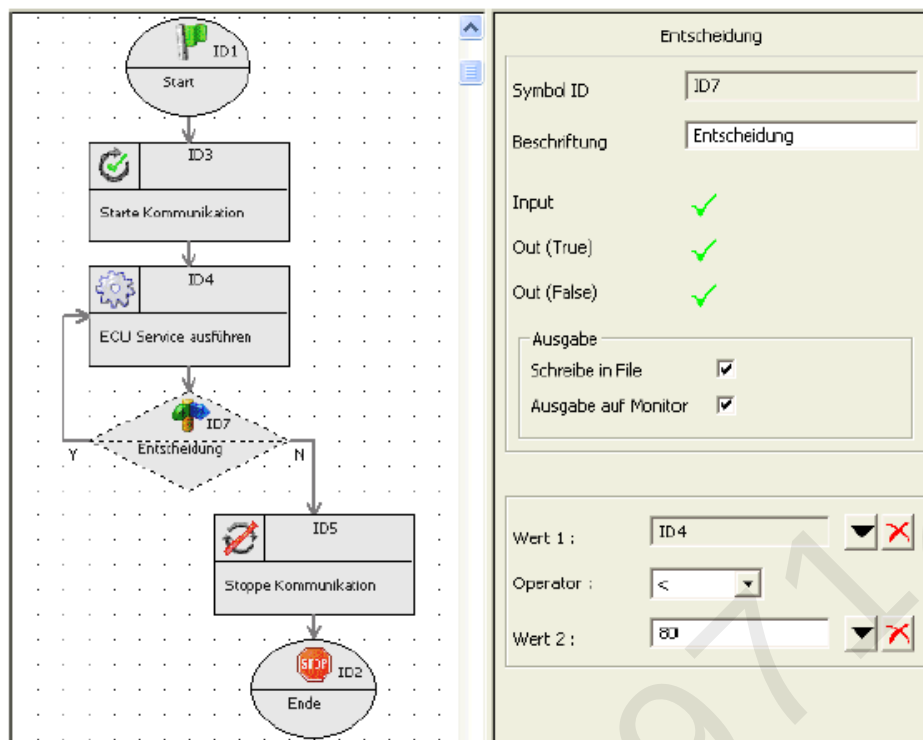
采用 CRC 校验和将程序保存在加密的 XML 文档文件中。可以设定密码。可以保护程序防止对其进行读取或者写入（编辑）。

确认

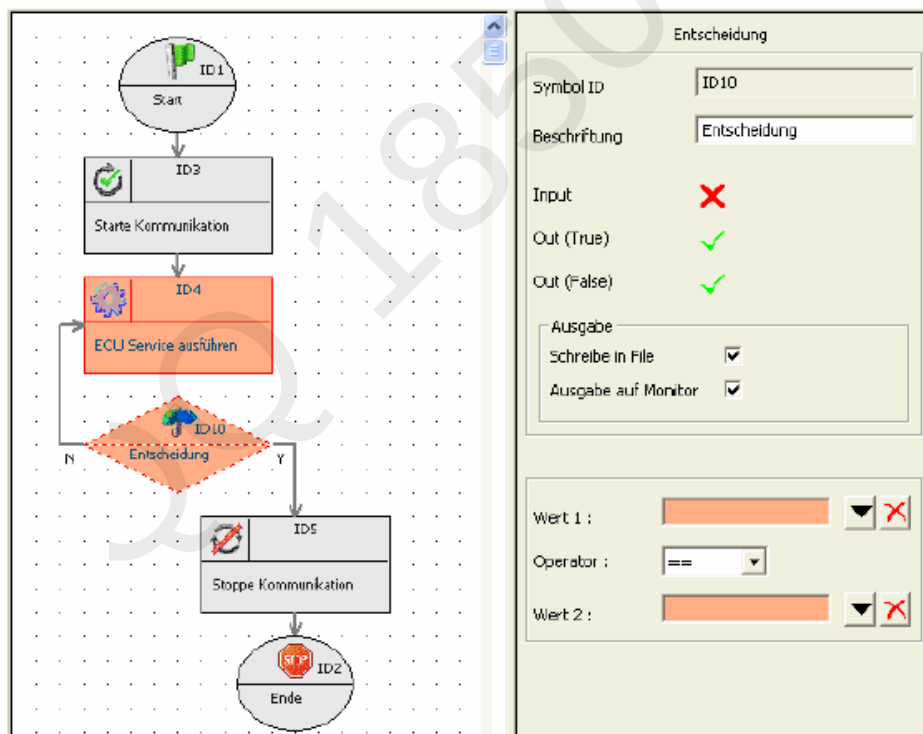
在编辑以及载入之后，会对程序进行确认。采用图形化方式显示结果。

- 程序有效
 - 动作符号标志为灰色。在属性窗口中，所有连接均显示为灰色 OK 符号标志。
- 程序无效
 - 错误的动作元素显示为红色背景。在属性窗口中，红色十字标记表示属性不正确。红色背景的输入框为强制信息。

有效的程序



无效的程序



自动确认

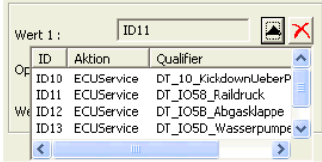
每次变化之后均可以在选项(工具条中的按钮)中切换打开或者关闭自动确认功能。

手动确认

如果关闭了自动确认，则可以点击工具条上的按钮手动执行。

属性窗口 – 采用其它动作的返回数值

根据绘制页中的聚焦的动作符号标志，则属性窗口中将包含上下文敏感的输入框。您可以在这里使用其它动作的结果，或者让用户在运行时采用手动方式输入数值。

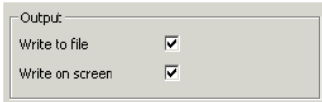


在任何动作中，如果必须采用其它动作的结果，列表框中允许从所有可用结果中进行选择。选择其中一个结果会覆盖文本框中的内容。如果选中了某个结果，则不可手动编辑文本框。仅可从列表框中选择另外一个结果。

按钮可以删除文本框中的内容，而且允许手动输入数值。

保存结果

在运行时，可以将所有执行的动作日志记载到文件及/或控制台中。默认情况下，会设置两个选项。此类选项可以在每个动作的属性窗口中进行变更。



写入文件：

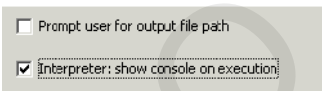
设定本选项会导致动作日志记载到文件中。文件的默认路径是序列文件的文件夹。文件名称包括采用时间戳的序列的文件名以及扩展名 LOG。

在程序的属性窗口中，日志文件路径的选项可以变更。在程序执行结束的时候，会弹出文件打开对话框，要求用户选择路径以及文件名。

写入窗口中：

设置本选项会在 Ecout 的状态窗口中显示所有日志信息。

变更默认日志文件路径：



日志文件路径：

如需覆盖默认日志文件路径名称，则打开程序的属性窗口（点击绘制页的空白区域）..选择选项 “User input for log file path” 会导致在执行程序之后打开文件对话框。该对话框中，用户可以选择路径以便保存输出日志。

打开控制台：

程序中的属性中的选项 “Open console during execution” 会导致弹出单独日志窗口，以便日志记录输出。

程序生成器的配置

在工具条中点击按钮 “routine generator options”，即可打开具有多个选项的窗口。尤其是 “Configure action symbols” 按钮允许重新分类、显示或者隐藏动作符号标志。

其它选项名称则具有自解释功能。

V Sequence generator Settings [X]

Settings

Font / size: ...

Catch distance:

Grid:

Cursor grid:

Configure tasks / symbols

View

☒ Show ID

☒ Show Description

☒ Grid

Misc

☒ Validation on any change

☒ Execution: show progress in sequence

Apply

Cancel

2.4.4.17 标准对象

标准对象（比如，解锁）通常指代为别名，因为此类对象还指向另外一个服务（比如，取决于 ECU，标准对象解锁之后是 DiagJob DJ_Entriegeln 或者服务 FN_Zugriffsberechtigung），而且给予此类服务一个通用的名称。这样就可以给予相同功能内容的服务以相同的名称，而不论 DIOGENES 数据中采用何种不同的名称。可以从两个不同级别提供标准对象。一种情况下，是 ECU 系统中系统特定的标准对象。此类标准对象通常会影响到多个 ECU。在其它情况下，可以在 Vediamo 系统配置中针对单个 ECU 定义独立标准对象。

标准对象在 ECU 中接受特别的处理。对于子文件夹 Name 中输入的每个对象而言，会在菜单 Name 中生成一个菜单项。如果不存在此类名称的菜单项，就会创建此类菜单项。这样就可以采用标准对象定制 Ecouste 接口，并且在相关的分组中对经常使用的服务进行分类（其中包括 Java 程序）。

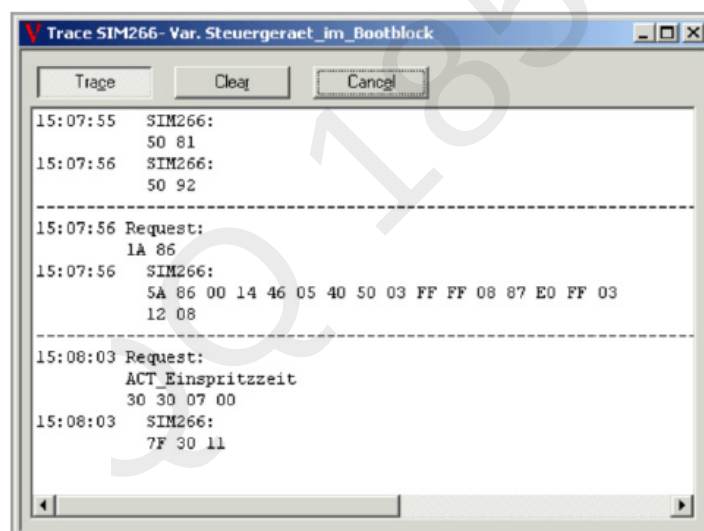
如果需要执行标准对象，那么可以从选择窗口（双击或者点击回车键）中进行启动，或者选择相应的菜单项。

注意：

如需采用下划线字符（热键）创建菜单项，则必须在名称之前采用“&”。如果在文件夹“&Errors”中输入标准对象“&Special”（详情参见系统配置部分），则 Ecouste Error 菜单中显示了对应于“S”键的额外的菜单项 Special。如果并未包括“&”，则会在菜单项 Special 中创建二级菜单 Errors（不具有下划线），同时也并未分配热键。

2.4.4.18 显示跟踪以及监测数据

功能 Extras / Trace window 允许 ECU 相关的通讯信息以及需要在单独窗口内显示相应的 ECU。可以停止或者再次启动显示内容。显示的信息会自动存储在日志文件中。如果可以提供多个 ECU，则必须在选择功能之后必须选择适当的 ECU。该 ECU 的通讯信息显示了下列窗口以及格式：



采用按钮 Cancel 选择窗口。

按钮 Trace 可以停止或者启动输出显示。如果激活了 Trace，则按钮就显示为按下状态。如果停止了 Trace，则按钮会再次弹起。

可以采用按钮 Clear 删除窗口中的全部内容。

CTRL+C 组合键会将窗口中选中的所有文本复制到剪贴板中以便日后进行处理。

会将输出内容自动写入名称为<ECU>Trace.log 的文件中。（<ECU>是 ECU 的名称，比如，ME20）。所有新信息均会附加到该文件的末尾。

通道监控

本功能允许针对所有可用的 CAESAR 连接部分显示跟踪窗口。Ecoute/Vediamo 在“listen only”模式中，后者主要涉及这种情况下正在进行的通讯，即，不会向 ECU 发送任何数据。

总是采用“Bytes & Timing”格式显示数据。

在启动功能之后，CAESAR 协议、连接以及是否应当使用 CAN 协议，则必须指定波特率。

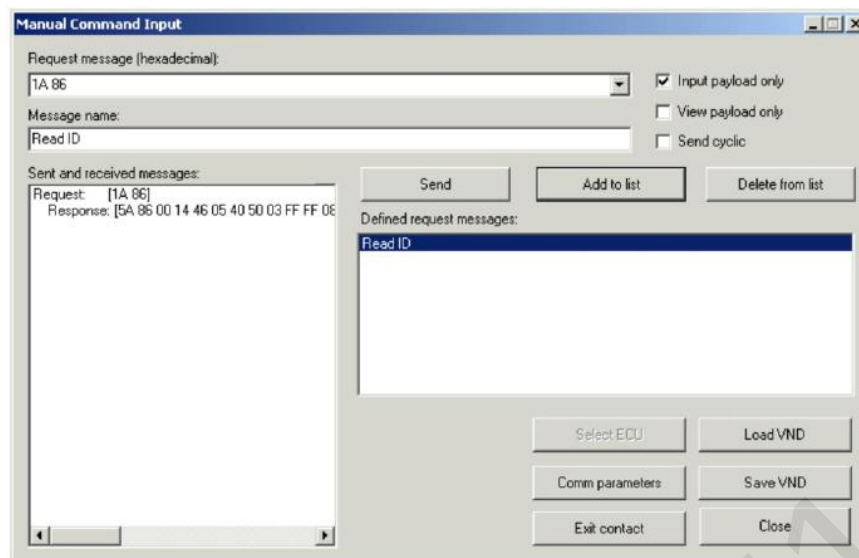
注意：

通道监控需要 CAESAR 资源。如果，比如，采用了 K 线路 1，则不会针对正常诊断提供该资源，直至关闭监控对话框为止。

2Q 1850971v

2.4.4.19 手动命令输入

采用菜单项 ECU / Manual Command Input 打开菜单指令输入：



手动指令输入接口的各个部件：

输入/输出框 Request message（十六进制）

- 输入消息：
输入需要采用十六进制格式（比特数值）发送的消息。可以将消息存储在 VND 中（详情参见：添加至清单）。此外，还可以提供之前的消息。
- 显示定义的消息：
如果已经载入了 VND，在 Defined request messages 列表中选择的消息字节均会再次显示。

输入/输出框 Message name

- 输入消息：在 Request Message 输入框中输入定义的消息名称。
- 定义消息的显示：如果已经载入了 VND，则会在在此显示 Defined request messages 列表中选定的消息名称。

选项 Input payload only

选项 option Input payload only 指定了应当如何处理输入行的输入信息 – 发送输入的消息或者首先创建协议投诉消息，而且将输入的字节包含为有效载荷。

选项 Display payload only

选项 Display payload only 指定了应当如何显示接收的消息内容 – 仅显示有效负载，或者完整的消息，其中包头部以及校验和。

选项 Send cyclic

选项 Send cyclic 指定在 ECU（最后一次）回复之后，发送的请求消息是否应当如何自动重发。

按钮 Send

可以采用 Send 按钮将请求消息发送给 ECU。在接收或者循环发送以及接收的时候，Send 按钮具有标签以及功能 Cancel。可以采用本按钮中止响应消息系列（其中包括 ECU 对于某些请求的响应）。最后一次消息之后（或者终止回复系列），按钮则恢复原有的标签以及功能。

按钮 Add to list

按钮 **Add to list** 可以用于将新定义的消息插入到 VND 之中。首先放入请求消息中，然后是对应的消息名称。

按钮 **Delete from list**

采用 **Delete from list** 删除在定义的消息列表中选中的消息。如果后续存储了 VND，则可以从 VND 中永久性地移除删除的消息。

输出框 **Sent and received messages**

输出框 **Sent and received messages** 中显示了所有通讯报文。输出主要取决于选项 **Display payload only** 的设置。

输出框 **Defined request messages**

本输出框 **Defined request messages** 显示了当前定义的消息。

按钮 **Select ECU**

如果打开的系统具有一个以上的 ECU，则可以采用 **Select ECU** 按钮选择需要联系的 ECU（详情参见打开系统的情况下建立联系）。对于未打开的系统而言，本按钮会打开 ECU 参数输入对话框，器中可以输入协议以及 ECU（详情参见未打开系统的情况下建立联系）。

按钮 **Load VND**

使用本按钮载入 VND。

按钮 **Establish contact**

按钮 **Establish contact** 控制与 ECU 的联系，并且显示当前的通讯状态。本按钮具有两个状态：

建立联系

未发起 ECU。使用本按钮建立联系。

终止联系

已发起 ECU。使用本按钮终止联系。

按钮 **Comm parameters**

可以采用 **Comm parameters** 按钮来设定协议特定的通讯参数。

按钮 **Save VND**

点击 **Save VND** 之后，会弹出 **Save file as** 对话框。针对您的 VND 选择适当的文件夹，并且在选择的文件名中存储该文件。

选项 **Monitor**

采用本选项在诊断服务器上打开或者关闭监测跟踪。

按钮 **Close**

采用 **Close** 终止手动指令输入。关闭窗口之后，会恢复打开窗口之前与 ECU 的联系的状态。

可以打开手动指令输入，而是否载入系统无关。但是，建立联系的流程存在差异。因此可以采用下列方式说明两种情况。

手动指令输入，建立 ECU 与打开的系统的联系

- 如果系统仅具有一个 ECU，那么可以采用 **Establish contact** 按钮建立联系。
- 如果系统具有多个 ECU，采用 **Select ECU** 按钮。这会弹出 **Select ECU** 对话框。选择需要建立联系的 ECU。然后点击 **Establish contact** 按钮。
- 一旦建立了与 ECU 的联系，则按钮上的标签会变更为 **End contact**。此外，状态窗口中会出现 **Contact to ECU <ECU name> established.** 消息。
- 如果未能与 ECU 建立联系，则按钮上的标签保持不变，依然为 **Establish contact**。此外，会在状态窗口内显示错误消息（比如，**ECU cannot be initialized (activation failed). Reason: ComCoordinator: 02005: Timeout P2**）。

注意:

在打开系统的情况下, 会采用来自相应的 CAESAR/DIOGENES (CBF 文件以及 GBF 文件) 的通讯参数 (ComParameter), 即, 通常无需为了建立联系而设定通讯参数。

如果并未打开系统, 而且您需要与 ECU 建立通讯, 则存在两种不同的情况:

1、VND 可用:

- 点击 Load VND 按钮。采用 Open file 对话框选择所需的 VND。
- 选择连接。
- 点击 Establish contact 按钮。
- 一旦与 ECU 建立联系, 则按钮上的标签会变更为 End contact。
- 如果无法建立与 ECU 的联系, 那么按钮上的标签仍然保持为 Establish contact。此外, 会在状态窗口内显示错误消息 (比如, ECU cannot be initialized (activation failed). Reason: ComCoordinator: 02005: Timeout P2)。

2、VND 不可用:

- 点击 Select ECU 并且选择某个协议。这是会显示对话框, 其中可以指定协议特定的通讯参数。
- 输入已显示的通讯参数的数值, 然后点击 OK。
- 点击 Establish contact 按钮。
- 与 ECU 建立联系之后, 按钮上的标签会变更为 End contact。
- 如果无法与 ECU 建立联系, 那么按钮上的标签仍然保持为 Establish contact 字样。此外, 状态窗口中会显示错误消息 (比如, ECU cannot be initialized (activation failed). Reason: ComCoordinator: 02005: Timeout P2)。
- 将您的设置存储在 VND 中。

手动指令输入, 输入通讯参数

点击 Comm parameters 按钮打开 Communication parameters 对话框。

本对话框列出了所有有效的 ECU 通讯参数。各个通讯参数的名称显示在 Parameter names 列表中。

如需更改通讯参数的数值, 则应当采取下列操作步骤:

- 在 Parameter name 列表中选择需要更改的通讯参数。
- 输入框中现在显示的是通讯参数的最新数值。
- 点击 Set 按钮。将通讯参数设定为您指定的数值。

Default 按钮将所有通讯参数设定为默认数值。默认数值是首次 (成功) 建立联系时设定的数值, 或者是如果尚未与 ECU 建立联系的情况下, 则为从 GBF 文件读取的数值。

示例:

初始化通讯参数 ME20: KLINE 协议

CP_TRIGADDRESS = 1

CP_RESPSOURCEBYTE = 1

CP_RESPONSEMASTER = 1

CP_REQTARGETBYTE = 1

示例：

初始化通讯参数 CR3: CAN 协议

CP_REQUEST_CANIDENTIFIER = 2016

CP_RESPONSE_CANIDENTIFIER = 2024

CP_BAUDRATE = 500000

手动指令输入，载入 VND

VND (Vediamo-Nachrichten-Datei = Vediamo 消息文件) 用于存储与 ECU 建立联系所需的 ECU 以及通讯参数，以及经常使用的消息以及名称列表。如果存在此类文件，那么可以采用 Load VND 按钮载入该文件。这是会弹出标准的 Open File 对话框，并且会显示以 VND 结尾的所有文件。

选择 VND 之后，会在 Defined request messages 列表中显示其包含的消息的 ID。

如果采用从系统中选中的 ECU 进行工作（载入了 VSB 文件），那么会忽略 DCDIVND 文件中包含各个参数；如果未打开任何系统，那么会将 ECU 以及通讯参数设定为文件中包含的数值。下次打开手动指令输入的时候，会自动打开上次使用的 VND 文件。

保存 VND 文件

点击 Save VND 按钮之后（在标准文件对话框输入文件名之后），会保存包括当前 ECU 以及通讯参数的 VND 文件，以及定义的请求消息。

如果已经更改了消息及/或参数，会在关闭 Manual input entry 对话框的时候提供给用户，以便将更改内容保存在 VND 文件中。

定义的消息

用户可以将需要发送的消息内容以十六进制格式（输入行 Request message）进行输入。用户可以选择输入完整的消息，以及仅输入有效载荷（Enter payload only 选项）。可以在 Message name line 中输入消息的名称（任何文本）。点击 Add to list 按钮将该消息复制到定义的消息列表（仅会在 Defined request messages 列表框中显示名称）。此外，还可以更改定义的消息（通过更新输入）或者从列表中将其清除（点击 Delete from list 按钮）。

此后可以将定义的消息保存在 VND 文件中。

将消息发送给 ECU

可以采用多种方式向 ECU 发送消息。规则如下所示：如果点击了 Send 按钮，则会发送输入行中显示的消息。

输入行中的消息可以采取以下方式进行处理：

- 直接输入
- 从输入行历史中选择
- 或者从定义的消息列表中选择（这种情况下，将选中的消息复制到输入行中）。

Send 按钮时对话框的默认按钮。这意味着，不仅可以直接点击该按钮，还可以点击回车键或者双击定义的消息列表中的某一项。

如果已经输入了完整消息之外的有效载荷字节，程序会采用正确的校验和字节从有效载荷中创建完整消息，并将其发送给 ECU。

已发送消息的事项会显示在 Sent and received messages 窗口中。如果 ECU 响应消息，则也会显示该消息。用户可以选择整个消息或者仅显示有效载荷。

VND 文件

Vediamo 消息文件包括手动指令输入的信息。此类消息是可以编辑的 ASCII 文件。文件的结构大致对应于 INI 文件中的相关内容。可以使用注释行，此类行的开头为“//”。空格忽略不计。由于参数均为 ECU 专用，因此文件仅适用于相同类型的 ECU。

文件包括三个部分：

- ECU 参数
 - 通讯协议
 - ECU 地址
- 初始化通讯参数
 - CAESAR 需要的通讯参数，用于与 ECU 建立联系（比如，CP_TRIGADDRESS）。
- 消息定义。
 - 请求报文及其 ID（比如,读取 ID=3C 00）
 - 行内首先列出的消息 ID，以及“=”与十六进制有效载荷，二者以空格分隔。如果字节包括在括号中，则需要将字节解释为完整的消息，否则将此类消息视为有效载荷。

ECU 参数示例

示例：
[ECU]
Protocol = KW2000P
Address = 01
参数“Pin”为十进制，“Protocol”指定为关键字。“Address”以及其它通讯参数采用十六进制格式输入。

初始化通讯参数示例

示例：
[COMMUNICATION]
CP_REQTARGETBYTE=69
CP_RESPONSEMASTER=69
CP_RESPSOURCEBYTE=69
CP_TRIGADDRESS=69

消息定义的示例

示例：
[MESSAGES]
Read ID = 3C 00
Read error = (81 01 F3 1A 8F)

专家系统

手动指令输入使得用户通过点击“Send/Stop sending”按钮即可将任何请求直接提交给 ECU：本按钮具有两个状态。点击后，该按钮处于按下状态，并且会循环发送消息。再次点击，则按钮的状态会变更为弹起状态，同时会停止发送。“Close”按钮：如果更改了任何消息或者参数，则要求用户将内容保存在 VRS 文件中。

对于载入的系统来说，还可以从树型视图中拖拽之手动指令输入窗口。

请求存储在选定服务的诊断参数化方式之中，然后显示在手动指令输入中。存储的服务修饰符输入到 Message name 输入框中，请求消息采用十六进制格式（如果可以确定的话），然后输入到 Request message 输入框中。

如果在 Defined request messages 列表框中存储服务，则还可以自动添加至预先定义的消息列表中。

用户可以在提交请求之前对字节进行变更或者操作，因此可以对 ECU 进行特定的测试。

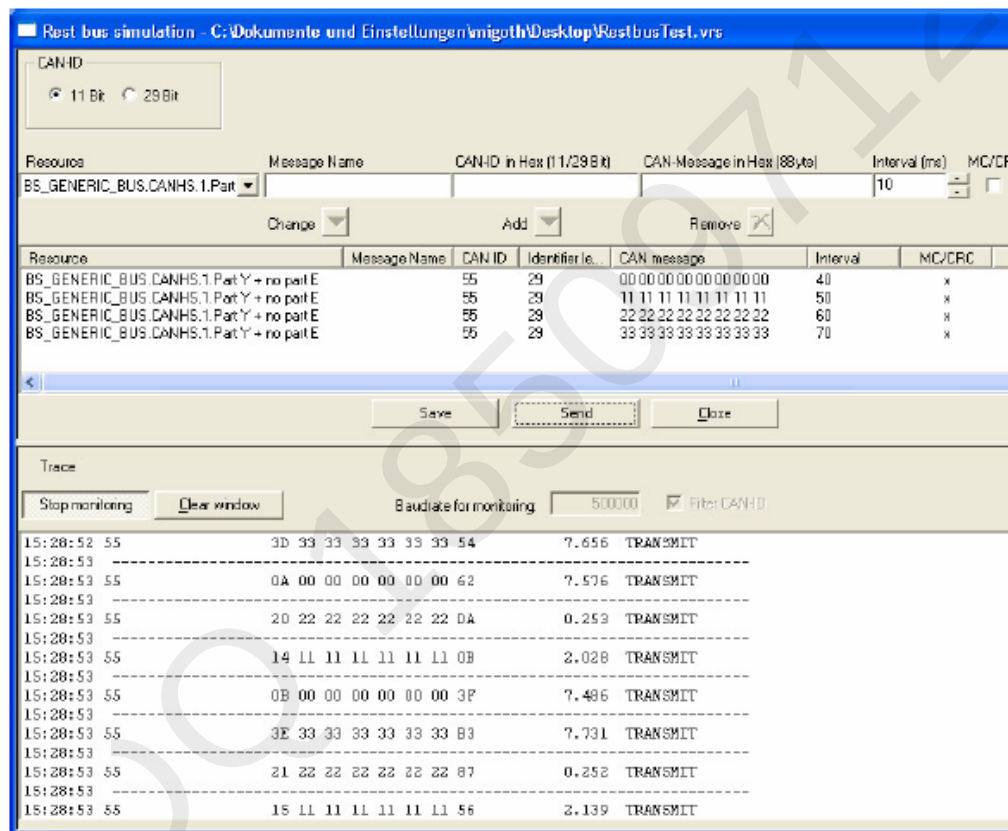
此外，还可以采用复制以及粘贴方式（比如，从编辑器中）在 Request message 输入框中输入任何十六进制字节序列。

2.4.4.20 CAN 总线仿真

本功能允许对并未附加到网络中的 ECU 的 CAN 消息进行仿真。这种情况是针对其它 ECU 发出的特殊消息而对 ECU 功能进行测试所需的，同时无需完整的网络模型。

采用 ECU / CAN Bus Simulation 菜单项激活本功能，这是会弹出窗口，在其中可以设定下列参数：

- CAN-ID（十六进制）
- 需要发送的消息（十六进制）- 11 或者 29 个比特位长度
- 以毫秒为单位的时间间隔
- 资源
- 消息计数器
- CRC 校验和



参数“CAN-ID”：

将 CAN ID 的长度设定为 11 或者 29 比特位。默认值为 11 比特位。

参数“Ressource”：

列表框中包含了当前可用的 CAN 资源：

- HSCAN: BS_GENERIC_BUS.CANH5
- LSCAN: BS_GENERIC_BUS.CANLS

第一个为默认使用的数值。

参数“Message Name”：

用户可以定义的消息名称。

参数“CAN-ID in Hex”：

在此处输入十六进制的 CAN-ID。

参数 “CAN-Message in Hex”:

在此处输入十六进制 CAN 消息。长度限定为 8 个字节（64 比特位有效负荷）。

参数 “Interval”:

输入消息应当重复的间隔（毫秒为单位）。在点击 “Send” 按钮之前，将最初的 0 数值更改为正数。激活发送的同时不可能更改间隔。

单选框以及 “MC/CRC and Options MC/CRC” 按钮:

本特性可以用于生成消息计数器及/或相关 CAN 消息的校验和。“option” 按钮显示了对话框，其中可以配置各种必要的参数：对于 CRC 来说，必须计算指定 CRC 所需的区域，以及 CAN 消息中存储 CRC 的字节位置。对于消息计数器来说，必须输入比特位位置以及比特位长度。已经制定的参数则显示在相关的列表项中。如果忽略了参数/保持为空，则不会激活相关的功能（MC bzw, CRC）。

Message Name: TEST

CAN-Message: 11 11 11 11 11 11 11 11

Message Counter (MC), values in bits

Note: the following values belong to the 8 bytes long CAN message. START: bit position of MC in the CAN message. LENGTH: length of the MC in the CAN message, beginning at START. The counter must not cross byte boundaries.

START: 0 bit (0...63)

LENGTH: 8 bit (1...8)

Checksum (CRC), values in bytes

Note: START: position of the first byte of the area on which the CRC shall be calculated. LENGTH: length of the area in bytes. TARGET: byte position, where the CRC shall be stored in the CAN message.

START: 0 in byte (0...7)

LENGTH: 8 in byte (1...8)

TARGET: 6 in byte (0...7)

OK Cancel

“change” 按钮:

使用 “change” 按钮，可以更改消息列表中需要发送的消息项。

“add” 按钮:

使用本按钮将编辑的消息项添加至需要发送的消息列表中。根据使用的硬件的不同，最多可配置并且仿真 10 条独立的消息。

“remove” 按钮:

使用本按钮或者 “delete” 按键从消息列表中移除选中的消息项。

“Send / Stop sending” 按钮:

本按钮具有两种状态。点击之后，会变为按下状态，这时会循环发送消息。再次点击则会变更为弹起状态，并且停止发送消息。

“Close” 按钮:

如果更改了任何消息或者参数，则要求用户将相关内容保存到 VRS 文件中。

“Save” 按钮:

全部配置均可保存在 *.vrs（Vediamo Rest bus Simulation）中。这时会弹出 “file save” 对话框要求选择路径以及文件名称。默认文件名为 ...\\VediamoDaten\\yyyy-mm-tt_[Corrent/last system].vrs。如需使用本文件，请使用菜单项 “ECU / Load CAN bus simulation”。

跟踪分析：

设置或者载入配置并且开始发送之后，跟踪窗口将显示数据量以及单个字节与时间戳。无法保存该窗口中的内容。

使用切换“**Start/Stop monitoring**”按钮激活/去活监控功能。

按钮“**Reset window**”清空监控窗口中的内容。

为了更便于阅读，请使用“**filter CAN-ID**”单选按钮以便仅显示仿真的 ECU 消息。

注意：

CAN 总线仿真属于标志通讯功能的扩展功能。因此，并非所有通讯硬件均支持本特性。特别需要注意，eCOM (PDUAPI) 并不支持 CAN 总线仿真。

2Q 1850971v

2.4.4.21 快照文件存储

点击 **Save** 按钮或者 **CTRL+F2** 组合键可以将错误、驱动器以及测量窗口的快照存储在文件中。点击 **Save** 按钮之后，会弹出对话框，其中用户可以输入数据、备注以及快照文件的名称。对话框中会显示推荐的文件名称。可以接受或者更改该名称。（按钮 ... 输入框 Snapshot 文件旁边）。

快照文件的默认格式为 **HTML**。快照文件可以采用 **HTML** 或者文本格式进行日志记载。主要根据文件扩展名进行选择。“**htm**”或者“**html**”结尾表示采用 **HTML** 格式。其它所有情况则表示保存为纯文本文件。

需要保存的所有快照均采用快照文件中的 **ID** 块项。快照可以附加到已经存在的文件中。最后一次使用的快照文件会显示在弹出的对话框中。点击 **OK** 按钮即可将快照写入文件中。

首次存储的时候会保存下列信息：

- 版本，比如 **Ecouste 05.00.00** 版
- 系统名称
- **ID** 块
- **CBF** 版本
- 用户数据（需要由用户填写）
- 备注

话路期间联系的系统中所有可用 **ECU** 的 **ID** 块均予以保存。如果选定目标文件不存在任何通讯，则会在建立通讯的时候保存 **ID** 块。如果终止通讯而且重新建立通讯，则会在通讯建立期间检查，查看是否 **ECU** 的 **ID** 块已经包括在文件中，如果是，则不会保存该 **ID** 块，否则，会将该 **ID** 块写入文件中（**ECU** 交换）。

如果已经设置了目标路径，则采用 **CTRL+F2** 即可将错误、驱动器以及测量窗口的快照存储在文件中。这种情况下不可输入备注或者用户数据。

2.4.4.22 ECU 通讯仿真

可以在 Vediamo 中对与 ECU 的通讯进行仿真，而无需建立与 ECU 真正的物理连接。
仿真模式

服务器的仿真模式具有三种状态：

状态	模式	说明
正常	0	正常状态，与物理存在的 ECU 进行通讯，默认设置。
仿真	1	仿真的 ECU 通讯。
记录仿真数据	2	记录与物理存在的 ECU 的通讯，以便在仿真模式中进行回放。

通过 Vediamo.ini 文件中的某一项对状态进行控制：

[Server]

Simulation=<Mode>

Vediamo.ini 项在服务器启动之后，或者是在后续重新初始化导致服务器复位之后生效。在 Vediamo.ini 中可以使用 INI 编辑器对该项进行永久性更改。您还可以采用 Start Center 中的仿真选项来启动 Ecoute。但是请记住，仿真模式仅涉及服务器本身，而不涉及任何客户端。因此，不可能在另外一台客户端已经在没有仿真而启动的情况下，与另外一台客户端采用仿真模式一同工作。

在仿真模式中，无需任何 CAESAR 硬件即可使用 Vediamo。采用仿真模式启动服务器以及检查 CAESAR 硬件 ID 的时候，无需任何密码。

仿真模式中不会调用 CAESAR 通讯 API。因此不可能出现 API-I 调用（Ecoute：手动指令输入）、接收监控数据（跟踪）以及编辑 COM 参数。

系统说明中介绍的服务过滤器在仿真模式中不会进行任何更改即可发挥作用。

记录仿真数据

针对常规诊断过程中记录 ECU 表现而实施了某种机制，并且在仿真过程中将其回放。

在服务器启动或者重新初始化的时候，采用 Vediamo.ini 项[Server] Simulation=2 来激活 SIM 数据记录。

记录的数据存储在系统文件夹的适当 SIM 文件中。

这种情况下，会记录下列数据：

- ECU
- 最后一次变体
- 服务
- 修饰符，结果

如果多次执行某种服务，则会记录最多可达 10 次最后出现的序列记录。

- 错误

这种情况下会记录最后一次“读取错误”流程中的出现的错误的 P 代码及其环境数据（如果激活环境数据情况下读取错误）。

SIM 数据记录期间会生成尚未提供的 SIM 文件。已经存在的文件则予以更新。当前系统关闭的时候，记录的数据会被写入最近受影响的 SIM 文件中。无法规定记录模式是否会对 Vediamo 的性能产生不利影响。

仿真数据

仿真数据的存储方式便于针对下列适用情况编辑文本文件：

- 系统文件夹中针对每个 ECU 均存在一个 SIM 文件。如果动包括多个同名的 ECU，则会此类 ECU 均会采用同一个文件。文件的名称取自 ECU 的修饰符以及扩展名 “.Sim” 比如，“ME28.Sim”。从客户端无法选择或者更改文件名或者文件。仅可直接在窗口中通过复制、切换、编辑等方式进行变更。服务器通过每个 ECU: Init () 的方式读取文件。
- SIM 文件中并未包括的所有必要的的数据，比如名称、说明、规格以及数值限值等，均尽可能从可用参数化中获取。

ECU 设置

仿真了下列功能：

- ECU: Init () 以及 Exit ()
- 使用 SIM 文件中输入的变体。如果没有可用的 SIM 文件，则使用基本变体的默认数值。GetIDBlock () 功能仅返回 ID 块。点击 ECU 中系统树形控件中 ECU 项，即可显示 ECU 相关功能菜单。仿真模式中，该项扩展包括 Select variant 功能。该菜单项尽在仿真模式中显示。在仿真模式中执行该功能会弹出对话框，其中包括选中了需要显示的所有 ECU 参数化变体。选中某个变体之后，会在 Vediamo 服务器内部激活，以便进行仿真。系统树形控件也随之更新。

SIM 文件中的各项（示例）：

示例：

[General]

Version=1

SIM 文件版本

;ECU=CRV

备注，仅供参考。SIM 文件名称决定了具体的 ECU。

Variant=C47

需要仿真的变体。默认数值：空=基本变体

Error="TimeOut P2"

默认值：空。如果在此输入某个数值，如果在仿真中建立联系失败，则输入的数值也返回错误。

ECUInitDelay=100

仿真模式中建立联系所需的延迟，单位为毫秒。默认数值=0。

ReadAllErrorsDelay = 500

仿真的“读取错误”操作中的延迟，单位为毫秒。默认数值=200。

ServiceExecutionDelay = 0

仿真的“执行服务”操作中的延迟，单位为毫秒。默认数值=200。

诊断服务器的结果

执行某项服务的时候，SIM 文件中规定的数值返回为结果。如果 SIM 文件中并未指定任何数值，则返回“？”。还可以在 SIM 文件中指定采用逗号分隔的数值列表。这种情况下，会按照一定次序返回数值，第一个数值用于首次调用，第二个数值用于二次调用，以此类推。一旦达到列表中的最后一个数值，则序列调用再次返回第一个数值。

如果在 Ecouste 对话框中输入了某项准备，那么会检查其有效性，但是不会影响执行的结果。

结果的规格可以从参数化中获取。

不会对设定数值与可能可用的 OutputRef 之间的关系进行仿真。SIM 文件中的相应项适用于对以下情况进行的仿真

- 一般服务
- 测量结果
- 驱动器以及调整状态
- 功能

所有此类 Vediamo 类别的各项在 SIM 文件中均是相同的。

各项：针对需要仿真的每种服务，在 SIM 文件中提供一项。

示例：

[DT_01_Oeltemp]

服务项。

Value=85.0

单一数值。

...

[_S_DT_01_Bordspannung]

下一个服务项。

Value=12.4,12.3,12.8,13.4

各个数值的列表。

...

读取错误

读取错误返回 **SIM** 文件中规定的 **P** 代码，并且在适当的情况下，返回环境数据列表以及 **SIM** 文件中输入的数值。如果没有可用的 **SIM** 文件，或者 **SIM** 文件中不包含任何错误信息，则不会返回任何错误以及任何环境数据。

示例：

[P0110]

错误项。(P 代码)

EESActive=1

默认数值=0。

EESStored=1

默认数值=0。

EESMIL=0

默认数值=0。

;All entries with a semicolon are comments

用于结构化以及解释说明的相关备注。

ENV_Drehzahl =0.0

环境数据-修饰符以及数值。

ENV_MotorTemp=-40.0

环境数据-修饰符以及数值。

...

[_E_P1605]

下一个错误项...

...

变体编码

仿真采用 **CAESAR API** 提供的“离线变体编码”。这种情况下会显示编码服务以及片段。仅可在 **CAESAR API** 框架内进行片段数值的仿真读取以及编码。必须提供适当的参数化。此外，还必须连接具有相应的访问级别的 **CAESAR** 硬件。

闪存处理

无法对闪存处理过程进行仿真。

2.4.4.23 夹子 15 处理方式

由于在 OBD 套接字中将夹子 15 省略为针，因此作为信号，在诊断 CAN 中提供了点火开关状态的替代方式。CAESAR 可以评估该信号，但是必须进行相应的配置。详情参见夹子 15 省略问题的相关 CAESAR 文档记载内容。

在 Vediamo 中，已经针对系统说明中的系统进行了此类配置，但是仍然可以提供与标准状况无关的系统说明。

某个 INI 文件中包了夹子 15 参数的配置，该参数与具体型号有关（文件名称：“Ignition.ini”）。INI 位于程序文件夹中。

该文件包括与型号有关的下列数据：

- 通过“硬件针”或者“取自 CAN”，实现识别“自动化”
文件中的可选项采用下列方式进行读取：AUTOMATIC、HARDWARE、CAN
 - 文件名夹子 15 CBF
 - 夹子 15 识别专用的 ECU 名称
 - 变体（取决于型号）
 - 点火读取服务
 - CAN 唤醒服务
- 文件内容示例：

[Models]

NumberOfModels=3

Model1=BR211

Model2=BR220

Model3=BR203

[DEFAULT]

IgnitionMeasurement=HARDWARE

CBFFFileName=

ECU=

Variant=

ReadIgnitionService=

CanWakeupService=

[BR211]

IgnitionMeasurement=AUTOMATIC

CBFFFileName=Clamp15.cbf

ECU=CLAMP15

Variant=1_Bit_Auswertung

ReadIgnitionService=

CanWakeupService=

[BR220]

IgnitionMeasurement=CAN

CBFFFileName=Clamp15.cbf

ECU=CLAMP15

Variant=2_Bit_Auswertung

ReadIgnitionService=

CanWakeupService=

[BR203]

IgnitionMeasurement=HARDWARE

CBFFFileName=

ECU=

Variant=

ReadIgnitionService=

CanWakeupService=

“ReadIgnitionService”以及“CanWakeupService”均仅供参考 – 不再继续使用。

在 Ecoute 中，在菜单项“extras”中存在一个菜单项“clamp 15 handling”，该菜单项用于调整点火识别的行为，具体取决于相关的型号。如果选中了菜单项，则会弹出对话框，其中可以选择型号。点击“OK”，则会依照选中的型号设定配置。最后一次设定的配置/型号会存储在 Vediamo.ini 文件中，并且会在下次启动 Vediamo 的时候载入并且设定。

Vediamo.ini 中的键值：

[Servers]

...

IgnitionMeasurement=BR203

如果 VSB 包括一个有效的设置（所有可用的信息），则 VSB 配置会覆盖当前选定的配置。可以采用 Ecoute/其它客户端/测试序列更新各种配置的设置。如果针对 Ecoute 中的每个对话框设置型号，而且不存在规定的夹子 15 CBF（=并处于 CAESAR 当前使用的.CBF 列表中），则会提供警告；但是不会设置配置。其它客户端以及测试序列在设定设置的时候，均会接受此类信息作为返回数值。

2.4.5 Ecouté 菜单

System 菜单

Select

使用本菜单项载入需要一共使用的系统说明。已经打开的系统会在此期间予以关闭。

Close

采用 **System / Close** 菜单项关闭系统。显示 ECU 相关数据（系统、错误、测量、跟踪）的所有窗口均会在此期间予以关闭。

Open OBD2

打开 OBD2 连接。采用本协议的情况下，可以读取诊断数据，而无需文件相关的特殊 ECU。会打开特殊 OBD2 窗口。

Close OBD2

关闭 OBD2 连接以及 OBD2 窗口。

Open session

采用 **Open session** 菜单项打开话路文件。可以通过文件选择对话框完成本操作，该对话框会显示以 .vsc 结尾的所有文件。

通过本对话框，可以选择需要打开的话路。在标题行中，显示激活的话路文件。

Save session

采用 **Save session** 菜单项将当前打开的 **Ecouté** 话路中变化保存至激活话路文件中。

可以采用特定话路启动 **Ecouté**。话路文件的完整路径必须通过指令行参数传送至程序中。或者，可以采用上次存储的话路自动启动程序。因此，必须在选项对话框中激活 **load last session** 选项。

Save session as

Save session as 菜单项允许采用不同名称存储 **Ecouté** 话路。采用文件存储对话框进行此类操作。可以指定文件名称以及次数位置。

Close session

Close session 菜单项关闭话路文件。此类操作之后，选定系统以及所有打开窗口均会予以关闭。依照 INI 文件来恢复主窗口。

List of last used files (MRU list)

最后打开的四个系统或者话路的名称会显示在 **System** 菜单中。点击此类文件即可将其打开。

Exit Program

该项会关闭 **Ecouté**。如果没有其它激活的 **Vediamo** 客户端，则会自动结束 **DiagServer**。

ECU 菜单

Init contact

Init contact 启动过程以便与 ECU 建立联系。还可以采用 F3 功能键，双击选择窗口中的 ECU，或者点击状态行中的 ECU 按钮来完成此项功能。

Exit contact

结束与 ECU 的通讯。替代方式：F4 按键，双击选择窗口，状态行中的 ECU 按钮。

Read ID block

本指令（或者采用的 ATL+I）从 ECU 中读取 ID 数据。对于多个 ECU 来说，必须首先选定一个 ECU。为了读取 ID 块，必须建立与 ECU 的联系。

Communication Parameters

有经验的用户可以采用本菜单项更改通讯参数。

Manual Command Input

本指令激活窗口，以便在低级水平上与 ECU 进行通讯（CAESAR-API-I）。字节级别消息可以在此传输，而且可以对回复进行分析。

Properties

本菜单项会打开包括 ECU 属性的窗口。

CAN bus simulation

对 CAN 消息进行仿真。

Load CAN bus simulation

载入预先设置的 CAN 消息，以便进行仿真。

Channel Monitoring

使用的本功能可以显示针对现有任何 CAESAR 连接的跟踪窗口。

Error 菜单

New error window

即使窗口已经打开的情况下，也会打开一个新的错误窗口。采用当前的错误消息填充该窗口。

Read errors

打开新的错误窗口，或者更新已经打开的错误窗口。

Read permanent errors

打开新的错误窗口，或者更新已经打开的错误窗口，其中包括了永久性错误。

Clear errors

清空 ECU 中的错误消息。如果在下次读取错误的时候会显示所有变化，具体取决于相关的 ECU。

Read, save and clear

为了简化需要经常重复的动作，本功能可以执行下列三种不同功能：

- 读取错误
- 在快照文件中日志记载读取错误
- 删除 ECU 中的错误

Quick test

执行快速测试。读取或者清空当前建立通讯的所有 ECU 的错误存储器（比如，用于整车）。

Read errors by Status

读取所有支持的错误代码（KW2000: “REQUEST SUPPORTED 2 BYTE HEX DTC AND STATUS (\$03) ”）。协议特定。仅针对 KW2000 以及 UDS 有意义。

Service Group 菜单

New service group

打开新的服务分组窗口。可以采用服务进行填充，并且存储为服务分组文件（VSG）。

Open service group

打开存储的服务分组。

Open/Analyze Recorded Series

载入之前记录的一系列测量结果（VSR 文件）。

Coding 菜单

Variant coding

打开窗口以便读取以及设置 ECU 变体。如果在系统说明文件（VSB）中正确输入了具体信息，则会首先执行各种服务，此类服务令 ECU 进入正确的模式（前置条件）。如果并未出现此类情况，则还可以从选择窗口中手动启动各种服务。

Flash

打开窗口以便采用闪存处理方式将特定固件放入 ECU 中。

Services 菜单

Execute Routine

可以从多个分组中的一个分组或者 Java 程序中选择并且启动某种服务。如果您必须在没有系统窗口的情况下开展工作，那么此类选项是非常有用的。

传统意义上，本菜单还可以用于在标准对象的帮助下插入定制的菜单项。

Extras 菜单

Find

还可以采用键盘快捷方式 CTRL+F 执行“Find”功能。

对于用户来说，可以采用多种方式在 Vediamo 相关数据中查找某个文本：

- 在当前系统中
- 在 Ecout 窗口中
- 在文件中

“在当前系统中搜索”、“在 Ecout 窗口中搜索”或者“搜索文件”分别属于两个不同的窗口，用户可以使用 tab 按键进行切换。

Searching in the current system

需要查找的文本专用的输入框：

输入框与文件中搜索专用的输入框相同。输入的关键词会出现在两个窗口中。输入框提供了包括之前输入关键词的历史记录。可以从剪贴板中粘贴关键词。

Search in current window

如果选择了“Search in current window”，则会选择当前激活窗口中下一个找到的结果，而且显示该结果。会从开始位置或者从当前选中的位置搜索相关的窗口。

Search in tree:

如果选中了“Search in tree”，则会立即搜索当前载入的所有系统，而且将命中结果添加至结果列表中。结果列表包括各种服务的修饰符/名称，此类服务包括需要的关键词。如果用户点击结果列表中的某个服务，则会选择系统树形控件中的相关项。可以采用拖拽的方式将结果列表中的各项服务添加至测量窗口中。

Searching in files

查找文本专用的输入框：

输入框与当前系统中搜索专用的输入框相同。输入的关键词会出现在两个窗口中。输入框提供了包括之前输入关键词的历史记录。可以从剪贴板中粘贴关键词。

文件夹专用输入框：

指定文件夹，后者包括需要搜索的文件。默认为当前系统路径。

搜索文件：

解析具有选中的扩展名的文件。

结果列表：

将包含指定关键词的文件添加至结果列表中。列表中还会显示文件路径名称。如果用户点击结果列表中的某个文件项，则会打开并且显示相关文件。

Options

打开窗口，以便编辑 Ecout 以及服务器的 Ecout 相关选项。

Flashdata Administration

打开对话框以便处理闪存文件（详情参见闪存对话框）。

Font

需要在测量、驱动器以及错误窗口中使用的字体，可以采用本菜单项进行选择。

Clear status window

本指令会清空状态窗口中的所有项（输出窗口）。

Server Reset

重新初始化 **DiagServer**。如果需要采用更改的设置或者更新的参数化继续工作，那么必须这样做。可以将当前 **Ecoute** 状态（打开系统、打开的窗口）缓存，并且在复位之后进行恢复。某些情况下（更改某些设置，后者需要进行复位），会询问用户是否应当重启服务器。

Clamp 15 Handling

打开窗口以便选择某个型号用于夹子 15 处理。

Show trace

打开跟踪窗口。

Macro

可以在此存储一系列 **Ecoute** 指令，或者可以回放存储的系列指令。

Config toolbar

本特性允许工具条适应特定需求。在自有窗口中进行配置：

左侧，列出了功能可用的工具条按钮，右侧，当前指定的工具条按钮。

采用“add”按钮，左侧选中的某项可以添加至工具条中，同样地也可以在右侧列出。

类似地，采用“Remove”按钮的情况下，右侧选中的某项可以从工具条中予以移除。

“Up”以及“Down”按钮指向工具条的当前内容，而且位于右侧列表中。凭借此类帮助，可以将列表中当前选中的项向上或者向下移动，进而确定按钮出现在工具条中的次序。

在任何位置采用“Separator”项，均可插入分隔线，比如，从二者之间抽取单个功能组。

通过拖拽的方式，工具条可以停靠在主窗口的顶部/底部/左侧/右侧，并且可以拖拽至指定位置。

Window 菜单

Cascade, Tile Horizontally, Tile Vertically

此类指令安排了打开的各个窗口。

Status

打开或者关闭状态窗口（输出窗口）。

System window

打开或者关闭选择窗口（系统窗口）。

Routine manager

打开或者关闭窗口以便控制 Java 程序。

Macro Window

打开或者关闭窗口以便检测回放宏。

Update current window

采用本菜单项或者 F5 功能键更新激活错误、驱动器或者测量窗口的内容。CTRL+F5 组合键立即更新所有窗口的内容。

Show toolbar

显示或者隐藏工具条。

“?” 菜单

Help topics

打开在线帮助。

About Ecoute

显示 Ecoute 的“业务卡”。

Keyboard Operation

显示在线帮助以及所有键盘指令。

Additional information

对于 Ecoute 而言，用户可以提供自有的额外的支持信息，具体采用的是.chm 以及.PDF 文件格式。载入新系统的时候，会显示下列文件：

- 位于文件夹..\Data\VediamoDaten\中的所有.chm- 以及 .PDF 文件均列于“?” 菜单中。
- 位于文件夹..\Data\VediamoDaten\[current system]中的所有.chm- 以及 .PDF 文件均列于“?” 菜单中。
- 此外，所有此类文件还位于系统树形控件的新文件夹中。

如果选择菜单上的文件，或者双击系统树形控件中的该项，则会打开文件并且显示其中内容。

2.4.6 键盘操作

在 windows 以及鼠标大行其道的时代，程序不一定可以从键盘进行操作。但是，为了方便起见，比如，在测试驱动的过程中，还是可以不用鼠标来执行所有重要的 Ecouté 功能。例外情况是调整窗口大小以及位置，以及其它很少需要使用的功能，而且此类功能进出现在测试驱动的准备以及配置阶段。包括读取错误、显示一组选中测量结果等等在内的此类活动，均可采用快捷键方式启动。规定的热键主要基于早期的 DOS 诊断程序 MODI 以及 ISOSCAN。

下表包了所有 Ecouté 键盘指令的完整列表。

按键	动作
F1	在线帮助
F3	与 ECU 建立联系
F4	终止与 ECU 的联系
F5	更新激活的窗口
F6	如果不存在已经打开了错误窗口，则会打开新的错误窗口。如果已经打开了错误窗口，则会更新其中的内容。
F7	读取所有测量结果
F8	读取所有驱动器
F9	执行上次的宏
F10	启动快速检测
F11	打开 OBD2 窗口
F12	手动指令输入
Alt+A	选择系统
Alt+B	删除状态消息
Alt+D	激活服务菜单
Alt+E	读取一次（比如，测量结果）
Alt+F	闪存处理 ECU
Alt+G	显示树形控件中当前服务的属性
Alt+I	读取 ID 块
Alt+K	激活编码菜单
Alt+J	ECU 属性
Alt+L	清空所有错误
Alt+O	打开话路
Alt+P	保存快照（比如，错误窗口）
Alt+Q	显示跟踪
Alt+S	激活系统菜单
Alt+T	激活 ECU 菜单
Alt+V	执行变体编码
Alt+Z	关闭系统
Alt+F	激活错误菜单
Alt+M	激活测量菜单
Alt+R	激活驱动器菜单
Alt+N	激活窗口菜单
Alt-F4	退出 Ecouté
Alt+F6	读取永久性错误

Alt+F7	打开服务分组
Alt+F8	打开驱动器分组
Alt+F9	打开宏选择列表
Ctrl+F	查找功能
Ctrl+F5	更新所有窗口
Ctrl+F6	读取、保存以及清空错误
Ctrl+F4	关闭激活服务分组、驱动器分组、错误窗口
Ctrl+F7	测量：新分组
Ctrl+F8	驱动器：新分组
Ctrl+F9	记录宏
Ctrl+F10	设定标准系统路径
Ctrl+TAB	切换不同窗口
Schift+F6	打开新的错误窗口
ESC	关闭激活的窗口
TAB	在激活的窗口内部跳转

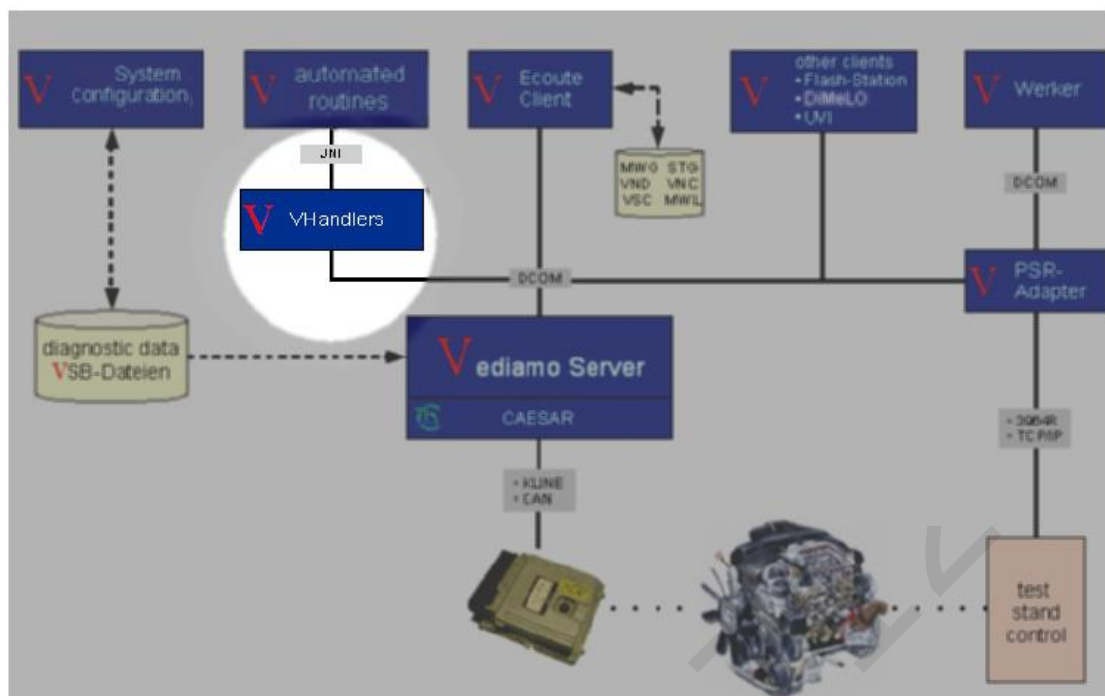
选择窗口（系统窗口）

按键	动作
向右箭头	展开项
向左箭头	收起项
向上箭头	在选择窗口中向上移动
向下箭头	在选择窗口中向下移动
回车键	开始动作： ECU：建立或者终止联系 服务、Java 等：执行 文件夹：展开/收起

图形测量窗口

按键	动作
回车键	启动/停止记录数值
空格键	启动/暂停循环读取
Del	从存储器中擦除记录的数值
光标指针	选择数值指向需要显示的区域（如果已经读取了任何测量数值的话）

2.5 Java 句柄功能



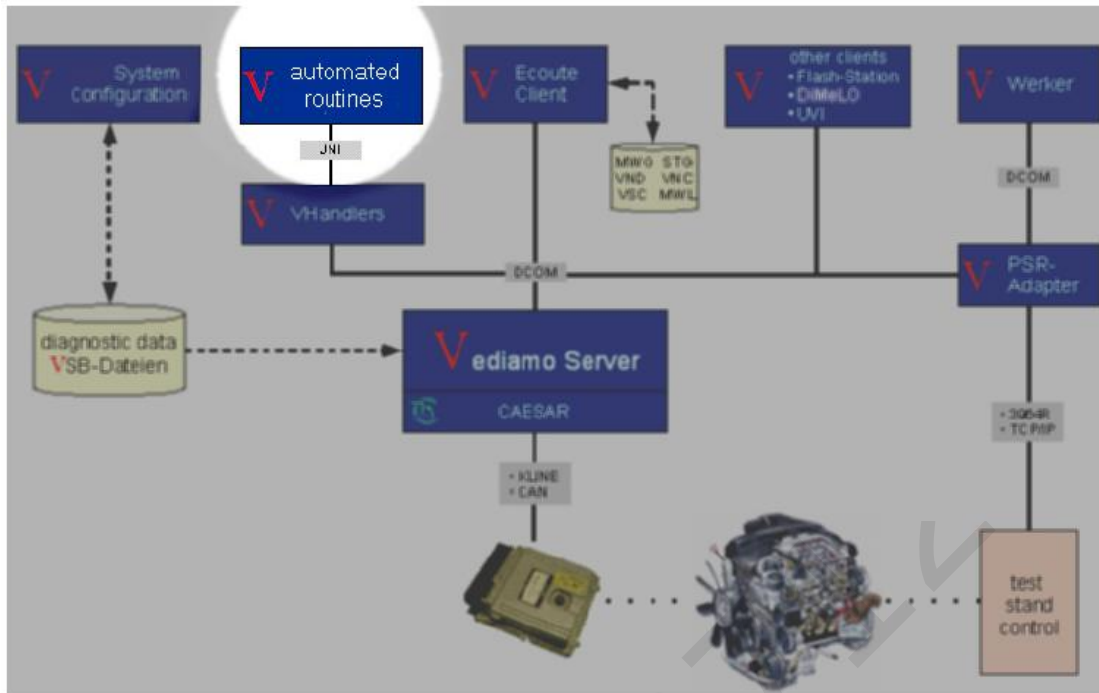
在 Vediamo 中，可以在诊断话路中执行 Java 程序。Java 程序属于用户可以自行写入/编辑的 Java 程序，并且可以从 Vediamo 诊断服务器调用此类程序。因此，Vediamo 提供了多种 Java 类实现的接口。

2.5.1 Vediamo Java 接口

VHandlers Java 接口文档（[点击查看详情...](#)）介绍了提供的所有 Java 类，这些类主要用于访问 DiagServer 的功能。这些类可以用于采用 Java 语言自行开发的应用程序。

根据使用的 Java 解释器，与 Vediamo 自有 Java 句柄类连接方面可能存在一些冲突：某些 Java 开发环境/解释器需要命名包中的 Vediamo 自有的 Java 句柄类，比如，“JVHandlers”。因此提供了两个版本的 Vediamo 中的 Java 句柄类，一个位于未命名包中，另外一个位于命名为“JVHandlers”包中。两个版本的功能绝对相同。VHandlers.dll 模块同时支持未命名包以及 JVHandlers 包。“未命名”包以及“JVHandlers”包中的句柄类均可安装在..\Java 文件夹下的单独目录中。

2.6 Java 程序 (Java 程序)



2.6.1 引言

编程功能需要增加经常重复使用的功能以及动作的效率。在早期诊断系统 (MODI、ISOSCAN) 中, 采用自有 Pascal 语言编写的功能链证明具有一定的可靠性。很多其它程序均可以记录以及回放宏, 后者在极端情况下采用自有编程语言进行编译 (比如, MS 办公应用程序中的 Visual Basic)。

在 Vediamo 中, 可以对定制的 Java 程序进行编程。Java 的优势显而易见:

- Java 是一种标准的广泛使用的编程语言, 各种语言的文档均非常丰富。无需学习 Vediamo 特定的语言。
- 可以免费获取所有主流操作系统的 Java 程序。无需针对 Vediamo 开发专用解释器, 而且也无需大量的测试, 因为 Sun Microsystems 公司的 Java 解释器已经直接整合进了 Vediamo 程序中。
- Java 不仅可以完成 Vediamo 系统功能, 还可以提供各种可用的库。因此可以针对任何已知的复杂程度、窗口、菜单、图形等编写独立应用程序, 而且可以实现 Vediamo DiagServer 的最简便可访问的诊断功能。

2.6.2 从 Ecoute 或者其它客户端执行 Java 程序

Java 程序最初用作 ECU 参数化的扩展, 并且作为一种额外的 DiagJobs。这样就可以在每次建立联系的时候执行一系列特定诊断服务, 或者在启动的时候执行特定的动作 (驱动激活、变体编码...)。为此, 可以采用系统配置将 Java 程序添加至系统说明中。载入系统之后, Ecoute 选择窗口中会显示以及访问此类内容。程序序列通常非常简单; 无需自身提供任何接口, 并且可以假定已经提供了某种 ECU, 并且在启动的时候已经建立了联系。

选择并且承认 (鼠标双击, 点击键盘上的回车键) 选择窗口中的相关项, 即可从 Ecoute 中启动此类程序。Ecoute 输出窗口中会显示程序的所有输出结果。如果程序被声明为初始化程序 (初始化序列), 则可以即使执行此类程序, 而无需用户的参与。

采用“600”指令可以从 PSR 客户端中执行系统说明中列出的程序。为了与测试工作台控制器建立联系，Vediamo-Java 接口拥有 JVPSR 类，后者可以向 PSR 发送消息。

2.6.3 作为独立客户端的 Java 程序

凭借 Java 以及可用库可以生成完整的应用程序。此类应用程序无需与特定 ECU 链接。此类应用程序可以执行、选择系统、联系 ECU、接收与处理数据，而且独立于所有其它 Vediamo 客户端。

此类 Java 客户端通常生成为 JAR 文件。可以采用以下方式执行此类文件。

常规：

```
javaw -classpath <JavaKlassenPfad_1>;<JavaKlassenPfad_2>;...; <routine name>
```

示例：

```
"C:\Program Files\Vediamo\JRE\bin\javaw" -classpath "C:\ProgramData\Vediamo\VediamoDaten\AllgemeineScripte";"C:\Programme\Vediamo\Java"; Classname.jar
```

2.6.4 示例：程序、编译以及执行简单的程序

首次尝试采用 Vediamo-Java 接口进行编程的时候，请使用附件中的示例文件。您需要从 Sun Microsystems 获取免费的 Java 开发套件（当前版本为 J2SE 5.0），以便编写以及编译 Java 程序。或者可以使用诸如 Borland 公司的 JBuilder 等综合开发环境。本示例主要适用于 Java 初学者，而且假定安装了 J2SE 或者更早的版本。

如何实现第一个 Vediamo 专用的 Java 客户端：

- 将示例文件中的内容存储为纯文本文件，文件名称为 **Example.java**。
确保采取一致的方式使用大写以及小写字母：Java 文件名称必须与包含的类名称相同（本示例中为“Example”）。
- 在定义部分，更改 ECU 的修饰符（ID）以及服务（您希望与之通讯的）。使用 **Ecoute** 来建立与 ECU 的首次联系，并且记下您需要访问的 ECU 以及服务修饰符。必须将 **Ecoute** 中的显示内容（Extras / Options / General / Display）发送给 **Qualifier**（而非 **Name**）。
- 将源代码文件编译为二进制代码。

编译命令行为：

```
javac -classpath <Path to the Vediamo handler classes>JVHandlersPackage.jar <Path to Java source code> <Source code Name>.java
```

在我们的示例中，如果您的文件位于 C:\JavaTest\Quellen 文件夹中：

```
javac -classpath "C:\Programme\Vediamo\Java\JVHandlersPackage.jar" C:\JavaTest\Quellen\Example.java
```

系统变量 **PATH** 必须包含 Java 程序的路径（这种情况下指向 Java 编译器 **javac.exe**）。

或者，在命令行中输入完整的路径：

```
C:\Program Files\Java\jdk1.4.2_07\bin\javac -classpath  
"C:\Programme\Vediamo\Java\JVHandlersPackage.jar" C:\JavaTest\Quellen\Example.java
```

编译的文件拥有相同的名称以及扩展名.class: **Example.class**。为了方便起见，本文件应当位于系统的子文件夹中，以便确保程序的有效性。如果命名了系统文件，比如 **CR4test.VSB**，则应当将 **class** 文件复制到 **..\VediamoDaten\CR4test** 文件夹中。

- 将程序与正确的系统相连（其中包括了需要访问的 ECU）。
使用系统配置从事此类操作。打开系统，找到 **routines** 项，鼠标右键点击该项。然后可以查找并且选择刚刚生成的 **class** 文件。还可以添加必要的说明，或者将说明字段留空。
- 启动 **Ecoute**
- 载入命名的系统
- 在系统窗口查找新的程序，双击执行该程序。

您会在“routine manager”窗口中看到激活的程序。程序结束的时候会自动关闭窗口。调用客户端即可输出结果。Trace（）结果采用文本形式输出到 Ecouté 状态窗口中。如果激活了多个客户端，则会在启动程序的客户端中输出结果。

- 尝试其它功能

Vediamo-Java 接口说明中包括了所有类及其方法的完整列表（类功能）。Sun Microsystems Java 主页包含 Java 编程语言的完整说明。

2.6.5 特殊情况

放弃 Java 程序

如果从 Ecouté 中采用直接方式或者使用标准对象启动 Java 程序，则可以采用 Active routines 对话框放弃该 Java 程序。如果至少执行了一个 Java 程序，而且激活了 routine Manager 选择，则会显示该对话框。对话框中会显示 Ecouté 客户端启动的当前执行的所有 Java 程序。此外，还会显示每个 Java 程序的启动时间。如需放弃 Java 程序，请在对话框中选择相应的行，并且点击 Abort routine 按钮。无法放弃初始化程序以及作为前置条件在 Ecouté 客户端前台运行的 Java 程序。

注意：

如果正在执行的 Java 程序定期调用 Vediamo 句柄功能，则仅可由 Vediamo 系统放弃该程序。编写 Java 程序的时候，请务必考虑这种情况。

使用.jar 文件格式的 Java 程序的特殊特性

Java 程序可以作为.class 或者.jar 文件方式来运行。如果运行.jar 文件，则必须注意特殊特性：如果希望在 Java 程序以及启动程序的客户端之间建立联系，比如，通过 JVClient.Information（）函数，则启动客户端传送唯一标识符作为 Java 程序的最后一个指令行参数...仅当将该指令行参数传送给函数 JVUtil.SetScriptIdentifier（）（最好在其 main（）方法启动之后立即使用）的情况下，才可启用通讯。否则，无法将客户端识别为预期的通讯目标。如果运行.class 文件，则 Vhandlers 框架会自动从事此类工作，如果使用.jar 文件，则必须由自身的 Java 程序从事此类工作。

2.6.6 配置（INI 参数）

为了确保正确执行 Java 程序，必须将下列数值正确地输入到配置文件 Vediamo.ini 中：

[INTERPRETER]

ClassPath – 路径、JAR 文件以及使用的 Vediamo-Java 类。

VHandlersLib – 提供的 VHandlers.DLL 的路径。

JavaInterpreter - Java 解释器的路径。

Vediamo 设置可以输入此类，并且仅可由用户在意外情况下进行更改。

此外，以下项

UseJVHandlersPackage=0

用于控制是否需要在定制 Java 程序中使用各种包（比如，新版本的 JBuilder 规定必须这样做）。

2.7 黑盒

2.7.1 引言

即使没有进行规划 – 但是不幸的是，也会出现崩溃的情况。必须确定相关原因，并且在今后力争避免出现此类情况。在航空业中，这种情况通常会采用黑盒辅助实施此类措施。

在本次情况中，黑盒属于一种检测 Vediamo 模块功能（服务器以及客户端）的程序，具体做法是从应用程序中接收当前程序状态项。如果发生重大事件（服务器或者客户端崩溃，丧失联系），则会将缓存的项存储在日志文件中。

2.7.2 结构以及功能

可以采用 INI 参数 “Run” 来激活或者去活黑盒功能。安装之后，默认情况下处于关闭状态。黑盒会在后台采用静默方式运行。可以采用任务栏中的  图标读取其菜单。黑盒通过 DCOM 接口与所有 Vediamo 应用程序（服务器以及客户端）建立联系。此类应用程序将其日志项传送至黑盒，后者将其存储在环形缓冲中。一旦诱导应用程序或者用户的请求（通过任务栏菜单），则会将最近的几个可用项（最多可达 5000 行，但是通过 INI 参数可以更改这一数字）存储在文件中。

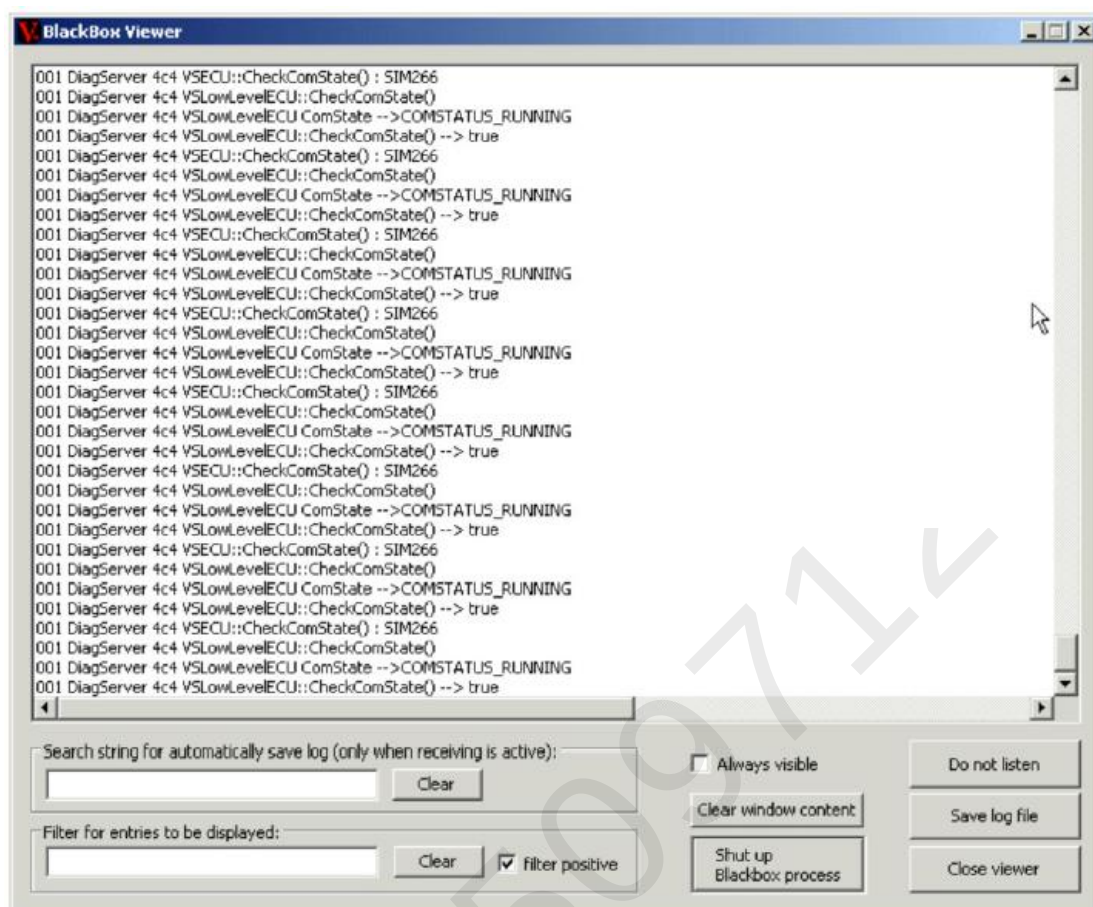
2.7.3 黑盒功能



可以通过任务栏菜单访问下列功能：

- Info about BlackBox。显示的信息窗口包括程序版本以及配置文件等信息。
- Log Viewer。启动程序，后者会持续显示日志项。
- Save Log into File。如果发生异常事件，则用户可以保存日志，以便提供给开发人员进行分析。
- Shut up BlackBox。如果发生异常情况，可以导致黑盒无法自行结束（比如，应用程序崩溃之后）。本菜单项可以结束黑盒并且恢复至定义的初始状态。

2.7.4 BlackBoxViewer: 运行时显示的日志



多数此类窗口控制元素均具有自解释功能。搜索字符串掩码需要特别注意。

- 用于自动保存日志的搜索字符串。在此输入文本字符串。一旦在日志项中找到该字符串，则会将直至该点的所有可用项存储在日志文件中。

请注意：

搜索对于大小写敏感。

- 需要显示的各项专用过滤器：此处您可以限制仅显示选中的信息。如果激活了 **filter positive** 选项，则只会显示包括搜索字符串的项。过滤器仅会影响窗口中的显示内容，可能存储的日志文件内容不会受到影响。

注：

运行时显示会增加 CPU 的负担。在某些情况下，这可能导致显示与关闭 **BlackBoxViewer** 之后显示不同的序列。对于依赖于时间的序列来说，我们建议重复这一情况而不打开日志窗口，并且采用任务栏图标保存日志。

2.7.5 与其它应用程序链接

如果您开发了定制的程序，而且该程序与 **Vediamo** 配合使用，那么可以同样可以针对其采用黑盒。如需获取 **DCOM** 接口的说明，请联系 **Vediamo** 团队。

2.7.6 配置 (INI 参数)

本程序在 Vediamo.ini 中拥有三个参数。

- Run – 黑盒激活 (1) 或者去活 (0)。
- LineNum – 日志文件中的项计数。
- 指定了路径以及日志文件名称起始字符串。采用该项构造全名, 并且辅以日期以及时间。

示例:

```
[BLACKBOX]
```

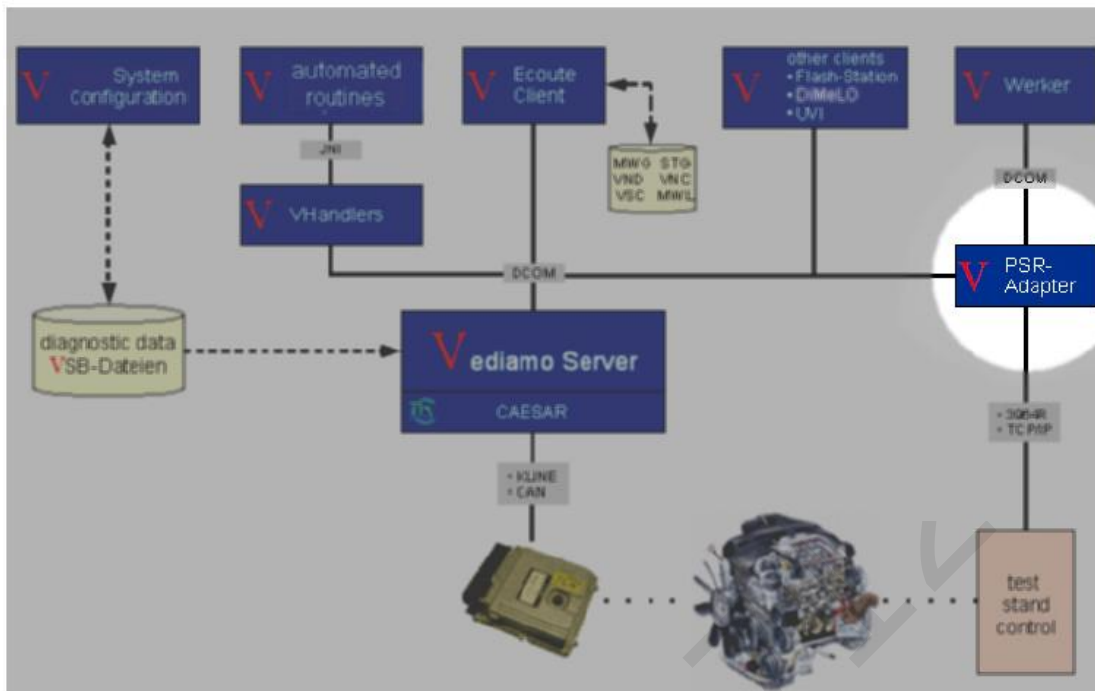
```
Run=1
```

```
LineNum=5000
```

```
LOGFILE=D:\Data\Log\BlackBox
```

这种情况下, 可以对文件进行命名, 比如 BlackBox_03-11-2005_17-23.log。最多可包含 5000 行的日志信息。

2.8 PSR 适配器



2.8.1 引言

以下文字中，PSR 的缩写是指测试台控制器 (= Prüfstandrechner, PSR)。

PSR 客户端是一种实用程序, 给予测试工作台控制器访问 DiagServer 诊断功能的权限。其目的摘自针对与测试工作台控制器的通讯中采用综合 Vediamo 接口。

激活工人客户端的 Connect to Server command, 即可启动 PSR 客户端。PSR 客户端打开与 PSR 的接口, 并且激活 DiagServer。如果工人客户端终止 (如果一个以上被激活, 那么会终止最后一个工人客户端), 那么 PSR 客户端也会终止, 此外还会终止 DiagServer (如果没有 Ecoute 会饿 Java 程序等其它客户端处于激活状态)。

如果无需任何接口, 则还可以通过 DCOM 调用方式, 从测试工作台中直接启动 PSR 客户端。这种情况下, 关闭测试工作台软件会停止 PSR 适配器。此外, 为了此类标准的目的, 还可采用命令 (702) 从 PSR 中, 或者从任务栏菜单 (右键点击  任务图标) 中终止 PSR 客户端。

2.8.2 Vediamo 与 PSR 之间的通讯

通过 PSR 客户端的单独消息中可以出现 PSR 以及 Vediamo DiagServer 之间的通讯。此类消息块可以采用两种方式进行传送:

- 通过 COMx 接口采用 3964R 协议进行串行传递
- 通过与采用 TCP/IP 协议的 HDLC 联系的网络

两种方式中的消息内容均相同。

测试工作台控制器 (PSR) 为主, PSR 客户端为从。这意味着, PSR 发出的消息具有指令的含义, 可以启动、停止某些特定动作, 或者执行一次此类动作。PSR 客户端可以随时并且采用可变次序接收指令。指令是否具有预期功效主要取决于系统状态。

PSR 接收 PSR 客户端发出的各种信息，此类信息是执行各种动作以及各种事件得出的结果。PSR 客户端发出的某些消息是由一个指令生成的（比如，指令 600 会生成消息 601），其它消息是在各种情况下发生各种数量改变而生成的（比如，循环测量读取过程中生成的消息 301）。此外，还有消息无需请求即可发送至 PSR。此类消息主要涉及系统错误以及系统状态的变化等（比如，与 ECU 失去联系）。

下列规则适用于所有消息：

PSR 适配器会发送每一条消息，直至 PSR 承认已经收到为止。例外：在循环测量读取的过程中，会从缓存中清空未发送的测量，这样可以确保进发送最新的数值。

通讯中断时的行为

如果与 PSR 的通讯中断，则不会丢失任何数据。PSR 客户端会对接口进行初始化，并且一旦建立联系即会发送缓冲中包括的所有块。

这样导致 PSR 在建立联系之后会首先接收来自 PSR 客户端的消息，比如，PSR 重新启动，而 Vediamo 仍然处于测试运行状态，同时缓存中存在消息的情况。

对于测试工作台系统来说，如果此类行为导致出现某些问题（比如，来自 AT&T 的原有测试工作台），则可以设置 INI 项 COMTIMEOUT，这样可以确保结束测试运行并且在通讯中断时间较长的情况下清空传输缓存。新启动的 PSR 之后会查找处于规定状态的 PSR 客户端。

消息块结构

长度：2 个字节 (HI, LO)	时间戳：2 个 字节 (HI, LO)	服务编号：2 个字 节 (HI, LO)	可选数据：0 至 1018 个字节词 语、字符串或者二者组合
块的长度，包括长 度、时间戳以及服 务编号	程序启动之后 的时间，单位 为 1/10 秒	在 DiagServer 中 启动某种动作的 唯一编号	取决于服务编号。采用 (HI, LO) 的次序传送词语。字符串是指非 零结尾的 ASCIIZ 字符串。

PSR 发出的每个块均表示发送至 DiagServer 的指令。DiagServer 的回复（承认、数据、信息）均采用单个块的方式发送。一直重复尝试传送，直至接收方（PSR）依照协议的相关要求承认收到为止。但是，循环传送测量结果是例外情况 – 此类信息仅发送一次。

通过 3964R/ASYNC 或者 HDLC/TCP/IP 向/自 PSR 发送指令块。

需要在 PSR 中设定下列参数：

- 3964R / ASYNC:
 - Prio=1（即，如果同时进行传送，则 PSR 适配器会切换接收并且给予 PSR 相关优先级）
 - COM 接口：19200, 8, e, 1
 - 字节延迟=220 毫秒（块中各个字节之间的时间）
 - Ack 延迟=550 毫秒（承认所需的延迟）
 - 重试=6（传送每个块时协议最多会尝试 6 次）
- HDLC/TCP/IP
 - IP 端口=2049（可能存在超过 1024 的其它数值）
 - PSR 客户端=IP 服务器 - 首先打开套接字的 PSR 客户端
 - 控制计算机=IP 客户端 – 首先与正在运行的 PSR 客户端建立联系的 PSR

注：

可能超过 Byte Delay 以及 Ack Delay 数值，但是在接收端并不处于正在运行状态。这意味着，接收方需要至少针对该字节等待规定的时间。发送方必须在低于该时间段内完成传送。数值的增加仅会提高较慢接收方的接收保证，而不会影响运行时，因为通常情况下，会以相对较短的时间发送各个字节或者 ACK。

2.8.3 PSR 客户端的功能

PSR 客户端提供了最常用的 Vediamo 功能，此类功能与测试设施有关。这一范围之外的各种必要项均采用 Java 程序进行编程，并且集成到测试运行之中。

这些单独的功能如下所示：

- 从选定系统中载入并且初始化数据。
- 建立以及终止与 ECU 的联系。
- 采用以及不采用环境数据的情况下读取错误。单次或者循环直至指令结束为止。
- 读取所有或者选中的测量结果。单次或者循环。

测量读取时可以进行时间优化的。引用的时候可以采用编号而非 ID，这样可以降低数据的数量。对于通过 3964R 进行通讯的情况，这一点显得尤为重要，因为这种情况下的数据吞吐量可能低于 Vediamo 与 ECU 之间的数据吞吐量。

- 执行各种服务，其中包括驱动器以及调整（采用 ID 进行引用）。
传送（必要的情况下）参数，返回结果。此外还适用于测量工作。
- 闪存处理 ECU。
- 发送并且接收 API-1 消息，目标是 ECU。
- 执行 Java 程序。

Java 接口具有一个 JVPSR 类，后者包括各种功能，此类功能用于从 Java 代码将消息直接发送至测试工作台。

- 从 Vediamo 中读取各种信息：
 - CAESAR 版本
 - ID 块
 - 定义的测量（在系统配置中主动进行过滤）
 - 服务列表
 - 服务详情
 - ...
- 重启 DiagServer。

在更新 CAESAR 而且必须采用服务器再次对其进行读取等情况下，这是非常必要的。PSR 客户端保持激活状态，而且与 PSR 的联系保持不变。

仅可采用 Java 程序而非 PSR 指令来实现变体编码。

测试运行的功能规则是事件控制的。这意味着，某些事件（比如，来自 PSR 的指令，失去与 ECU 的联系）会导致各种动作，而此类动作的时间或长或短，并且可能更改整个系统的状态。将 PSR 客户端中的每个动作通知 PSR。由于是在不同的平行任务中平行执行各种任务的，因此发送至 PSR 的消息序列可能与事件（指令）的发生次序存在一定的偏差。

必须在 PSR 程序中考虑以下问题，即，消息之间的次序或者时间无需保持恒定状态。但是同一个动作的消息属于例外情况，比如，启动 Java 程序以及程序的结束可能永远也不会采用相反的次序执行。

以下显示了测试运行序列的示例。

发送方	代码	参数	解释
PSR	3	“CR4_1”	马达的载入数据在发动机表中识别为“CR4_1”。
Vediamo	5	“CR4psr”	载入“CR4psr”系统数据。发动机表规定了何种系统说明属于何种马达。
PSR	100		建立与 ECU 的联系。
Vediamo	101	“CR4”	建立与 ECU “CR4”的联系。不可将系统命名为 ECU 之类的名称。
Vediamo	112		结束初始化阶段。在此之前可能存在其它消息，比如，是否自动执行初始化程序。
Vediamo	201	“P0115”，“Temperature sensor defective”	ECU 报告某种错误。建立联系之后，ECU 自动循环读取并传送此类信息。如果未出现任何错误，则仅会显示“End of error list”消息。
Vediamo	201	“”，“”（2 个空字符串）	错误列表结束。
Vediamo	201	“P0115”，“Temperature sensor defective”	下一个错误读取循环。循环读取错误，直至 PSR 将其结束为止。
Vediamo	201	“”，“”	
PSR	204		停止读取错误。
PSR	306	2，7	对测量 2 以及 7 进行一次读取。
Vediamo	301	2，“Off”	测量 2 的结果为“Off”。
Vediamo	301	7，“3250”	测量 7 的结果为 3250。
PSR	500	“ACT_Drosselklappe”，“75”	将驱动器“QACT_Drosselklappe”设定为目标数值 75。
Vediamo	502	3，“ACT_Drosselklappe”	错误：Vediamo 无法激活驱动器。（比如，未打开访问权限）
PSR	550	“FN_Zugriffsberechtigung”	执行服务。
Vediamo	551	“FN_Zugriffsberechtigung”，“Access privilege granted”	执行服务，并且提供了结果文本。
PSR	500	“ACT_Drosselklappe”，“75”	多次请求驱动器调整。
Vediamo	501	“ACT_Drosselklappe”，“75.2”	顺利完成驱动器调整，新数值为 75.2（由于浮点数向字节的转换问题，数值可能存在少许差异）。
PSR	700		结束测试。
Vediamo	102	“CR4”	关闭与 ECU 的联系。
Vediamo	701		完成测试运行。

如需自行编程进行测试运行，则应当打开所有消息的完整列表，而此类消息是可以在 PSR 以及 Vediamo 之间传送的。

2.8.4 配置

与其它 Vediamo 程序类似，PSR 客户端还可以从 Vediamo.ini 文件中获取相关设置。与其它 Vediamo 程序类似，可以在程序存储的同一个文件夹中搜索 INI 文件。

但是，通常情况下，包括相应的 DLL 以及 INI 文件的所有 Vediamo 程序均安装在相同的文件夹中，即，所有程序均采用通用的 Vediamo.ini。

在[PSR]部分设置 PSR 相关项。

关键词	解释
COMPORT	用于 PSR 以及 PSR 适配器之间通讯的连接。 0=与 PSR 无通讯。程序未激活。 1 或者 2=采用 COM1 或者 COM2 的 3964R 协议。如果数值更大，则必须按照具有其它端口的 I/O 卡。 2049=采用 IP 2049 端口的 HDLC/TCP（其它数值可能超过 1024）。
COMTIMEOUT	时间以毫秒为单位。如果与 PSR 的通讯长时间中断，则可能结束测试运行（与 ECU 的连接中断）而且从传送缓冲中删除所有数据。 0 表示无超时。一旦重新建立联系，则会继续测试运行，而且不会删除任何消息。 默认数值：0
ENGINETABLE	发动机表：定义各个马达的文件（指定马达 ID – 系统说明+初始化程序+可选程序参数）。该文件是必须强制使用的。
LOGFILE	采用字节方式日志记载 PSR 与 PSR 客户端之间的数据交换的文件。仅可在搜索错误的情况下激活，正常情况下，该项为空。

示例：

COMPORT = 1

LOGFILE = "C:\Log\LUCA.LOG"

ENGINETABLE = C:\Vediamo\Systembeschreibungen\Engine.vet

注释：

大多数情况下，可以在 INI 文件中输入不带引号的文件名。由于第三方来源的程序模块会使用 LOGFILE 项，因此必须确保采用引号方式列出路径名称，其中也包括驱动器盘符。

2.8.5 发动机表

发动机表规定了采用 ECU 的何种数据集对哪一台发动机进行诊断。文件结构如下所示：

- “;”开头的行表示注释，并且执行的时候予以忽略
- 所有（其它）行，均采用发动机 ID 开头，后面是系统 ID（VSB 名称），后面是可选的初始化程序名称，可能包括一些参数。采用空格或者 TAB 作为分隔符。
- 空行或者文件结束表示该表结束。

对于包括多个 ECU 的系统来说，可以选择需要测试的 ECU。为此，应当在系统 ID 后面指定 ECU ID，之间采用“/”分隔，不得采用空格或者 TAB。如果未指定 ECU，则找到的第一个 ECU 被视为处于选中状态。

如果系统包括多个同一种类型的 ECU，则采用括号中的数字方式自动对其 ID 进行扩展（比如，ME20（1），ME20（2）等等）。

示例：

```
;This is a comment
```

```
;
```

发动机名称为 Aklasse，具有一个 Sim4，我们采用 sim4serie.vsb。测试运行启动的时候会执行一个 Java 程序。

```
Aklasse sim4serie Init.class
```

```
;
```

汽油发动机采用了 ME2.0，没有 Java 程序。

```
Benziner me20
```

```
;
```

对于柴油发动机来说，我们拥有系统 cr2cr3.vsb，其中定义了 CR2 以及 CR3。原有的以及新的柴油发动机采用相同的系统，但是 ECU 不同，而且采用 Java 程序进行初始化，不过，采用了不同的参数：

```
DieselAlt cr2cr3/CR2 InitDiesel.class alt
```

```
DieselNeu cr2cr3/CR3 InitDiesel.class neu
```

```
;
```

具有两个相同 ECU 的系统。如果两台发动机采用相同 ECU 的时候，但是采用了不同的错误/测量过滤器的时候，适用这种方式。

```
DieselMitSLP Diesel/CR2(1) Varcode.class slp
```

```
DieselOhne Diesel/CR2(2) Varcode.class ohne_slp
```

系统何时会采用多个 ECU？

很少会出现同时与多个 ECU 联系方可测试某种发动机的情况。应当可以使用具有多个 ECU 的系统。特别是在测试工作台中替代性地使用各种不同的 ECU 类型。具有多个 ECU 的系统需要在运行时使用更多的 RAM，但是载入的 ECU 数据会永久性地保留在存储器中。这样做具有以下优点：

- 测试运行开始阶段（除第一种情况之外）一般较短，因为无需载入数据（根据数据量，可能需要 10-20 秒的时间）。
- 并未对存储器进行碎片化处理。这样可能导致 PC 运行速度不断变慢，而且必须在数百次测试运行之后重启。

使用此类系统不会对测试程序进行任何更改。PSR 也无需进行任何更改。仅需采取适当方式准备系统文件以及发动机表。

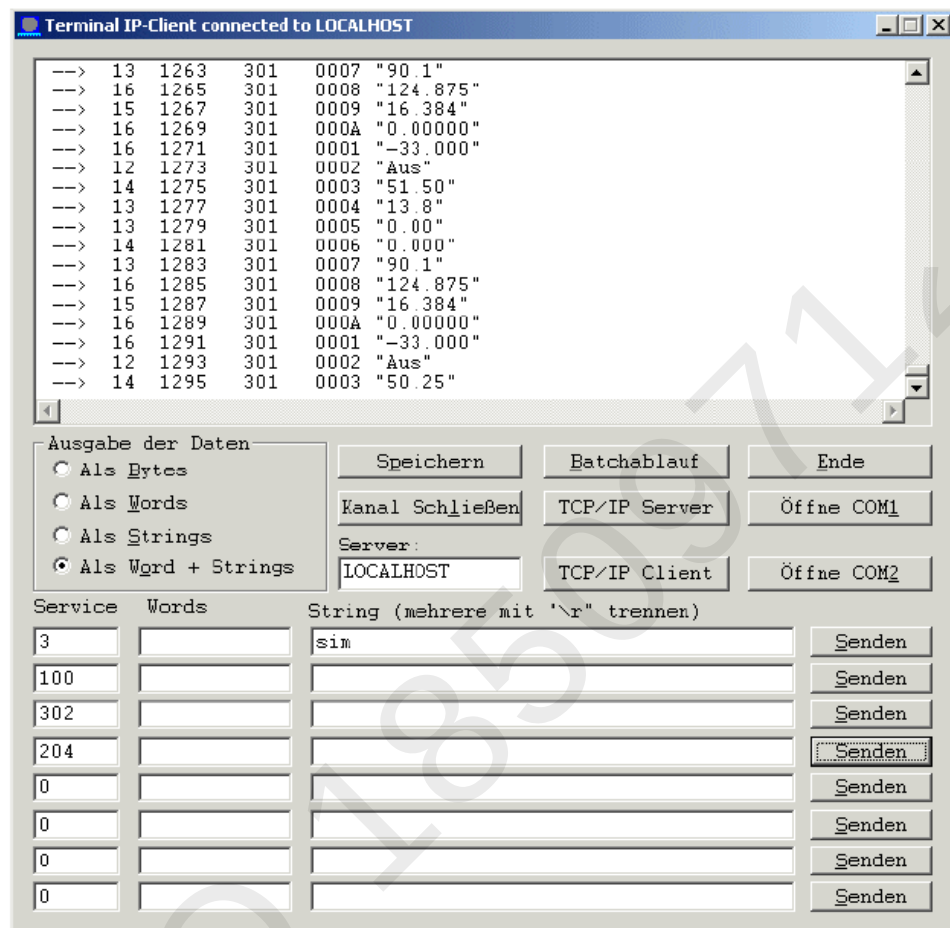
2.8.6 示例 – 如何...

无需 PSR 即可测试 PSR 客户端

可以在 Vediamo 文件夹中找到 PSRTerm.exe 测试程序。该程序使用的通讯接口与 PSR 客户端相同，而且允许采用手动方式输入通常由 PSR 发送的消息，还可以接收来自 Vediamo 的回复。

注：

我们不保证也不支持这种测试工具。



如需使用 PSRTerm，首先需要启动 PSR 客户端。

对于串行连接来说，正在运行 PSRTerm 的计算机 COM1 或者 COM2 必须采用空值调制解调器线缆与 Vediamo 计算机接口建立联系，后者是在 Vediamo.ini 中输入的。该线缆连接两个端口的 GND、TX 以及 RX 针，TX 与 RX 相连，RX 与 TX 相连，GND 与 GND 相连。然后点击 Open COM1 或者 Open COM2 即可与 PSR 客户端建立联系。

对于网络连接来说，两台计算机必须处于同一个网络之中，而且可以采用指定端口进行连接。此后仅需采用 PING 指令，无需 DCOM 配置、共享以及发布。可以通过在 Server 中输入 Vediamo 计算机名称或者 IP 地址，然后点击 PSRTerm 中的 TCP/IP Client 即可与 PSR 客户端建立联系。或者，PSRTerm 可以在同一台计算机上作为 PSR 客户端运行。这种情况下，计算机名称为“localhost”。您的计算机的地址为 127.0.0.1。如果无法在 PING 成功的情况下依然无法建立，请检查您的防火墙设置（比如，Windows 防火墙）。

如需重新建立中断的联系，请务必总是先点击 Close channel，然后在按照上述步骤继续操作。

不启动 Vediamo 的情况测试 PSR 接口

PSRTerm 同样适用于此类任务，首先启动 PSRTerm，并且在 PSR 建立联系之前，采用 TCP/IP Server（用于网络）或者 Open COMx（用于串行连接）激活连接。

注意！

采用 3964R 串行连接的情况下，如果两台机器均试图同时发送块的时候，连接会出现堵塞的情况。这是因为 PSR 以及 PSRTerm 均具有 PRIO 参数，而且均设定为 1。在正常的 PSR-Vediamo 操作中，不会出现此类情况，因为 PSR 客户端的 PRIO=0。

采用 DDLID 加速测量读取

采用动态定义的局部 ID (DDLID)，可以更高效而且更快速地从 ECU 中读取测量结果。另外一个优点是，会同时计算采用这种方式读取的数值。因此在可能的情况下，针对循环读取测量结果（指令 302）自动采用 DDLID。指令以及数据格式不会受到影响，只有执行速度会发生变化。

如需在 PSR 客户端中使用 DDLID，则必须满足下列条件：

- ECU 必须针对选定的测量结果支持 DDLID。
- 必须在系统文件（VSB）中正确输入 DDLID 的代码。
- 没有其它客户端使用同一个 ECU 的 DDLID。

缩短测试运行的初始化阶段时间

测试运行开始的时候，在与 ECU 建立联系之前，会首先载入数据并且进行初始化。在此期间，基本 ECU 变体中定义的所有测量服务均会作为软件对象进行初始化。建立联系并且识别 ECU 变体之后，Vediamo 会对识别的变体中定义的所有测量数值进行初始化。

现代 ECU 最多可拥有数千个测量数值，根据 PC 的性能，此类初始化可能需要 2-3 分钟的时间。但是，由于在测试运行中仅需测定其中一些测量数值，因此应当使用系统说明（VSB）中的系统配置来过滤掉无用的所有测量数值。应当针对基础变体（缩短建立连接之前的初始化阶段的时间）以及针对使用的变体（缩短建立连接之后的初始化阶段的时间）采取此类措施。

过滤掉其它服务（驱动器、调整等）并不会影响初始化阶段，因为此类服务与测量数值不同，仅在需要的时候才会在 PSR 适配器中进行初始化。

如上所述，通过采用包括所有必要的 ECU 的系统说明而非多个系统说明，即可明显缩短数据的初始化阶段的时间（建立联系之前）。每次发动机变更的时候，这种做法可以最多节省 10 秒的时间。如果必要的系统说明 与之前的系统说明相同，则 PSR 适配器不会重新载入数据。

监控 PSR 适配器与 PSR 之间的通讯

工人客户端可以显示并且存储交换的消息。通过这种方式可以对测试过程进行监控。

如果测试工作台与 PSR 适配器之间的通讯存在问题，可以在较低级别对通讯情况进行日志记载。请在 INI 文件中输入日志文件的名称。

示例：

[PSR]

LOGFILE=W:\LOG\Luca.LOG

下次运行 PSR 适配器的时候，所有实际传送以及接收的字节均会存储在日志文件中。以下文本框包含了日志文件的示例。

示例:

PSR Comm log file W:\LOG\Luca.LOG

Port: 2049

Date: 3.11.2005

Time: 9:4:19

-1721665.-634[01] hdlc->APPL 0000 00 0d 00 e0 00 03 73 69 6d 32 36 36 00 '...'..sim266.'

-1721665.-544[01] APPL->hdlc 0000 00 11 01 3e 00 32 53 69 6d 75 6c 61 74 69 6f 6e '...'>.2Simulation'

-1721665.-544APPL->hdlc 0010 00 ''

-1721654.-408[01] APPL->hdlc 0000 00 36 01 ac 00 32 46 45 48 4c 45 52 3a 20 46 65 '6...'>.2ERROR: Er'

-1721654.-408APPL->hdlc 0010 68 6c 65 72 3a 20 55 6e 67 fc 6c 74 69 67 65 72 'ror: Invalid'

-1721654.-408APPL->hdlc 0020 20 53 79 73 74 65 6d 6e 61 6d 65 20 53 49 4d 32 'system name SIM2'

-1721654.-408APPL->hdlc 0030 36 36 70 73 72 00 '66psr.'

-1721645.-986[01] hdlc->APPL 0000 00 06 01 a4 04 4c '...'\$.L'

-1721645.-916[01] APPL->hdlc 0000 00 13 02 03 04 4d 43 52 33 20 63 72 33 5f 75 70 '.....MCR3 cr3_up'

-1721645.-916APPL->hdlc 0010 73 72 00 'sr.'

-1721614.-110[01] APPL->hdlc 0000 00 30 03 3e 00 32 46 45 48 4c 45 52 3a 20 46 65 '0...'>.2ERROR: Er'

-1721614.-110APPL->hdlc 0010 68 6c 65 72 3a 20 55 6e 67 fc 6c 74 69 67 65 72 'ror: Invalid'

-1721614.-110APPL->hdlc 0020 20 53 79 73 74 65 6d 6e 61 6d 65 20 6e 61 67 00 'system name nag.'

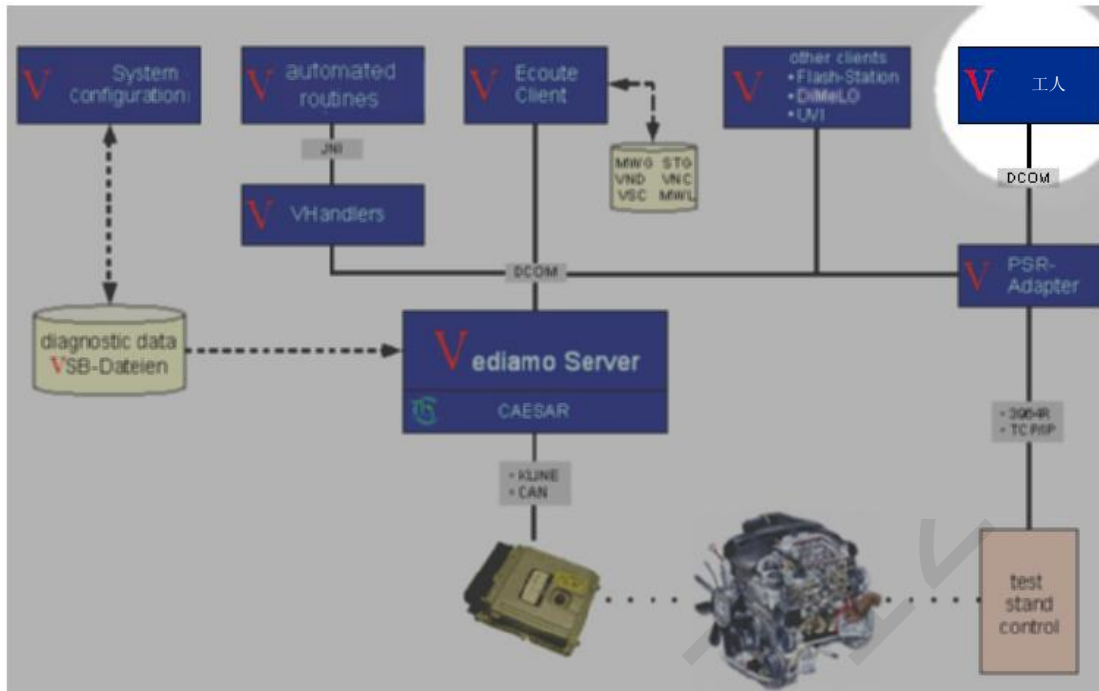
-1721155.-441[01] APPL->hdlc 0000 00 2e 15 2b 00 32 56 50 53 52 43 6c 69 65 6e 74 '...'>.2VPSRClient'

-1721155.-441APPL->hdlc 0010 3a 20 56 65 72 62 69 6e 64 75 6e 67 20 77 69 72 ': Connection is'

-1721155.-441APPL->hdlc 0020 64 20 67 65 73 63 68 6c 6f 73 73 65 6e 00 'being closed.'

Closing log file: 9:13:30

2.9 工人客户端



2.9.1 引言

工人客户端显示了图形化用户界面，主要用于初始化 PSR 客户端以及显示以下内容：

- 测量结果
- 错误
- 重要状态变更以及动作的状态消息

上述内容是在自动测试运行（由 PSR 进行控制）期间由诊断服务器传送的。应用程序不会对测试产生任何控制影响。仅会影响显示以及日志信息的保存方式。

建立与服务器的联系的过程中会对 PSR 客户端进行初始化。PSR 客户端属于独立的程序，该程序构成诊断服务器与 PSR 之间的接口。PSR 程序不具有用户接口（窗口）。因此，为了监控测试，必须采用工人客户端。

采用 **Connect with Server** 指令对 PSR 客户端进行初始化。在此之前，PSR 无法通过 3964R 或者 TCP/IP 与服务器建立联系。工人客户端的数量（可以在各种 PC 上执行）并不重要。关闭最后一个客户端与服务器的联系会结束 PSR 客户端，而且，如果没有其它激活的 Vediamo 客户端，则还会结束服务器。

工人客户端用户界面支持多种语言。在 Vediamo.ini 配置文件中的[COMMON]中的 Language 项可以设置语言。

2.9.2 结构

用户界面主要包括标题栏、菜单、三个显示区域以及独立的日志窗口：



测量窗口

该窗口中会显示 PSR 请求的所有测量结果（名称、数值以及单位）。如果结束了测量读取循环，则仍然会显示最后一次读取的数值，但是显示的时候会使用括号。一旦选择了新的测量或者开始新的测试运行，则会清空显示的数值。

错误窗口

测试运行期间报告的所有错误均会在此显示（代码以及文本）。错误代码旁边亮起的红灯表示最后一次循环中该错误报告为“current”状态。熄灭的红灯表示“已存储的”错误。窗口内会存储测试运行期间报告的所有错误。在开始新的测试运行之前，不会删除错误窗口中的内容。

状态窗口

在此显示重要的动作：与 ECU 建立联系（已建立、已结束、已初始化）、切换驱动器、启动与结束 Java 程序，以及诊断服务器中的错误以及其它重要事件。

标题栏

此处会显示测试运行的当前状态（程序启动以来的数量、测试的系统的名称、测试运行状态、ECU：ID 以及变体）。

日志窗口

点击 F9 = log window 的情况下会打开或者关闭该窗口。该窗口显示了 PSR 接收以及发送的数据块。仅会显示最近 1000 个块，但是还可以将数据继续存储在文件中。

2.9.3 功能说明

调试功能

F9 = Log window 可以打开或者关闭日志窗口。PSR 与 PSR 适配器之间接收以及发送的所有块均会在此显示。无论是否打开日志窗口，在运行时期间，均可连续将日志存储在文本文件中。日志窗口的内容还可以在较近的时间点存储起来，但是最近情况中仅包括最新的 1000 个消息块。

程序启动的时候，选项菜单中的适当项会自动打开日志窗口。可以在选项中指定日志自动存储所需的日志文件。

注意！

日志窗口中显示的内容不总是对应于 PSR 与 Vediamo 之间的数据交换。其中包括的是 PSR 发送而且 Vediamo 服务器正确接收的块，以及 Vediamo 准备发送给 PSR 的块。因此，如果 PSR 与 Vediamo 之间联系中断，那么日志中包括的块是 PSR 不再接收的块。此外，日志中列出的时间是来自诊断服务器的时间，而非传送的实际时间。

如果怀疑数据传送出现任何问题，则应当将数据交换保存在一个独立的文件中。可以在 Vediamo.ini 文件的[PSR]部分的 LOGFILE 项中指定该文件。PSR 客户端生成的该文件包括 Vediamo 与 PSR 之间实际交换的所有字节的准确记录。

F12 = Info...显示了包括当前程序版本以及当前实现的 INI 文件路径相关信息的窗口。

2.9.4 示例：如何...

显示测试的 ECU 的识别块

点击 F2 会显示已连接 ECU、已连接的系统以及与 ECU 已经建立联系的识别块。

显示 CAESAR 软件版本

点击 F3 会显示已实现的 CAESAR 软件版本 (c32s.dll)。

适配器用户界面

工人客户端的外观可以正常适应您的需求。拖拽鼠标即可更改窗口与子窗口的位置以及大小。此外，点击 F7 可以打开设置窗口，其中可以更改字体以及字体大小（详情参见配置）。程序结束的时候，所有更改均会自动保存到 Vedioamoini 文件中。

2.9.5 指令行参数

如果您采用指令行参数启动 Werker_Client.exe，那么程序会尝试立即与服务器建立联系。此类参数必须是计算机名称或者服务器的 IP 地址。

示例：

```
werker_client localhost
```

或者

```
werker_client 127.0.0.1
```

2.9.6 配置 (INI 参数)

与其它 Vedioamoini 程序相同，当前程序的设置也存储在 Vedioamoini 文件中。保留[Werker]部分用于工人客户端。

设置

Server

服务器名称

一旦建立连接，则会自动将 PSR 适配器的名称复制到 Vedioamoini 中。如果在后续程序启动的时候，在此调用建立连接，则会提示上次建立联系的服务器。

Warnings

显示警告

本参数仅可在 INI 文件中直接更改（采用 notepad.exe 或者 INI 编辑器）。设置完毕之后，视为中会显示警告信息，正常测试运行中可以忽略此类警告，但是此类警告对于正在工作中的测试工作台调试而言是非常有用的。

Logfile

如果将本项设定为有效的文件名，则打开日志窗口（F9 按键）的时候，会自动打开指定的文件，而且会持续保存所有日志项。

用户界面的配置

可以通过鼠标拖动方式更改窗口位置以及大小。此外，还可以采用选项对话框（菜单项 F7=Options）来更改下列设置：

- 三个子窗口的字体
- 测量显示为名称或者修饰符
- 在启动程序过程中自动打开日志窗口

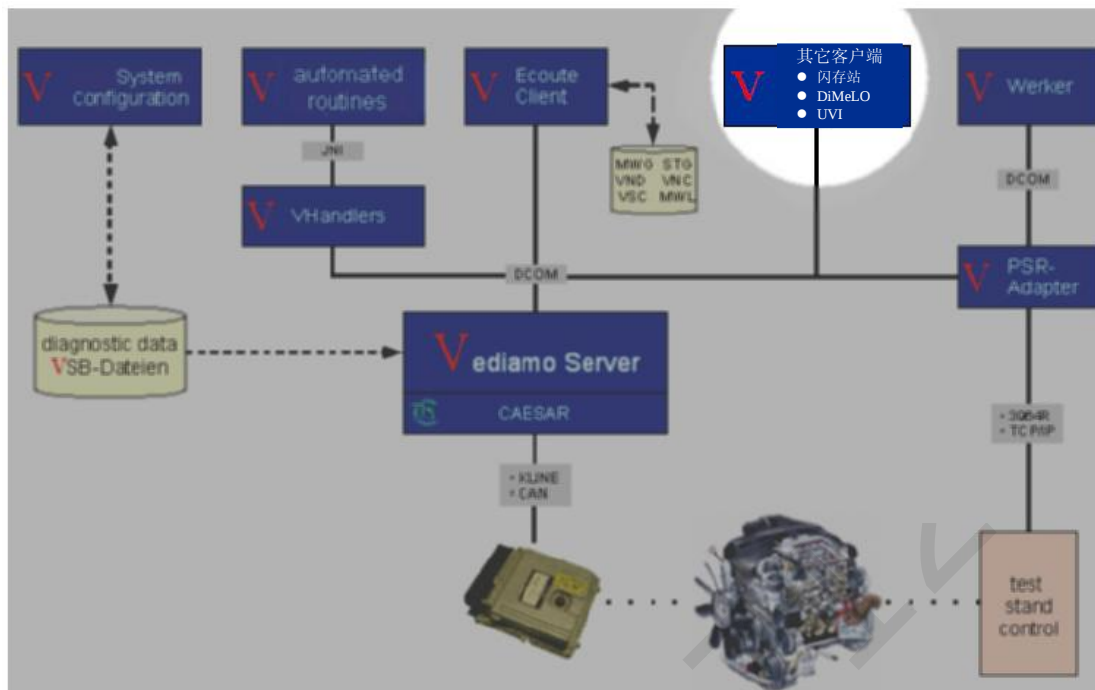
- 日志窗口打开的时候自动打开日志文件

此列设置均自动保存到当前的 `Vediamo.ini` 中。下次启动程序的时候，将保证程序拥有与上次运行相同的外观。

可以在客户端中直接更改工人窗口的外观适用项（大小、位置、字体），这种情况下无需使用编辑器。仅当出现“损坏”的用户图形界面窗口的时候，才推荐在编辑器中删除所有项，以便将窗口恢复为默认状态。这种情况适用于“Size”参数以及包含“Font”字符串的所有项。

QQ 18509711

2.10 其它客户端



为了保持介绍的完整性，这里提及了其它一些 Vediamo 客户端：

2.10.1 闪存处理工作站

Vediamo 闪存处理工作站

Vediamo 闪存处理工作站（VFS）主要用于自动批量闪存处理，以及确保流程的闪存处理。

用户对于选择适当闪存件的干预降至最低程度，或者可以通过配置完全省略选择步骤。此后，可以由未经培训人员进行闪存处理（比如，库存更新）。详情参见 Flash Station Help。

2.10.2 DiMeLo

Dimelo – 自动记录了测量数据。

详情参见 Dimelo Help。

2.10.3 UVI

UVI - Unipas Vediamo 接口

Unipas 通过本接口使用 Vediamo 来闪存处理 ECU，并且获得 XML 格式的诊断信息。

2.10.4 更多的客户端以及实用程序

- PSRChecker

本程序用于验收操作步骤，以便在测试工作台软件中实现 Vediamo 接口。该程序对很少但是可能出现的情况进行了仿真，此类情况是测试工作台程序必须予以处理的。

- PSRTerm

这是一种免费测试程序（不保证支持而且不承担任何责任），用于手动仿真通过 TCP/IP 与串行连接（3964R 协议）与 PSR 进行的通讯。

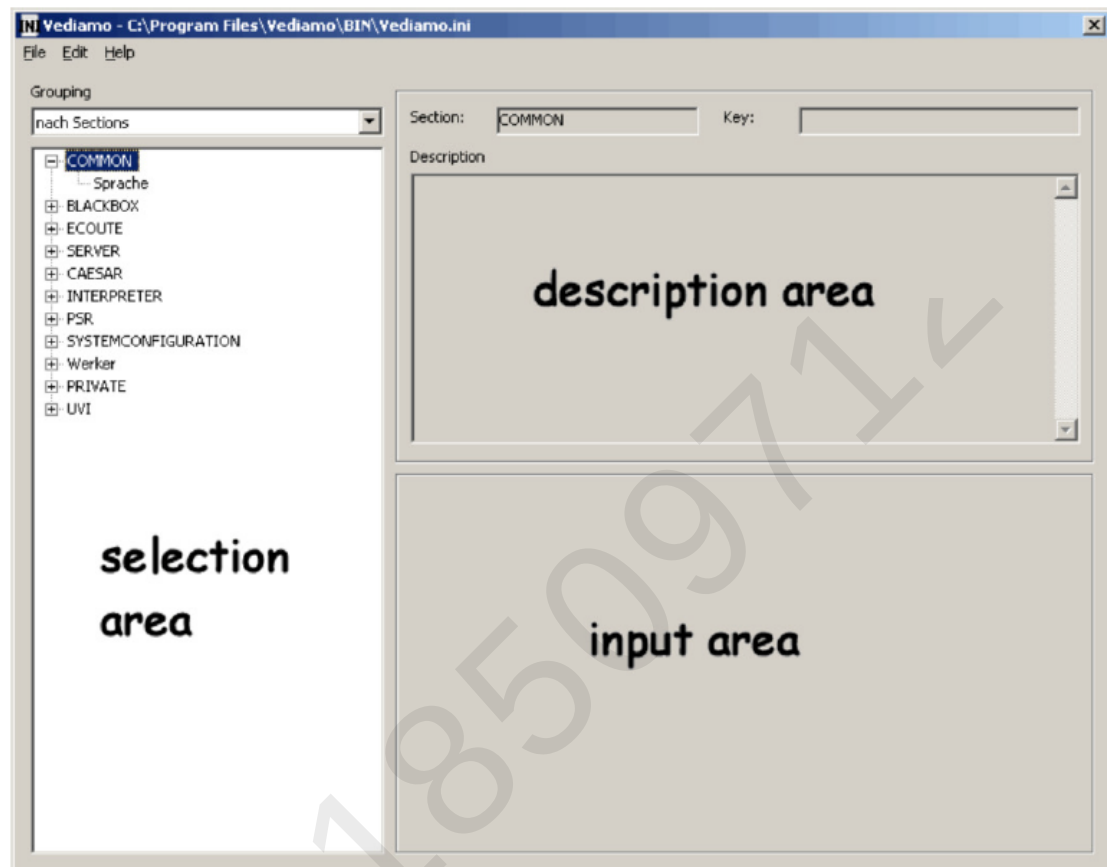
2Q 1850971v

2.11 INI 编辑器

对于设置所有参数来说，本应用程序用于 Vediamo 系统。点击工具条上的 Options 按钮即可从 Start Center 激活本应用程序。提供了每个参数以及某些情况下的合规数值的详细说明，设置存储在 Vediamo.ini 文件中。

请注意：

如果您更改的参数会影响某个模块的功能，那么重启之前此类更改不会立即生效。



2.11.1 菜单

- File
 - Open: 选择某个 INI 文件，并且将其载入。
 - Save: 将所有数值写入当前正在编辑的 INI 文件之中。
 - Save as: 选择某个 INI 文件，并且将所有数值写入其中。
- Edit
 - Reset this value: 将正在编辑的数值恢复为最初载入的数值。
 - Reset all values: 将所有数值（编辑器使用过的）恢复为最初载入的数值。
- Help
 - Help: 调用帮助文件。
 - About: 显示包含 INI 编辑器相关信息的对话框。

2.11.2 用户界面区域

在 Selection Area 内选择需要编辑的设置。可以提供一种或者多种不同的类别。

选择之后，选中设置的信息会显示在 Description Area 内。

更改设置所需的各种输入元素则显示在 Input Area 内。

2.11.3 输入元素

布尔值输入（真/假、是/否...）

A screenshot of a user interface element for a Boolean input. It consists of a light gray rectangular box. On the left side of the box, there is a small square checkbox with a black checkmark inside. To the right of the checkbox, the text 'StatusTraceToFile' is displayed in a black, sans-serif font.

如需更改该数值，请点击单选框，或者选择 ECU，然后点击空格键。

数字以及字符串输入

A screenshot of a user interface element for a numeric input. It features a light gray rectangular box. Inside the box, at the top, is a white text input field containing the number '1000'. Below the input field, the text 'Min: 100' is on the left and 'Max: 10000' is on the right, both in a black, sans-serif font.

如需更改该数值，请点击或者选择输入行，然后更改其内容。

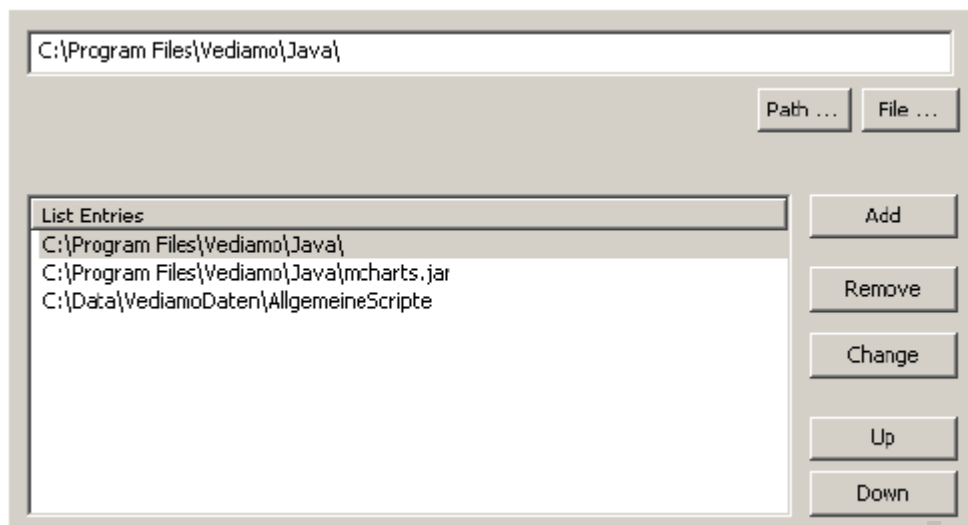
必须确保不得超过数字的最小以及最大限值。

路径以及文件的输入

A screenshot of a user interface element for a path input. It consists of a light gray rectangular box. Inside the box, at the top, is a white text input field containing the path 'C:\Data\VediamoDaten\Log\Vediamo'. To the right of the input field, there is a small rectangular button with the text 'Path ...' in a black, sans-serif font.

如需更改该数值，请点击“...”按钮，或者选中然后点击空格键，然后选择路径或者文件。

列表的输入



Add:

在输入行中输入某个数值，然后点击“Add”按钮。列表中，输入的数值处于选中数值的下方。

Remove:

从列表中选择某个数值，然后点击“Remove”按钮。

Change:

从列表中选择某个数值。更改输入行中的数值。点击“Change”按钮。

Up:

从列表中选择某个数值。然后点击“Up”按钮。

Down:

从列表中选择某个数值。然后点击“Down”按钮。

注意！

仅可将列表中的内容保存到 INI 文件中。保存时会忽略掉输入行中的内容。

预定义数值的输入



打开下拉菜单，进行适当的选择。

2.11.4 所有 INI 参数

附件中包括了 Vediamo.ini 文件中所有参数的完整清单。

3 如何...

- ...连接车辆
- ...没有车辆情况下连接 ECU
- ...对 ECU 进行闪存处理
- ...重启服务器
- ...从 ECU 中读取测量结果
- ...读取 ECU ID 块
- ...读取并且清空 ECU 错误存储器
- ...执行快速测试
- ...执行变体编码
- ...执行 Java 程序 (Java 程序)
- ...更改 K-线路与 CAN 之间的连接
- ...在我关闭的相同状态下打开 Ecouste

QQ 18509712

3.1 连接车辆

如需连接车辆，则可以采取下列任何一种方式：

- CAESAR Part A + B2 或者 Part Y or Part J，此外还包括 Part E4（配有 OBD 连接器）
- CAESAR Part C + 配有 OBD 连接器的特殊线缆
- CAESAR Part X + 计算机上的 WLAN 连接。Part X 配有 OBD 连接器。
- CAESAR Part W（SDConnect）

上述任何一种情况下，OBD 连接器均需要对应于硬件的线缆以及连接器。

将 OBD 连接器插入车辆诊断插槽内（位于驾驶员脚踏板附近，方向朝下）。

如果使用的 Part E4，请在 Vedio.ini 中配置[CAESAR] USE_SIPartEDriver=1。

请注意：

Part E4 必须在启动 Vedio 模块之前与车辆相连，因为 Part E4 需要车辆为其供电。如果 Part E4 并未与车辆相连，则 CAESAR 无法将其识别出来。

如果并未适用 Part E4，则不可能使用上述设置，即，[CAESAR] USE_SIPartEDriver=0。

3.2 没有车辆情况下连接 ECU

如需在没有车辆的情况下连接 ECU，则可以采取下列任何一种方式：

- CAESAR Part A + B2 + F
- CAESAR Part Y + F
- CAESAR Part C + 配有 Part F 的特殊线缆
- CAESAR Part J + OBD 连接器 + BreakOut-Box
- Part P - eCOM Box + F + USB2LAN 适配器
- CAESAR Part W (SDConnect)

以及

- 具有连接器、线缆线束或者适配器的 ECU。

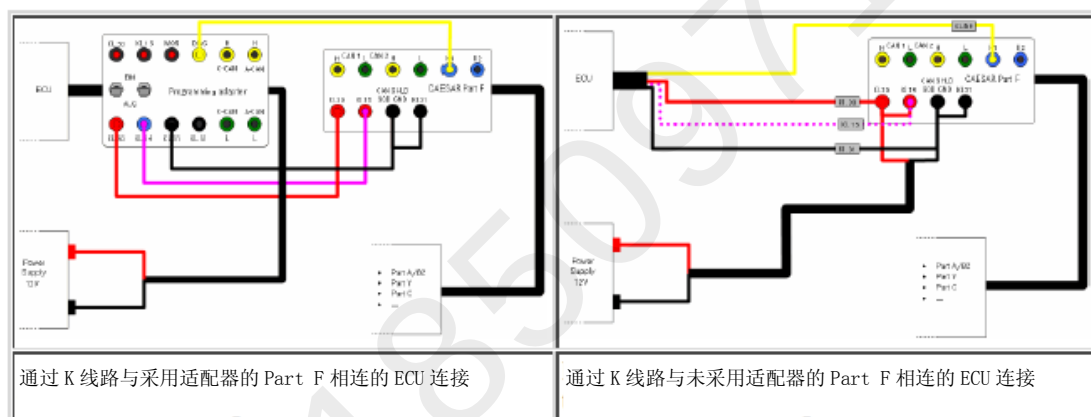
如需获取具有线缆线束或者适配器的连接器，请联系 ECU 开发商。

- 用于为 ECU 供电的电源（12 伏或者 24 伏）

如果使用了适配器，则还需要 4 毫米香蕉插头的测量线缆。

与 K 线路连接：

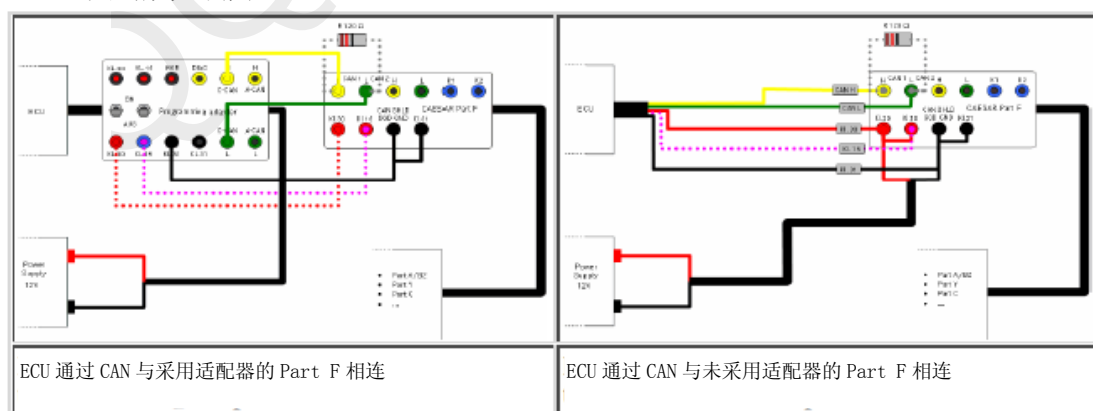
Ecoute 或者系统配置中的可用资源决定了是否可以通过 K 路线访问 ECU。如下图所示，连接 ECU（点击放大该图）。



请注意，通过 K 线路的连接需要为 Part F 提供电源。

与 CAN 相连：

Ecoute 或者系统配置中的可用资源决定了 ECU 是否可以访问 CAN。如下图所示，连接 ECU（点击放大该图）。



请注意:

- CAN 可能必须采用 120 欧姆端子, 这主要取决于 ECU 是否安装了终端电阻器。
- CAN SHLD (CAN 屏蔽/接地) 与 CAN Low 不同。无需连接此类触点。

2Q 1850971v

3.3 对 ECU 进行闪存处理

起点: .BIN-、.HEX- 或者 .S19 文件

Vediamo 仅可闪存处理 CAESAR 闪存文件 (CFF)，也就是说，您必须采用 DIOGENES 先将二进制文件从生产厂商的格式转换为 CFF 格式。

最佳方法：将文件以及各种技术规范传送给相关人员，令其摘取 ECU 数据文件，并且从此类人员那里接收 CFF 文件。

如果无法使用本方法（由于开发周期过短），还可以自行转换 CFF 文件。请注明摘取 ECU 数据文件的相关人员，因为这部分超出了本文档的范围。

请注意，CFF 格式的 ECU 名称（DIOGENES 名称）必须与 CBF 中的 ECU 名称保持一致。

起点: .CFF 文件

将文件复制到选项 (Vediamo.ini) 的 [CAESAR]CFFPfad 指定文件夹中。如果希望对 CFF 进行分类，那么还可以在指定的 CFF 路径中创建子文件夹，并且在此对 CFF 进行分类。

复制了文件之后，如果正在运行 Vediamo 服务器，则必须将其重启（详情参见任务栏）。

起点: CFF 位于 CFF 路径之中，而且启动了服务器

- 启动 Ecouté。
- 与需要闪存处理的 ECU 建立联系。
- **禁止运行**任何服务或者 Java 程序！应当通过闪存处理作业设置访问权限。
- 打开闪存处理对话框
- 按照 Ecouté 闪存处理的相关步骤进行操作。

载入 CFF 文件时候的行为：

为了减少服务器启动时间，在服务器初始化的时候载入 CFF 文件，具体取决于某些.ini

参数设置：

在 Vediamo.ini 中，存在一个参数 [Server]LoadCFFs。

该参数影响了服务器初始化时的 CFF 文件载入情况，以及何时使用此类文件。可以将该参数设定为 4 种不同数值：

0 载入所有 CFF：服务器启动的时候会载入位于 CFF 路径中的所有 CFF（默认设置）。

1 不载入任何 CFF：服务器启动的时候不会载入任何 CFF。（通过“Flashdata administration”载入 CFF）。

2 从 ECU 子文件夹中载入 CFF（必要的情况下）：系统选择的时候会载入位于\的所有 CFF。

3 从标准列表中载入 CFF（必要的情况下）：系统选择的时候会载入\\Flashfiles.cfg 文件列出的所有闪存文件。

详情参见 Ecouté “Flashdata administration”。

3.4 重启服务器

下列情况下，必须重启服务器：

- 在适当文件夹中放入了 CAESAR 文件或者 Vediamo 系统说明（VSB），并且希望 Vediamo 检测到此类文件。

注意：

在重启的时候，务必确保没有自动化应用程序或者 Java 程序正在运行当中，否则服务器重启的时候可能造成其崩溃或者至少无法正常运行。

Ecoute 运行的时候：

从菜单中选择 Extras/ Restart Server 菜单项。

其它：

请求服务器的时候，结束所有应用程序。

注：

StartCenter 也需要服务器。

3.5 从 ECU 中读取测量结果

读取单个测量结果

- 启动 Ecouste
- 打开包括关注的 ECU 的系统
- 与 ECU 建立联系
- 打开 ECU 下的树形控件
- 打开分组 Measurement 下的树形控件
- 双击关注的测量结果
- 读取状态窗口中的事件

同时读取多个测量结果或者循环读取测量结果

- 启动 Ecouste
- 创建服务分组
- 点击相关按钮读取测量结果

3.6 读取 ECU ID 块

- 启动 Ecouste
- 打开包括关注的 ECU 的系统
- 与 ECU 建立联系
- 选择 ECU/ID Block 菜单项或者使用组合键 ALT+I
- 如果此处并未显示您需要的信息，可以采用系统配置向 ID 块中添加测量结果。

注：

并非所有 ECU 均支持 ID 块中理论上可以显示的数据。还应当注意，采用 1A 86 以及 1A 87 读取的 ID 块之间的区别。

2Q 1850971v

3.7 读取并且清空 ECU 错误存储器

- 启动 Ecoule
- 打开包括关注的 ECU 的系统
- 与 ECU 建立联系
- 选择 Error/...菜单项，具体取决于是否需要读取或者清空

注：

您需要针对读取错误以及在错误窗口中清空错误而设置参数。

2Q 1850971v

3.8 执行快速测试

注意:

在执行快速测试之前，需要快速测试数据。可以从 Vediamo 主页下载此类数据。

- 启动 Ecoute
- 选择 Error/Quicktest 菜单项, 或者从 StartCenter 中采用参数“Quicktest”启动 Ecoute, 或者采用指令行参数 -k 启动 Ecoute
- 配置快速测试并且执行快速测试。

3.9 执行变体编码

- 启动 Ecoule
- 载入包括需要编码 ECU 的系统
- 建立与 ECU 的联系
- 选择 Coding / Variant coding 菜单项
- 如果在系统说明中正确输入了所有前置条件，则在打开 Varcode 窗口的时候会自动执行此类条件。否则，可以双击选择窗口中的相关位置手动执行必要的服务(比如，发布、编程模式等)。
- 按照上述步骤进行操作。

3.10 执行 Java 程序 (Java 程序)

存储 Java 程序

可以在任何文件夹中保存 Java 程序。为了更便于查找，建议将其保存到 VedioamDaten 子文件夹中。

最好将对应于特定 ECU 的 Java 程序,比如 CR4 保存到 VedioamDaten\CR4 文件夹中,将对应于各种 ECU 的 Java 程序保存到 VedioamDaten\AllgemeineScripte 文件夹中。必须总是在 Vedioam.ini 的[INTERPRETER]ClassPath 中指定 Java 程序的数据路径,从而可以确保 Java 解释器能够找到该程序。务必多加小心,确保不同版本的 Java 程序(相同的程序名称)应当处于不同的文件夹中,因为无法保证执行预期版本的程序。

从 Ecoute 或者 PSR 中执行:

为了从 Vedioam 应用程序 (Ecoute 或者 PSR) 中执行 Java 程序 (即,显示该程序),则必须通过系统配置方式在 VSB 中输入程序的名称。

系统窗口 routines 中的 Ecoute 中显示了 Java 程序,而且双击即可启动该程序。PSR 适配器中启动程序主要使用了 PSR 的“600, routine name”指令。

采用指令行执行

必须在批处理文件或者指令行中输入下列内容:

一般:

```
javaw -classpath <JavaKlassenPfad_1>;<JavaKlassenPfad_2>;...; <routine name>
```

示例:

```
"C:\Program Files\Vedioam\JRE\bin\javaw" -classpath
```

```
"C:\ProgramData\Vedioam\VedioamDaten\AllgemeineScripte";"C:\Program Files\Vedioam\Java";
```

```
ExampleRoutine
```

3.11 更改 K-线路与 CAN 之间的连接

- 如上所述建立与 ECU 的连接
- 启动 Ecoute
- 载入包含关注的 ECU 的系统
- 从 ECU 上下方菜单中选择 Properties 菜单项（右键点击 ECU）
- 从弹出的对话框中选择预期的连接，必要的情况下，断开当时不需要的 ECU 连接。
- 建立与 ECU 的联系。

如果预期的连接不可用，可能是由于存在以下原因：

- 无法通过本连接与 ECU 通讯。物理连接的存在并不意味着 ECU 可以提供此类连接。如果 ECU 提供此类连接，那么并不意味着可以对其进行参数化。关键词参数化，CBF。
- 无法实现载入的固件的协议。

关键词 CaesarGo ↔ TLSlave 固件。

2018509714

3.12 在我关闭的相同状态下打开 Ecout

- 启动 Ecout
- 按照希望下次启动的方式配置 Ecout（系统、窗口位置等）
- 选择 System / Save session as 菜单项保存话路。
- 选择 Extras / Options 菜单项打开 Ecout 选项。选择 Start 页面，然后激活 Load last session 选项。
或者还可以：
 - 如果仅需重新载入上次的系统，则选择 Extras/Options 菜单项。前往 Start 页面，然后激活 Load last system 选项。不会保存窗口位置。
 - 采用指令行的方式将话路文件的名称传送给 Ecout。
 - 针对 StartCenter 创建启动档案，采用刚才用指令行方式传送给 Ecout 的保存话路。

4、词汇表

术语	解释
3964R	PSR 与 Vediamo 的串行连接专用的连接协议。
区域	需要闪存处理的 ECU 存储器部分。
CAN	控制器局域网。 用于车辆内 ECU 以及车辆与测试仪之间的通讯。
CAESAR	汽车电子系统的通用访问需求。CAESAR 包括软件以及硬件组件。
CBF	CAESAR 二进制格式 – 具有 ECU 说明数据的二进制格式。
CCF	CAESAR 编码文件 – 具有变体编码数据的二进制格式。
CFF	CAESAR 闪存文件 – 具有闪存数据的二进制格式。可以包括多个区域/闪存键。
配置文件	Vediamo.ini（详情参见下文）
诊断针	ECU 与测试仪之间的物理 K 线路针（比如位于 CAESAR Part C）
DiagService	诊断服务 – 与 ECU 的象征性通讯，比如，读取测量结果
DiagJob	诊断作业 – CAESAR 解释的操作步骤，可能包括 DiagServices 等
DIOGENES	诊断数据输入以及采用 SGML 的数据管理
DCOM	分布式通用对象模型 – 微软专门用于分布式应用程序编程的模型。
ECU	电子控制装置
闪存键	识别需要闪存处理内容的唯一键
闪存件	永久性存储在 ECU 中的软件
网关	ECU 网关允许访问与之相连的 ECU 组件。很多情况下，网关采用 K 线路而且允许访问 CAN 总线，此类总线是 ECU 组件以及网关自身与之相连的。
GBF	GPD 二进制格式 – 二进制格式包括预编译的 GPD
GPD	通用协议说明 – 通讯协议的通用协议说明语言
JNI	Java 原生接口 – 将 Java 代码与 Win32 代码相连的接口（比如，用于 C++ 编程 DLL 中）。用于 VHandlers.dll 的情况下，确保 Vediamo 类对于 Java 程序是“可见的”。
Java	Sun Microsystems 面向对象的编程语言。用于 Vediamo 的情况下主要用于自动化程序。
句柄函数	亦称：内置函数。 永久性“集成的”Vediamo-Java 接口函数（VHandlers.dll）
ISOSCAN	ECU K 线路诊断专用的应用程序
K 线路	与 ECU 的串行连接
日志文件	包括日志信息的文件。详情参见日志记载。
日志记载	亦称：跟踪。 用于调试的程序序列记录（比如，调用的函数名称等）。
LUCA	Langner 通用通讯 API – 通讯协议专用软件。用于 PSR 客户端。

含义	需要闪存处理的特定区域的内容。
组件	仅可通过网关 ECU 诊断的 ECU 组件（详情参见上文）。
MPF	发动机测试设施（Motorenprüffeld）
MIL	故障指示灯 – ECU 错误状态比特位
名称	某种服务的语言化名称（比如，“RPM”）。不具有唯一性。
OutputRef	从 ECU 中获取另外一项服务（比如，驱动器）设定数值的服务。
准备	服务的输入参数。
展示	从 ECU 回复中解释的结果。
前置条件	调用的服务，其中执行时 ECU 参数化中另外一个动作的前置条件。 与之相对的是，用户可以指定前置条件，而且无需 Vediamo 文件中的 CBF 规范（VSB、MWG、STG），而且可以引用 Java 程序与诊断服务。
程序	亦称：Java 程序，自动化测试程序，Java 程序，Java 客户端。 可以采用 Vhandlers 接口访问 DiagServer 的诊断功能的一种 Java 程序。详情参见 Java 程序的相关章节。
PSR	测试工作台控制器（Prüfstandrechner）-控制发动机测试工作台。可以采用 PSR 客户端从 Vediamo 获取诊断数据。
PSR 协议	串行以及 TPC/IP 连接专用协议。现有的协议是出于 PSR 与诊断计算机之间的。本协议自二十世纪九十年代初就已经应用于 MPF 程序（仅串行连接）。
QDE	质量数据采集计算机（Qualitätsdatenerfassungsrechner）
修饰符	服务的唯一 ID（比如，“DT_RPM”）。
脚本	Java 的原来称谓（之前被称为“功能链”）。在 Vediamo 中，自动化程序均采用 Java 编写。详情参见 Java 程序相关章节。
西门子协议 3964R	PSR 与诊断服务器之间的串行协议，用于 COMx。
标准服务/标准对象	在 DIOGENES 中采用不同命名方式的某种服务。因此在 Vediamo 中可以采用固定的标准名称（比如，Bandend、Unlock、Initialize），以便确保 ID 的唯一性。
系统	更准确地说应当称为：ECU 系统。 一个或者多个 ECU 的组合，被称为 Vediamo 中的系统，比如具有两个 ECU 的马达即为一种系统。
TL 从	在 CAESAR 主中执行的 CAESAR 固件
监控	日志记载测试仪与 ECU 之间的诊断协议。
Vediamo	发动机专用的分布式诊断应用程序（Verteilte Diagnose Anwendung für Motoren）
Vediamo.ini	Vediamo 模块专用的配置文件。在该文件中设置 Vediamo 系统所需的所有参数。按照模块对各项进行分类。某些设置由程序自动进行存储流程图仅可由用户采用 INI 编辑器进行编辑。
VOM	Vediamo 对象模型-发动机测试设施相关的对象模型，用于诊断 ECU。
VND 文件（vnd）	Vediamo 消息说明文件-包括手动指令输入的相关消息。
VSB 文件（vsb）	Vediamo 系统说明文件-包括 ECU 系统的相关信息。采用系统配置进行创建，采用 DiagServer 进行载入。

5、INI 参数

通用		
名称	说明	备注
Language	本项规定了应用程序的语言。先决条件是可以获取资源文件的相应版本。采用缩写方式（EN=英语，FR=法语，SP=西班牙语等）指定语言。针对每个应用程序的每种语言创建一个 DLL 文件。默认语言为德语。 DiagServer 的翻译文本包括文件 VCommon_Res_XY.dll，其中 XY 是语言的缩写。其余应用程序使用的文件，其名称是由相应的应用程序名称构成的：[应用程序名称]_Res_XY.dll，比如，Werker_Client_Res_EN.dll。	默认数值： DE 可选数值： DE 德语 EN 英语
黑盒		
名称	说明	备注
LogFile	程序日志文件的路径以及主要文件名称。一旦发生错误或者用户发布显式指令，则所有 Vediamo 程序日志输出均由黑盒进行缓存，并且仅可保存到此处指定的文件中。保存文件的时候，当前数据以及时间都会自动附加到文件名中。此外，还会添加扩展名“.log”。 默认数值为（该项为空的情况下）：“\BLACKBOX”，因此日志文件会存储在程序文件 BlackBoxServer.exe 所在的文件夹中。	默认数值： ...\Vediamo\Log\BlackBox
LineNum	需要可存储在环形缓存中的行数。对于数量较大的运行 Vediamo 程序而言，以及对于更为复杂的动作而言，该数值应当更大一些。 每一行使用 150 个字节的主要 PC 存储器。 默认数值：5000，足以用于日常使用。 最小数值：500，输入更小的数值则会使用 500 行。	默认设置： 5000 最小数值： 500
通用		
名称	说明	备注
StatusTraceToFile	该项规定了是否应当将 Ecout 输出窗口的内容保存到某个文件中。输出存储在“<[ECOUTE]ProgramLogPath>\Status.log”之中。	默认设置： 0
ProgramLogPath	该项规定了 ECU 状态日志文件保存的文件夹（状态窗口中的输出）。	默认设置： ...\Vediamo\Log\Client
TraceFileMaxSize	该项规定了日志文件的最大的大小（以字节为单位）。 默认设置为 524288（1/2 兆）。 0 值表示大小不受限制。	默认设置： 524288 最小数值： 0
DCDiChannelLogPath	该项规定了保存 Ecout ECU 相关日志文件所需的文件夹。	默认设置： ...\Vediamo\Log\Ecout
KlineTraceMode	该项规定了是否应当将 ECU 通讯按块、详情或者不采用上述两种标准显示在跟踪窗口中。 输出还会保存在“ECOUTE]DCDiChannelLogPath>\<system>\<ECU>Trace.log”中。 默认设置是不显示。	默认设置： 0 可选数值： 0 不显示 1 显示块 2 显示详情（字节以及时间） 3 显示块以及详情
ShortTestDataDir	该项规定了快速测试文件夹。快速测试的数据必须在该文件夹的子文件夹中提供，而且对应于不同的型号。 默认设置为“\Data\VediamoKurztestDaten”。	默认设置： ...\Vediamo\VediamoShorttestData
LoadLastSession	该项规定了在 Ecout 启动的时候是否应当载入上次保存的话路文件。	默认设置： 0
VSCPATH	该项规定了上次保存的话路文件的路径以及文件名。	
AutoInitOnSystemSelection	该项规定了后续系统变更的时候，是否应当尝试自动与新系统的所有 ECU 建立联系。	默认设置： 0
AutoInitOnContactLost	该项规定了后续丢失联系的时候，是否应当循环尝试重新与系统 ECU 建立联系。 默认设置为否。	默认设置： 0
AutoInitOnFailedUserInit	该项规定了如果用户未能建立联系，是否应当循环尝试建立联系。 默认设置为否。	默认设置： 0
ExecuteInitSequence	该项规定了是否应当执行初始化程序。默认设置为不执行程序。	默认设置： 0 可选数值： 0 并执行初始化程序 1 选择系统之后执行初始化程序 2 建立联系之后执行初始化程序 3 选择系统以及建立联系之后执行初始化程序
SnapshotFormat	该项规定了保存窗口内容的文件格式（HTML 或者文本）。 该项不适用于采用循环次数>0 的情况下循环保存测量结果（总是采用 CSV）。	默认设置： html 可选数值：

	默认设置为 HTML 格式。	文本 文本格式 html HTML 格式
VarCodStringFormat	该项规定了编码字符串格式。 默认设置为十进制格式。	默认设置： 十进制 可选数值： 十进制 十进制 十六进制 十六进制
TreeStyle	系统服务以及程序在系统窗口中显示为树形控件。树形控件的各个分支采用 Vediamo 中规定的类别进行分类，比如，测量、驱动器、功能等。或者，还可以根据 DIOGENES 中参数化的类别对服务进行分类。 这种情况下，系统窗口中的服务处于未过滤状态（即，会显示所有参数化的服务，而不仅仅是 VSB 文件中规定的服务）。 默认设置为 Vediamo 系统说明	默认设置： VSB 可选数值： VSB 作为 Vediamo 系统说明 DIO 作为 DIOGENES
UseFilters	该项规定了 VSB 中规定的过滤器是否适用于诊断服务器。 默认设置为是。	默认设置： 1
GlobalFilter	根据前缀过滤某些服务。一旦创建了系统说明，则具有此处规定前缀的服务显示为过滤部分。系统配置中一般会禁用过滤。Ecout 中不会显示具有规定前缀的服务。例外情况是已经采用配置工具禁用的过滤。 如果使用不具有.vsb 的 Ecout（即，根据 DIOGENES 参数化（默认.vsb）的系统说明），则此处规定的前缀同样适用。 示例： GlobalFilter = DNU_, WVC_, SES_, NR_, RVC_, SRC_	默认设置： DNU_, WVC_, SES_, NR_, RVC_, SRC_
DisplayQualifiers	该项规定了树形控件中是否显示名称或者修饰符。	默认设置： 1 可选数值： 0 名称 1 修饰符
GraphMaxPoints	该项规定了测量曲线可能拥有的测量点的最大数量。 默认设置为 1000。	默认设置： 1000 最小数值： 100 最大数值： 10000
ShowDefaultSystems	该项规定了系统选择显示中是否应当包括 DIOGENES 系统参数化专用的系统（CBF）。 默认设置为否。	默认设置： 0
ShowVSBSystems	该项规定了系统选择显示中是否应当包括 Vediamo 系统说明中的系统。 默认设置为是。	默认设置： 1
ShowVariantDetectionDialog	该项规定了如果在建立联系的时候仅识别了 ECU 的基本变体，那么是否应当显示自动变体识别的对话框。 默认设置为是。	默认设置： 1
EcuExitInitAfterFlashing	该项规定了如果在闪存处理之后并未执行关机循环，那么是否应当自动结束与 ECU 的联系并且重新建立联系（ECU 复位）。 默认设置为是。	默认设置： 1
服务器		
名称	说明	备注
SystemPfad	该项规定了诊断服务器可以在哪个文件夹中查找应当使用的系统说明。此外，还会在该路径中查找 Vediamo 脚本（详情参见 [INTERPRETER] 部分中的 ClassPath）。有效的项是具有驱动器以及文件夹的正常路径规范。 默认设置为“.”（当前文件夹）。	默认数值： .\
SystemConsistencyCheck	该项规定了读取系统说明文件的时候是否需要一致性检查。此项检查可以确定创建 VSB 文件之后相关的 CBF 或者脚本文件是否已经发生了变更。默认情况下，不会进行一致性检查。	默认设置： 0 可选数值（采用逻辑 AND 运算符对数值进行组合）： 0 不进行一致性检查 1 针对 CBF 文件进行一致性检查 2 针对 Java 程序进行一致性检查 3 针对 GBF 文件进行一致性检查（仅在错误的时候创建日志项）
BrokerTraceToFile	该项规定了是否在跟踪文件中日志记载在诊断服务器与其客户端之间采用代理级别执行的指令。 标准设置为否。	默认设置： 0
Name	该项规定了正在运行的诊断服务器过程以及 Ecout 应当与之通讯的计算机的名称。如果没有指定名称或者名称为空的字符串（Name=），则按稳定正在同一台计算机上运行诊断服务器。可以采用 UNC 命名方式（比如，\\Server）、DNS 命名方式（比如，Server.com），或者采用 IP 地址（比如，123.45.67.89）输入名称。如果计算机处于相同的域名，则仅需输入	

	名称（比如，Name=August）。 注意，本 INI 文件中的服务器以及 CAESAR 设置不适用于与客户端处于同一台计算机上运行的诊断服务器过程（Ecout/PSRClient/...）。	
ProgramLogPath	该项规定了 CAESAR 日志文件存储的文件夹（详情参见[CAESAR] Debuglevel）。	默认设置： ...\Vediamo\Log\Server
DCDIChannelLogPath	该项规定了 ECU 相关服务器日志文件应当保存的文件夹。	默认设置： ...\Vediamo\Log\Server
Simulation	Vediamo 服务器共有三种工作模式：正常工作、仿真以及 SIM 记录。默认设置为正常工作。	默认设置： 0 可选数值： 0 正常工作 1 仿真 2 记录仿真数据
KLineTraceMode	该项规定了是否应当按块、按详情或者不按两者方式来日志记载 ECU 通讯（K 线路以及 CAN）。 输出保存到 “<[SERVER]DCDIChannelLogPath>\<System>\<ECU>Trace.log”之中。 默认设置为日志记载。	默认设置： 0 可选数值： 0 不日志记载 1 按块日志记载 2 按详情日志记载 3 按块以及详情日志记载
LoadCFFs	该项控制服务器初始化时载入的 CFF 文件以及合适调用 LoadCFFs 函数。	0=载入所有 CFF 1=不载入任何 CFF 2=必要的情况下，从 ECU 子文件夹中载入 CFF 3=必要的情况下，从标准列表中载入 CFF
OBDDProtocol	protocol.gbf 中定义的一个协议，用于 OBD 通讯	KW2C3PE
OBDBaudrate	OBD 通讯专用的 CAN 波特率	50000
OBDServiceFile	该文件定义了所有服务、PID、TID、MID 等。 其中还包括了对于某个文件的引用，该文件定义了所有错误代码（DTC）。两个文件均必须位于同一个文件夹中。	.\OBD\OBD2.xml
OBDP2_MAX	一般来说，在协议中定义 P2 即可。如果出现计时问题（可能与 eCOM 有关），则采用更高的数值可能使得通讯更稳定。	250
OBDDREQREPCOUNT	一般来说，每个消息仅会发送一次。如果发生计时问题（可能与 eCOM 有关），则采用更高的数值可能使得通讯更稳定。	1
CAESAR		
名称	说明	备注
LANGUAGE	该项规定了 CAESAR 生成的文本的语言（错误消息、服务名称）。先决条件是提供适当的语言文件（*.CTF）。采用缩写词表示语言（比如，EN=英语，FR=法语，SP=西班牙语等）。	
CBFPFAD	该项规定了 CAESAR 查找 CBF 文件所在的文件夹。有效的项是具有驱动器以及文件夹的正常路径规范。 默认设置为“.”（当前文件夹）。	默认设置： .\
GBFPFAD	该项规定了 CAESAR 查找 GBF 文件所在的文件夹。有效的项是具有驱动器以及文件夹的正常路径规范。 默认设置为“.”（当前文件夹）。	默认设置： .\
DRIVERPFAD	该项规定了 CAESAR 查找驱动器文件所在的文件夹。有效的项是具有驱动器以及文件夹的正常路径规范。 默认设置为“.”（当前文件夹）。	默认设置： .\
CFFPFAD	该项规定了 CAESAR 查找 CFF 文件所在的文件夹。有效的项是具有驱动器以及文件夹的正常路径规范。 默认设置为“.”（当前文件夹）。	默认设置： .\
CTFPFAD	该项规定了 CAESAR 查找 CTF 文件所在的文件夹。有效的项是具有驱动器以及文件夹的正常路径规范。 默认设置为“.”（当前文件夹）。	默认设置： .\
CCFPFAD	该项规定了 CAESAR 查找 CCF 文件所在的文件夹。有效的项是具有驱动器以及文件夹的正常路径规范。 默认设置为“.”（当前文件夹）。	默认设置： .\
DEBUGLEVEL	该项规定了 CAESAR 动作在文件 “<[SERVER]ProgramLogPath>\CAESAR.log”中日志记载的详细程度。总是会生成此类文件！	默认设置： 1 可选数值： 0 无输出 1 模块相关 2 功能相关 3 内部功能详情

		4 最大
CHANNEL DEBUGLEVEL	该项规定了日志记载通道相关的 CAESAR 动作的详细程度。针对每个通道均存在一个日志文件 “<[SERVER]DCDICChannelLogPath>\<System>\<ECU>Kanal.log”。 采用 ECU 可以打开或者关闭日志记载。	默认设置： 1 可选数值： 0 无输出 1 模块相关 2 功能相关 3 内部功能详情 4 最大
TraceDebug Output	该项规定了 CAESAR 生成的日志信息（DebugOutput 以及 ChannelDebugOutput）是否显示在 BlackBoxViewer 窗口中。 增强调试，比如，开发诊断作业的时候...（详情参见 DEBUGLEVEL 以及 CHANNELDEBUGLEVEL 各项） 默认设置为否。	默认设置： 0
USE_SISerialDriver	该项规定了是否应当载入 CAESAR Part D 专用的驱动器。 默认设置为是。	默认设置： 1
USE_SIPartC Driver	该项规定了是否应当载入 CAESAR Part C 专用的驱动器。 默认设置为否。	默认设置： 0
USE_SIPC MCIADriver	该项规定了是否应当载入 CAESAR Part A 专用的驱动器。 默认设置为否。	默认设置： 0
USE_SIPartX Driver	该项规定了是否应当载入 CAESAR Part X 专用的驱动器。 默认设置为否。 此外，针对操作 CAESAR Part X 而使用驱动器文件夹中 CAESAR 自有的 slave.ini 文件中的设置。 默认设置为否。	默认设置： 0
USE_SIPartY Driver	该项规定了是否应当载入 CAESAR Part Y 专用的驱动器。 默认设置为否。	默认设置： 0
USE_SIPartJ Driver	该项规定了是否应当载入 CAESAR Part J（传送车辆通讯接口）专用的驱动器。 默认设置为否。	默认设置： 0
USE_SIPartE Driver	该项规定了是否应当载入 CAESAR Part E 专用的驱动器。 默认设置取决于安装过程中选择的诊断服务器子组件。 默认设置为否。	默认设置： 0
PinMapping	该项规定了在使用 CAESAR Part E 的时候，是否应当激活 PIN 映射。 如果设置在诊断服务器安装中选择子组件中的 CAESAR Part E，则激活 PIN 映射。	默认设置： 0
GPDFlashCaching	针对 CAESAR 从中的 GPD 激活缓存机制：如果闪存 EEPROM 中并不包括所需的 GPD，则将其写入闪存 EEPROM 中。 从闪存中执行 EEPROM 中存储的 GPD。 如果 GPDFlashCaching 为 0，则总是在 CAESAR 从的 RAM 中执行 GPD。	默认设置： 1
BootNewFirmware	对于 CAESAR 发布 2.8 版本而言，提供了一种替代性新的固件，使用该固件时，GPD 在 CAESAR 主上运行。截至目前，该固件支持 KW2000 协议。新的固件具有以下几个优点： - CAN 可用通道数量：40/背负式 - 数据吞吐量增加，比如，明显加快了闪存处理过程 使用的 CAESAR 硬件必须采用新的固件进行引导，以便激活该特性。通过逗号来分隔硬件，采用新的固件来引导指定的硬件。 下列设置导致采用新的固件来引导第一以及第四个硬件（在列表中采用小于 100 的单独编号进行指定）：BootNewFirmware=0,3 如果固件应当适用于所有部件，则仅需在列表中输入“All”。 如果应当使用原有的固件，则列表必须为空。 默认设置为全部。	默认设置： 全部
BootBusSimFirmware	作为 2.8.0 版，CAESAR 提供了总线仿真所需的功能。该功能可以用于 VEDIAMO 乃至 Java 程序。为了使用总线仿真，必须采用特殊的固件引导相应的 CAESAR 硬件。为了采用总线仿真固件选择性引导可用的 CAESAR 硬件，应当插入 BootBusSimFirmware 键： - BootBusSimFirmware=ALL 在采用总线仿真固件启动 CAESAR（ComConstruct）的时候，引导所有识别的 CAESAR “卡”。 - BootBusSimFirmware=0,2 为了允许选择性引导单个 CAESAR “卡”，适用下列规则：如果在键中规定了一个或者多个大于 100 的数字，那么将其解释为连续的卡编号。在上例中，采用总线仿真固件引导在引导期间识别的第一以及第三 CAESAR 卡。 上述规则同样适用于 BootNewFirmware 键，后者可以用于控制采用“新的”固件进行的引导。如果与规范发生冲突，BootNewFirmware 比 BootBusSimFirmware 具有更高的优先级，比如，如果	默认设置

	BootNewFirmware=0,1 而且 BootBusSimFirmware=0,1, 则会采用“新的”固件引导前两个卡。	
DefaultDeviceNumber	该项规定了默认的器械编号。	默认设置: 0
DefaultDiagPin	该项规定了默认的诊断针。	默认设置: 0
MasterSlaveTimeout	如果在其它事项中打开了通讯通道,那么会将参数传送至 CAESAR, 指定在 CAESAR 生成超时错误之前, CAESAR 主与从之间可以禁用的通讯时间长度。在此可以更改该数值(单位为毫秒)。默认设置为 15000 毫秒。	默认设置: 15000
MonitoringFilterCANIDFile	该项规定了包括跟踪显示的过滤器设置的文件。	
MonitoringFilterCANIDs	该项规定了是否过滤跟踪现场(详情参见 MonitoringFilterCANIDFile)。默认设置为否。	默认设置: 0
UseDriverTypes	采用 CAESAR 发布 2.6 版本的情况下,可以针对与 ECU 的通讯而在参数化中包括多个 GPD 引用。采用驱动器类型代码指定此类 GPD 引用。如果需要载入默认系统或者更早版本系统说明,则该项决定了最好采用何种驱动器类型代码应用程序。 下列数值有效: - NULL - KLINE - CANLS - CANHS - D2B - J1708 - MOST - J1850 - CCD - SCIENG - SCITRANS 可以指定一个以上的数值。可以定义与其它数值相比更偏好的使用的某些数值的次序。需要采用上述格式输入数值,并且采用逗号分隔。会忽略未列出的关键词。在启动诊断服务器的时候,会读取整个项。如果包括了多个 GPD 引用,则在访问资源以及 ECU 信息的时候,会保持偏好 GPD 引用的规定序列: 示例: UseDriverTypes = KLINE,NULL,CANLS,CANHS,D2B	
UseServiceTypes	该项可以对需要显示的服务进行预过滤,比如,生成系统说明的时候。如果设置了“STANDARD”,则不会显示下列不可执行的服务类型。DST_SYSTEM, DST_ENVIRONMENT_DATA, DST_GLOBAL, DST_NEGRESP, DST_BINARY_ACTUATOR_INP, DST_BINARY_ADJUSTMENT_INP。	默认设置: STANDARD 可选数值: STANDARD 不显示某些不可执行的服务 ALL 显示所有服务
TesterPresentInfo	针对更新通讯状态而控制通道相关的增强功能。对于某些 ECU/协议而言, Vedomo 诊断服务器可能必须循环向 ECU 发送消息,以便更新通讯状态。还可以采用该项配置功能。可以指定下列各项(采用列表格式): -特性应当激活的准确的协议 ID -采用十六进制格式作为“tester present”发生的消息 采用逗号分隔的单个项。 如果 Part I 用于相应通道诊断,那么会针对该通道根据类别禁用特性。总是在打开通道之后评估当前配置信息。 示例-2 个协议的设置受到影响: TesterPresentInfo=KW2C3PE,3e01,KW2C2PE,3e01	
通用		
名称	说明	备注
ClassPath	该项规定了 Java 徐南昌查找 Java 类的搜索路径。可以采用分号分隔多个文件夹。默认设置为当前工作文件夹(.\)。还可以搜索[SERVER]部分中 SystemPfad 下规定的路径。	
VHandlersLib	该项规定了针对脚本中 Vedomo 句柄函数使用的库(DLL)。默认设置为 VHandlers.dll。	
JavaInterpreter	该项规定了 Java 解释器。对于没有控制台窗口的 Sun Java Runtime Environment (JRE)而言,应当是 javaw.exe,对于有控制台窗口的 Sun JRE 而言,应当是 java.exe。相应的 JRE 安装路径中均包括了上述各种文件。	
UseJVHandlersPackage	采用两种方式提供了 Vedomo 的自有 Java 类:在“未命名包”中,或者在“JVHandlers 包”中。默认设置为“未命名包”。该项规定了是否应当使用“JVHandlers 包”。	默认设置: 0

PSR		
名称	说明	备注
COMPORT	该项规定了应当针对与 PSR 通讯使用何种接口。对于采用串行连接的通讯而言，有效数值为数字 0 至 n，其中 n 表示可用的串行接口。超过 1024 的数值解释为 TCP/IP 通讯的端口编号。数值 2049 保留用于 TCP/IP 通讯	默认数值： 1 最小数值： 0
EngineTable	该项规定了发动机表包括的文件。安装 Vediamo 的时候，生成并且规定了样本文件。	
PreloadSystem	在该项中，可以输入 Vediamo 系统的名称。如果命名了某个系统，则在启动 PSR 客户端的时候，会载入系统文件（VSB）。在这种情况下，第一次发动机测试运行无需等待开始载入 VSB 文件。 注意：系统名称大小写敏感。	
COMTIMEOUT	该项规定了与测试工作台通讯的超时，单位为毫秒。如果在此期间没有发生与测试工作台的通讯，则会结束测试运行以及与 ECU 的通讯，而且清空传送缓冲。 数值 0 表示没有超时（默认设置）。	默认设置： 0 最小数值： 0
LOGFILE	该项规定了采用哪个文件来日志记载字节级别的通讯（采用时间戳）。默认设置为空，而且意味着不会出现日志记载。正常工作情况下该项应当为空。	
通用		
名称	说明	备注
ProgramLogPath	该项规定了用于保存系统配置工具日志文件的文件夹。	默认数值： ...\Vediamo \\Log\SystemConfiguration
LogWin	该项规定了在启动应用程序的时候，是否应当在窗口内日志记载应用程序的动作。默认设置为是。	默认设置： 1
OutputLevel	该项规定了日志记载 CAESAR 动作的详细程度。消息保存在：“SYSTEMCONFIGURATION\ProgramLogPath>\VediamoSysConfLog.txt”。默认设置为模块相关。	默认设置： 1 可选数值： 0 无输出 1 模块相关 2 功能相关 3 内部功能详情 4 最大
LogFileMaxSize	该项规定了系统配置文件夹中的 VediamoSysConfLog.txt 日志文件中的最大大小（单位为字节）。 初始设置为 1000000（大约为 1 兆）。最小限值为 10000 字节。	默认设置： 1000000 最小数值： 10000
工人		
名称	说明	备注
Server	上次与程序连接的服务器的名称。不会自动进行连接，但是点击 F5 的时候，向用户推荐该服务器名称。可以承认或者覆盖。该项自动设置为服务器（与其建立连接）的名称。	
LogFile	记录日志窗口项的文件。如果该项遗失或者为空，则不会自动记录数据。在结束程序的时候，会依照选项窗口中的设置自动更新该项。	
Warnings	规定是否应当在工人客户端状态窗口中显示意外状况的警告（不会导致结束某个操作）。默认设置为否。 仅可直接采用 INI 编辑器更改该参数，但是不会使用工人客户端选项窗口。	默认设置： 0
私有		
名称	说明	备注
IncrementalBackup	该项规定了是否应当采用持续增加数字附加到后续的状态日志文件中（Status1.log, Status2.log, ...）。	默认数值： 0
TraceHandlerFunctions	Java 函数调用专用的补充设置。执行 Java 程序的时候简化调试。	默认设置： 0
TraceThreadIDToFile	日志文件中补充的线程 ID 项。简化用户编程调试。	默认设置： 0
UVI		
名称	说明	备注
SYSTEM	如果初始化规范并未包括其参数列表中的系统名称，则 UVI 载入默认系统。	

6、PSR 消息

从	至	解释	代码	数据	备注
PSR	DiagServer	新发动机	3	发动机 ID (S)	为每台发动机指定系统文件以及(可选)初始化程序。
DiagServer	PSR	载入的数据	5	系统修饰符 (S)	(VSB 文件的)系统 ID
DiagServer	PSR	未知发动机类型	9	错误消息 (S)	错误原因
PSR	DiagServer	获取 CAESAR 版本	10	-	传送 c32s.dll 的版本
DiagServer	PSR	CAESAR 版本	11	版本 (S)	比如, “02.07.12”
PSR	DiagServer	从 ECU 获取 ID 块	20	-	必须与 ECU 建立联系
DiagServer	PSR	从 ECU 获取 ID 块	21	ID 块 (S)	具有回车行的长字符串
DiagServer	PSR	ID 块错误	22	错误文本 (S)	
DiagServer	PSR	错误消息	50	作为字符串的消息 (S)	DiagServer 中的错误, 比如, 内部程序错误、错误的配置、硬件错误
PSR	DiagServer	与 ECU 建立联系	100	-	程序启动循环激活过程。顺利建立联系之前不会回复。
DiagServer	PSR	建立联系	101	ECU 名称 (S)	
DiagServer	PSR	结束联系	102	ECU 名称 (S)	结束联系而非无法建立联系的时候发送。例外: 如果没有载入系统, 则采用 102 回复 100, 而且没有 ECU 名称。
DiagServer	PSR	测量信息	109	数字 (W) 修饰符 (S) 单位 (空 S) (指令 113 之后)	传送所有测量的列表, 从 1 开始连续编号。编号为 0 的最后一个块以及空名称意味着列表结束。由于 CAESAR 硬件在此时无法识别, 因此作为空字符串传送该单位。但是, 现存的 PSR 程序需要字符串参数。
DiagServer	PSR	清空错误	110	-	
DiagServer	PSR	无法清空错误	111	错误代码 (W)	1=遗失诊断消息 5=无法清空错误
DiagServer	PSR	完成初始化	112	-	
PSR	DiagServer	传送测量修饰符列表	113	-	请求所有测量修饰符列表(服务 109)。 注意! 已经建立连接之后, 可以通过从当前 ECU 变体的额外设置附加测量列表。
PSR	DiagServer	协议信息	150	-	请求 ECU 协议 ID
DiagServer	PSR	协议信息	151	协议名称 (S)	比如, “KWP2000E”
PSR	DiagServer	发送测量	313	-	请求所有测量名称列

		名称列表			表（服务 109）。 注意！ 已经建立连接之后，可以通过从当前 ECU 变体的额外设置附加测量列表。
DiagServer	PSR	学习了节流阀	114	-	
DiagServer	PSR	学习了节流阀	115	-	
DiagServer	PSR	无法读取错误	200	错误文本（S）	
DiagServer	PSR	错误	201	错误代码（S）或者（详情参见是否为 235-238）状态（W）错误代码（S）名称（S）	如果不存在任何错误，则传送两个空的字符串，或者在错误列表末尾进行标记。
PSR	DiagServer	清空所有错误	202	-	
PSR	DiagServer	开始错误读取	203	-	服务器采用 201 块循环回复。默认情况下，在测试运行期间首次初始化 ECU 之后从 ECU 读取错误。
PSR	DiagServer	结束错误读取	204	-	还可以在采用指令 100 初始化之前而且仅在采用指令 3 载入系统之后发送。这种情况下，建立联系之后不会读取任何错误。
DiagServer	PSR	错误环境数据	211	错误代码（S） 环境数据（S）	环境数据包括名称、数值以及可能的单位，采用空格分隔。
PSR	DiagServer	开始读取具有环境数据的错误	213	-	服务器在 211 块之后采用 201 块循环回复。
PSR	DiagServer	读取一次错误	230	可选项：可以采用 WORD（0..4）或者下列任一字符串选择错误类型： POWERTRAIN (=0) CHASSIS (=1) BODY (=2) NETWORK (=3) UNDEFINED (=4) 默认：all (=0xFF)	服务器采用 201 块回复
PSR	DiagServer	读取一次错误，未过滤	231		服务器采用 201 块回复，随后是 211 块
PSR	DiagServer	读取一次错误，采用环境数据	232		服务器采用 201 块回复，随后是 211 块
PSR	DiagServer	读取一次错误，采用环境数据，未过滤	233		服务器采用 201 块回复，随后是 211 块
PSR	DiagServer	读取一次错误，采用状态字节	235		服务器采用 230-233 进行回复，但是 201 回复块从 WORD 数值（错误状态）开始，随后是 P 代码以及错误文本。211 块（环境数据）保
PSR	DiagServer	读取一次错误，采	236		

		用状态字节			持不变。
PSR	DiagServer	读取一次错误，采用环境数据，采用状态字节	237		
PSR	DiagServer	读取一次错误，采用环境数据，未过滤，采用状态字节			
DiagServer	PSR	无法读取测量结果	300	错误文本 (S)	
DiagServer	PSR	测量	301	数字 (W) 数值 (S)	
PSR	DiagServer	开始传送数据	302	-	DiagServer 持续传送数据 (测量)。如果在启动测试之后并未传送服务 304，则会传送 VSB 文件中定义的所有测量结果。
PSR	DiagServer	结束传送数据	303	-	DiagServer 结束持续传送数据。
PSR	DiagServer	选择测量数值	304	测量次数 (words 列表)	如果正在传送，将其取消。此后必须采用服务 302 再次请求测量，或者采用 305 读取一次。
PSR	DiagServer	读取一次测量	305	-	仅执行一次读取循环。采用 304 或者 306 选项。
PSR	DiagServer	选择测量并且读取一次	306	测量次数 (words 列表)	对应于 304、305。输入的数值保持选中状态，直至下次 304 或者 306 指令为止。
PSR	DiagServer	采用给定的闪存键对 ECU 进行闪存处理	400	闪存键 (S) 可选项： ECU-修饰符 (S)	采用空格分隔多个闪存键。如果闪存处理的 ECU 并非主要 ECU，则必须作为第二参数给出修饰符。必须默认数值：ECU 进行初始化而且闪存键必须有效。程序采用 401、405 或者采用 402 应答。
DiagServer	PSR	开始闪存处理	401	闪存键 (S)	
DiagServer	PSR	未能开始闪存处理	402	闪存键 (S) 错误文本 (S)	
DiagServer	PSR	完成闪存处理	405		
PSR	DiagServer	载入闪存件 (CFF 文件)	410	文件路径 (S)	需要完整路径。载入之前，会卸载之前的闪存件。如果输入空路径，则仅会进行卸载。采用 411 或者 412 进行应答。
DiagServer	PSR	载入闪存	411	闪存键 (S)	报告在闪存件中发现

		件			闪存键。
DiagServer	PSR	无法载入闪存件	412	Dateipfad (S) 错误文本 (S)	
PSR	DiagServer	载入闪存件并且对 ECU 进行闪存处理	430	文件路径 (S) 可选项: ECU-修饰符 (S)	与 410 以及 400 相同的动作, 但是仅当闪存件包括一个闪存键的时候。否则程序采用 402 应答。
PSR	DiagServer	显示闪存处理进度	440		
DiagServer	PSR	闪存处理进度	441	数字 (WORD) 在 0 到 100 之间	如果没有正在进行的闪存处理, 则返回 0。
PSR	DiagServer	显示闪存键	450	可选项: ECU-修饰符 (S)	显示在当前载入的闪存件中的所有闪存键
DiagServer	PSR	闪存键	451	闪存键 (S)	所有闪存键均采用空格分隔
PSR	DiagServer	激活驱动器	500	名称 (S) 数值 (S)	DiagServer 不会承认; 离散驱动器: 数值必须作为字符串予以传送, 比如, “FN_ELAB_AUS” (注: 大小写敏感!) 对于持续可调节驱动器而言, 比如, 可以输入数值 0.5 作为空终止字符串 “0.5” 或者 “0,5”, 主要取决于区域系统设置。
DiagServer	PSR	激活了驱动器	501	修饰符 (S) 结果 (S)	如果 OutputRef 可用, 那么会传送其结果
DiagServer	PSR	无法激活驱动器	502	修饰符 (S) 错误文本 (S)	
PSR	DiagServer	请求驱动器列表	510		对应于 560, 具有类型 “act”
PSR	DiagServer	请求驱动器信息	520	修饰符 (S)	对应于 561
PSR	DiagServer	执行任何服务	550	修饰符 (S) 参数 (S)	采用字符串方式传送参数, 采用 “&” 分隔。
DiagServer	PSR	执行服务	551	修饰符 (S) 结果 (S)	如果提供了 OutputRef, 则传送其结果。
DiagServer	PSR	执行服务时的错误	552	修饰符 (S) 错误文本 (S)	
PSR	DiagServer	请求所有服务的列表 (根据 VSB 文件进行过滤)	560	可选项: 类型 (S) act, adj, fun, pro, mea, gen. 默认值为 gen。	类型缩写词对应于: 驱动器、调整、功能、操作步骤、测量、通用。服务器采用 563 块进行回复。
PSR	DiagServer	请求所有服务的完整列表, 其中包括参数数量	561	可选项: 类型 (S) act, adj, fun, pro, mea, gen. 默认值为 gen。	服务器采用 563 块进行回复。
PSR	DiagServer	请求服务	562	修饰符 (S)	服务器采用 564 块进行

		的相关信息			回复。
DiagServer	PSR	列出规定类型的所有服务	563	修饰符 (S) 参数数量 (S)	对于 560 请求而言：仅为修饰符。 对于 561 请求而言：修饰符以及参数数量。 列表结束：空字符串
DiagServer	PSR	服务的相关信息	564	信息文本 (S)	每个服务多个块 最后一个块包括空字符串。
DiagServer	PSR	服务信息期间的问题	565	错误消息 (S)	
PSR	DiagServer	发送 API-I 消息	570	字节数组	仅输入有效负荷。程序负责计算头部以及校验和。
DiagServer	PSR	ECU 答复 API-I 消息	571	字节数组	仅有效负荷。
DiagServer	PSR	发送失败	572	错误文本 (S)	
PSR	DiagServer	发送 API-I 消息	580	作为字符串的字节数组	两位十六进制数字，采用空格分隔
DiagServer	PSR	ECU 答复 API-I 消息	581	作为字符串的字节数组	
PSR	DiagServer	启动 Java 程序	600	名称 (S)，可选参数 (S)	不可能同时启动多个程序，会发生错误消息（服务 602）。
DiagServer	PSR	启动 Java 程序	601	名称 (S)	启动可用程序的时候输出。
DiagServer	PSR	来自 Java 程序的结果，或者错误	602	结果 (W)	0=正确执行了程序 1=程序不可用 2=程序错误 3=程序正在运行 4=无法启动程序 5=其它错误
PSR	DiagServer	结束测试运行	700	-	结果为“测试运行结束”
DiagServer	PSR	结束测试运行	701	-	
PSR	DiagServer	结束程序	702	-	禁用 PSR 适配器。只要客户端应用程序访问 DiagServer，则 DiagServer 就会保持运行状态。（非推荐使用方式，仅为了兼容的目的而实现）。
PSR	DiagServer	服务器复位	710	-	对 CAESAR, CAESAR 文件，系统说明，发动机表进行复位。
DiagServer	PSR	服务器完成复位	711	-	答复 710
PSR	DiagServer	再次初始化服务器	720	-	重新载入 CAESAR、CAESAR 文件、系统说明、发动机表。
DiagServer	PSR	完成服务	721	-	答复 720

		器初始化			
DiagServer	PSR	未识别服务	800	服务 ID(Word)	
DiagServer	PSR	服务参数中的格式错误	801	-	参数格式错误（比如，字符串并未以空结尾）
DiagServer	PSR	格式错误	803	-	接收的消息长度错误（比如，小于6个字节，或者长度规范与消息长度并不匹配）
PSR	DiagServer	日志启动/停止	1000	文件名（S）	空名称：停止日志。仅采用 3964R 工作，未采用 TCP/IP
PSR	DiagServer	显示发动机表	1100		
DiagServer	PSR	发动机表	1101	发动机名称（S）	列表结束：空字符串
PSR	DiagServer	显示 Java 程序列表	1200		
DiagServer	PSR	Java 程序列表	1201	程序 ID（S）	列表结束：空字符串
最多可达 4999 – 为今后使用所保留					
DiagServer	PSR	更新服务	5000-3000	无，Word 或者 ASCII 字符串	提供用于用户定义的 Java 程序。详情参见附件 5.4 中的表 6。

7、示例：Java 程序

```
/**
 * Title: Example Java Routine
 * Description: Example of a test routine that is started from Ecoute
 */

import JVHandlers.JVECU;
import JVHandlers.JVSystem;
import JVHandlers.JVClient;
import JVHandlers.JVErrorService;
import JVHandlers.JVUtil;

public class Example {
    JVClient client = new JVClient();
    JVSystem system = new JVSystem();
    JVErrorService errorS;
    String[] ecus;
    JVECU ecu;
    //***** The following constants must be entered explicitly (depending on ECU). *****
    final String solleECU="SolleECU",
    functionQualifier="FunctionQualifier",procedureQualifier="ProcedureQualifier",
    adjustmentQualifier="AdjustmentQualifier",adjSetValue="AdjustmentValue";
    final String[] measQualifier={"meas_1","meas_2","meas_3"};
    //*****
    String measResult, funcResult, adjResult, adjGetValue, procResult;

    public Example() {
        // get the ECUs available in the system from the loaded VSB
        ecus = system.GetEcuQualifiers();

        // the program works with the desired ECU
        for (int i = 0; i < ecus.length; i++)
        {
            if (ecus[i].equalsIgnoreCase(solleECU)){
                ecu = new JVECU(ecus[i]);
                break;
            }
        }
        if(ecu==null){
            errorMessage("System does not contain desired ECU " + solleECU + "!",true);
            // "true" means that this error ends the program
            // the program would continue to run with "false"
        }
        client.Trace("ECU " + solleECU + " selected"); // output to Ecoute status window
        client.Trace("Establish contact..."); // establish contact with ECU solleECU...
        if (!ecu.Init()) {
            errorMessage("Unable to establish contact to " + solleECU + "!",true);
        }
        client.Trace("Contact established");

        // Measurements are read and displayed...
        client.Trace("Read measurements...");

        for (int i = 0; i < measQualifier.length; i++) {
            measResult = ecu.GetMeasurementValue(measQualifier[i]);
            if (ecu.IsErrorSet()) {
                errorMessage(measQualifier[i]+" read failed! - "+ecu.GetLastError(),false);
            }
            else {
                client.Trace(measQualifier[i]+" : "+measResult);
            }
        }

        // Execute function...
        client.Trace("Function " + functionQualifier + " being executed...");
        funcResult = ecu.ExecuteFunction(functionQualifier);
        if (ecu.IsErrorSet()) {
            errorMessage("Execution of function " + functionQualifier + " failed! - "+ecu.GetLastError(),false);
        }
        else {
            client.Trace(functionQualifier+": "+funcResult);
        }

        // Execute procedure...
```

```

client.Trace("Procedure " + procedureQualifier + " being executed...");
procResult = ecu.ExecuteProcedure(procedureQualifier);
if (ecu.IsErrorSet()) {
    errorMessage("Execution of procedure " + procedureQualifier + " failed! - "+ecu.GetLastError(),false);
}
else {
    client.Trace(procedureQualifier+": "+procResult);
}

// Set adjustment...
client.Trace("Adjustment " + adjustmentQualifier + " being set to " + adjSetValue + " ...");
adjResult = ecu.SetAdjustmentValue(adjustmentQualifier,adjSetValue);
if (!adjResult) {
    errorMessage("Setting of adjustment" + adjustmentQualifier + " failed! - "+ecu.GetLastError(),true);
}
else {
    client.Trace("Setting of adjustment " + adjustmentQualifier + " was successful.");
}

// Read adjustment value...
client.Trace("Adjustment " + adjustmentQualifier + " being read...");
adjGetValue = ecu.GetAdjustmentValue(adjustmentQualifier);
if (ecu.IsErrorSet()) {
    errorMessage("Reading of adjustment" + adjustmentQualifier + " failed! - "+ecu.GetLastError(),true);
}
else {
    client.Trace("Adjustment " + adjustmentQualifier + " is set to " + adjGetValue + " .");
}

// Clear all errors in ECU memory...
client.Trace("Clear all errors...");
errorS = new JVErrorservice(ecu);
if (!errorS.ClearAllErrors()) {
    errorMessage("Clear all errors failed!",false);
}
client.Trace("All errors cleared.");

// disconnect and end Java client
JVUtil.DisconnectFromServer();
System.exit(0);
}

public static void main(String[] args) {
    Example Ex_1 = new Example();
}

public void errorMessage(String text, boolean finish) {
    client.Beep();
    client.Information(text);
    client.Trace(text);
    if (finish){
        JVUtil.DisconnectFromServer();
        System.exit(1);
    }
}

} // class Example

```