

了解路由器 ATM 接口中的 SSCOP 消息

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[规则](#)

[了解 QSAAL 协议栈](#)

[什么是 SSCOP ?](#)

[了解 SSCOP 包尾](#)

[SSCOP 消息或 PDU](#)

[SSCOP 定时器](#)

[SSCOP 序号](#)

[调试输出示例](#)

[相关信息](#)

简介

协议通常定义作为通信规则两个设备之间的。信令协议定义了通信规则使用信令消息创建根据要求或交换虚拟电路的两个ATM接口之间的(SVC)传送用户数据。 ATM接口实际上支持包括“从 Q.2931用户网络接口(UNI)协议和特殊信令ATM适配层的发信号的协议栈(SAAL)的用户”信令消息。 SAAL撰写特定于服务面向连接协议(SSCOP)和Service-Specific Coordination Function (SSCF)。

清楚地， ATM信令引入许多缩略语，能一起做SSCOP似乎复杂，当确实执行一项简单任务——时请传输在UNI间的信令消息。

当调查原因意外的LAN仿真(LANE)客户端状态更改的时，对SSCOP的了解可以是一关键故障排除工具。当这样更改发生时，路由器打印下面消息对日志。

注意： 下面的输出线路出现在多条线路由于空间限制。

```
Aug 25 18:32:59.973 MEST: %LANE-5-UPDOWN: ATM0.1 elan default:
LE Client changed state to down
Aug 25 18:32:59.981 MEST: %LANE-5-UPDOWN: ATM0.39 elan admin:
LE Client changed state to down
```

本文在SSCOP提供直接的理论。它使用简表描述SSCOP协议数据单元(PDU)，序号和状态变量。它然后提交从**debug sscop events**命令的输出说明PDU、编号和变量如何出现在Cisco路由器。

注意： 本文焦点在作为UNI的用户端的Cisco路由器。本文不讨论网络之间的接口发信号。

先决条件

要求

本文档没有任何特定的要求。

使用的组件

本文档不限于特定的软件和硬件版本。

规则

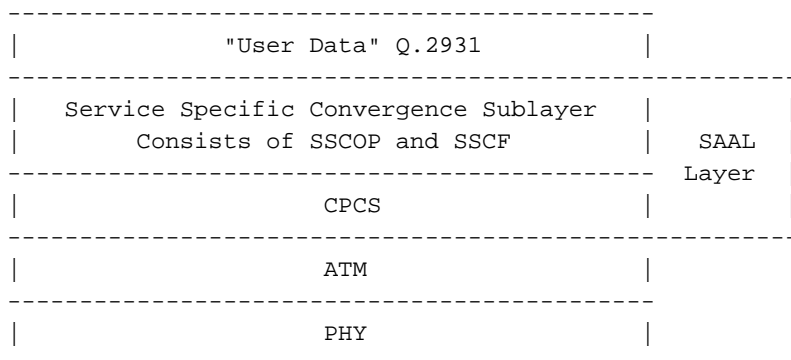
有关文档规则的详细信息，请参阅 [Cisco 技术提示规则](#)。

了解 QSAAL 协议栈

ATM是协议和协议栈。考虑如下图示和注释是重要的三协议栈如何在支持信令和网路管理的ATM接口平行运行。每协议栈提供一个不同的功能给接口的成功的操作。

		用户平面	管理平面
Q.2931 UNI发信号		语音、视频或者数据	集成本地管理接口 (ILMI)
SAAL	SSCF	ATM适配层 (AAL)	AAL
	SSCOP		
	公共会聚子层 (CPCS)		
ATM 层			
物理层— SONET/Synchronous数字体系(SDH)， DS3、E3、T1等等。			

在用户平面上，最普通的AAL是AAL5，提供一8字节报尾。SAAL代表AAL5的变化。什么做它不同的是包括SSCOP和SSCF的一个有服务特效的会聚子层(SSCS)。此图表说明这些层：



ATM接口传送信令消息“在波段外面”或者外部一般数据连接的带宽。他们以一种特殊Q.2931 SAAL (QSAAL)封装类型使用(PVC)配置的一个专用的永久性虚拟连接。

发出pvc vpi/vci命令在ATM路由器接口配置QSAAL PVC。

```
7500-3.4(config)# interface atm 3/0 7500-3.4(config-if)# pvc 0/5 ? ilmi Configure the
management PVC for this interface qsaal Configure the signaling PVC for this interface <cr>
7500-3.4(config-if)# pvc 0/5 qsaal
```

思科ATM交换机来预先配置与在每个接口的QSAAL PVC。发出show atm vc interface atm命令确认此默认配置。

```
ls1010-2# show atm vc interface atm 0/0/2
Interface      VPI   VCI   Type   X-Interface  X-VPI X-
VCI  Encap Status ATM0/0/2    0    5    PVC    ATM2/0/0    0    45    QSAAL  UP
ATM0/0/2      0    16    PVC    ATM2/0/0    0    37    ILMI   UP
```

SSCOP在几国际电信联盟电信标准化部门建议定义。Q.2110建议在ATM路由器接口提供信息与排除故障SSCOP相关的问题有关。

- [Q.2100](#) —定义了SAAL结构。
- [Q.2110](#) —定义了SSCOP作为协议实体。
- [Q.2130](#) —定义了UNI接口的SSCF。
- [Q.2140](#) —定义了NNI接口的SSCF。
- [I.363](#) —定义了CPCS。

注意： UNI和NNI接口使用SSCF不同的版本。 NNI在本文没有讨论。

什么是 SSCOP ?

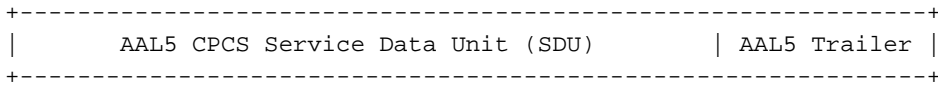
SSCOP是提供保证的传输协议，消息按顺序发送对在它上位于发信号的协议栈的信令协议。SSCOP也执行流量控制、错误报告到管理层面和一个保活功能。

此表描述许多重要功能该SSCOP提供对ATM接口：

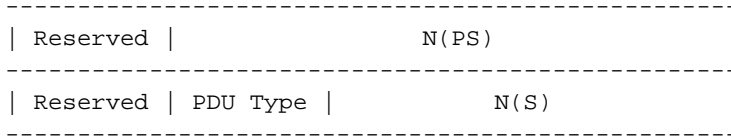
功能	说明
信令消息按顺序的和可靠传输	UNI Q.2931协议生成的信令消息构成“用户数据”在信令栈内。SSCOP通过序号和有选择性的重新传输保持这些消息定货。注意SSCOP不检查信令消息的内容。
流控制	定在对等体ATM接口传送SSCOP信息的速率的限额。
错误报告	检测并且报告在SSCOP的操作的错误。
Keepalive	交换POLL消息在固定间隔保证两端和连接依然是可操作和活动，特别在期限，当信令消息没有传送时。
本地数据检索	维护统计信息(看得见使用show sscop命令)在对等体ATM接口不“发布”或确认的信令消息。
状态报告	提供传达状态信息的信息，包括信息给管理层面。

了解 SSCOP 包尾

ATM用户网络接口接口使用Q.2931作为信令协议。SSCOP填充Q.2931消息对4个字节的多个并且添附总是4个字节的多个SSCOP特定的信息的报尾。



SSCOP包尾的内容随PDU种类变化，在下一部分描述，[SSCOP消息或PDU](#)。此图表显示SSCOP包尾的格式轮询PDU的：



SSCOP 消息或 PDU

SSCOP使用15消息类型或PDU执行功能。在发送和接收的每个PDU编号的show sscop命令提供统计信息。在此输出示例中，ATM接口3/0发送，并且接收11 PDU，包括8个POLL PDU和1请开始PDU：

```

7500# show sscop atm 3/0 SSCOP details for interface ATM3/0    Current State = Active, Uni
version = 4.0    [output omitted]    Statistics -    Pdu's Sent = 11, Pdu's Received = 11,
Pdu's Ignored = 0    Begin = 1/1, Begin Ack = 1/1, Begin Reject = 0/0    End = 1/0, End
Ack = 0/1    Resync = 0/0, Resync Ack = 0/0    Sequenced Data = 0/0, Sequenced Poll Data =
0/0    Poll = 8/8, Stat = 8/8, Unsolicited Stat = 0/0    Unassured Data = 0/0, Mgmt Data =
0/0, Unknown Pdu's = 0    Error Recovery/Ack = 0/0, lack of credit 0

```

此表分组根据功能的SSCOP消息：

功能	消息缩写	消息名称	说明
连接建立	B G N	开始	开始在两个ATM接口之间的SSCOP连接进程。初始化对等体缓冲区和传输和接收计数器。
	B G A K	开始确认	确认对等体连接请求。
	B G R E J	开始拒绝	拒绝对等体连接请求。对等体重新传输BGN PDU并且继续首次连接。
连接卸载	结 束	结束	发布两个对等体ATM设备之间的连接。
	E N D A K	结尾确认	确认版本请求。
再同步	R S	再同步	再同步信息缓冲器以及发射器和接收方状态变量或者计数器。

	R S A K	重新同步的确认	确认再次同步请求。
错误恢复	E R	错误恢复	从在活动连接时生成的错误恢复。
	E R A K	错误恢复确认	确认错误恢复请求。
确定数据传输	S D	程序化的数据	转移“从UNI Q.2931信令协议的用户”消息到对等体。
	P O L L	Status请求	请求关于对等体的状态信息。
	英 斯 塔 特	恳求的状态答复	代表对轮询PDU的一答复。提供关于SD PDU成功的接收的信息，最后轮询PDU的序号。它也包含指示的信用值多少更多消息对等体或不能在确认前发送。
	U S T A T	未经请求的状态答复	communicates丢失或未命中通过分析在其他PDU的序号检测的PDU。
不确定的数据数据数据量	U D	未编号的数据	传送“在对等体之间的用户”消息。不包括序号，并且能丢失没有通知。
管理数据数据数据量	M D	管理数据	传达管理信息给管理层面。不包括序号，并且能丢失没有通知。

注意：有未知PDU输入密码，不是32位对齐，也不是陈述的类型的PDU的适当的长度的ITU-T Q.2110建议定义了无效PDU作为PDU。

SSCOP 定时器

SSCOP跟随状态机，协议通过几状态移动在变为前活动。一套五计时器控制(一部分)，当SSCOP过渡了到另一状态。发出**sscop**命令在接口配置模式查看这些计时器。

```
7200(config-if)# sscop ?      cc-timer          timer (in secs) to send BGN/END/RS/ER pdu at the
                                connection control phase  idle-timer          timer (in secs) to send poll
pdu at the idle phase  keepalive-timer      timer (in secs) to send poll pdu at the transient
                                phase  noResponse-timer   timer (in secs) at lease one STAT PDU needs to
be                                received  poll-timer         timer (in msecs) to send poll pdu at the
active                                phase
```

此表描述五个SSCOP定时器：

计时器	说明	默认值
cc	连接控制(cc)是用于的一组流程建立，发布或者再同步两个ATM接口之间的一个SSCOP连接。cc计时器设置BGN重新传输，END或者RS PDU之间的时间，当等待确认时。max-cc值设置重试次数数量。	1秒 (秒)
	如果连接足够稳定的，并且没有数据信息传送的和没有未清确认，SSCOP交换机从计时器keepalive到计时器。	10 秒
keepalive-timer	当SD PDU没有为发射排队也不是未清待待确认时，控制轮询PDU的发射的之间最大时间。	5 秒
noResponse-timer	与其他两个计时器平行的运行-和Keepalive。设置期间至少一个英斯塔特消息必须接收以回应POLL的最大时间时间间隔在。如果此计时器超时，连接被中断。	45秒
	当SD PDU为发射排队或是未清待待确认时，设置传送轮询PDU之间的最大时间。	1000 毫秒 (msecs)

发出show sscop atm命令查看SSCOP定时器的默认值。

```
7500# show sscop atm 3/0 SSCOP details for interface ATM3/0    Current State = Idle,    Uni
version = 4.0    Send Sequence Number: Current = 0,    Maximum = 30    Send Sequence Number Acked
= 0    Rcv Sequence Number: Lower Edge = 0, Upper Edge = 0, Max = 30    Poll Sequence Number =
0, Poll Ack Sequence Number = 1    Vt(Pd) = 0    Vt(Sq) = 0    Timer_IDLE = 10 - Inactive
Timer_CC = 1 - Inactive    Timer_POLL = 1000 - Inactive    Timer_KEEPA L I V E = 5 - Inactive
Timer_NO-RESPONSE = 45 - Inactive    Current Retry Count = 0, Maximum Retry Count = 10    /---
Output suppressed.
```

SSCOP 序号

在ATM接口的SSCOP进程跟踪两套序号或状态变量，然后映射这些值到实际PDU的字段。特别地，SD PDU和POLL PDU连续和独立地被编号。发射器和接收方维护序号作为状态变量。这些变量然后映射到实际参数或字段SSCOP的PDU。show sscop命令显示序号的当前值。

```
ATM# show sscop SSCOP details for interface ATM0    Current State = Active,    Uni version = 3.1
    Send Sequence Number: Current = 79,    Maximum = 109    Send Sequence Number Acked = 79    Rcv
Sequence Number: Lower Edge = 93, Upper Edge = 93, Max = 123    Poll Sequence Number = 32597,
Poll Ack Sequence Number = 32597    Vt(Pd) = 0    Vt(Sq) = 1    Timer_IDLE = 10 - Active    /---
Output suppressed.
```

以下部分描述状态变量和实际PDU编号。

在发射器的状态变量

ATM接口保持开始与VT的一套transmit-side状态变量。

状态变量	名称	说明

VT	发送	该的序号与每个SD PDU的增量。当同样SD PDU被重新传输时，不增加。
VT(PS)	投票发送	该的序号与每轮询PDU的增量。
VT(A)	确认	预计其次确认SD PDU的序号。每次增量SD PDU确认。
VT(PA)	投票确认	预计其次接收作为确认到轮询PDU英斯塔特PDU的序号。
VT(MS)	最大发送	传送的接口能发送PDU的最高的序号(和将接受)的接收方，不用收据任一个下列的PDU：USTAT、英斯塔特、BGN、BGAK、RS、RSAK、ER或者ERAK PDU。换句话说，VT(MS)定义了传输窗口大小。VT高于VT(MS)不应该。
VT(PD)	轮询数据	SD PDU编号传送在两个POLL PDU之间。在SD PDU和重置的发射的增量到零在轮询PDU的发射。
VT(CC)	连接控制	未承认的BGN、END、ER或者RS PDU编号。如果ATM接口发送END PDU以回应协议错误，SSCOP迁移直接向空闲状态，并且不增加VT(CC)值。
VT(SQ)	发射器连接顺序	identifies重新传输BGN、ER和RS PDU。初始化到零，当SSCOP进程开始时然后映射到N(SQ)。

在接收方的状态变量

ATM接口保持开始与VR的一套接收侧状态变量。

状态变量	名称	说明
VR(R)	接收	接收方期待下个按顺序的SD PDU的序号。当该消息被看到时，它被增加。
VR(H)	最高预计	在SD PDU的最高的期望的序号。更新从下个SD或POLL消息，并且应该大致是相等的与对等体VT。
VR(MR)	最大接收	在接收方将接受的SD PDU的最高的序号。换句话说，接收方将允许至VR(MR) - 1，然后丢弃所有SD PDU用一个更高的序号。更新VR(MR)根据执行的。
VR(SQ)	接收方连接顺序	用于识别被重新传输的BGN、ER和RS PDU。当ATM接口接收这些PDU之一时，N(SQ)值与其自己的VR(SQ)值比较。如果两个值不同的，PDU处理作为一个新的消息。如果两个值是相等的，PDU识别作为重新传输。

状态变量翻译到PDU参数

接收和传输状态变量翻译或被映射到实际PDU与有些不同的名称的参数。此表显示他们派生的PDU参数和状态变量：

参数	映射从	说明
N(SQ)	VR(SQ)	连接顺序序号输入了一BGN、RS或者ER PDU。用于与VR(SQ)计数器在接收方识别这些PDU任何重新传输。
N	VT	发送序列号输入了每个SD或轮询PDU并且随新建的其中每一增加了，非转播的PDU。
N(PS)	VT(PS)	输入轮询PDU和匹配英斯塔特PDU有关两个消息。
N(R)	VR(R)	接收序号输入了英斯塔特或USTAT PDU。发送由对等设备，当确认一个或更多信令消息时收据。
N(MR)	VR(MR)	输入以下PDU：英斯塔特，USTAT，RS，RSAK，ER，ERAK，BGN，BGAk。指示剩余接收除帐数量，并且对等体是否能传送另一信息。例如，N(MR)值为5意味着对等体能发送5 PDU，无需等待答复。

调试输出示例

下面的输出通过发出debug sscop event atm 3/0命令生成在有PA-A3的一个7500系列路由器。用蓝色用于解释debug输出。

```
*Mar 21 03:18:43.440: SSCOP(ATM3/0): i Begin pdu, Idle state, length = 8
*Mar 21 03:18:43.440: SSCOP(ATM3/0): Rcv Begin in Idle State
*Mar 21 03:18:43.440: SSCOP(ATM3/0): receive window in Begin Pdu = 30
*Mar 21 03:18:43.440: SSCOP(ATM3/0): o Begin Ack pdu, Idle state, rcv window v(mr) = 30
!--- A BEGIN PDU is received by the router, which responds with a BEGIN ACK PDU. !--- The window
size V(MR) is initialized to 30. *Mar 21 03:18:43.440: SSCOP(ATM3/0): state changed from Idle to
Active *Mar 21 03:18:47.968: SSCOP(ATM3/0): o Poll pdu, state = Active, n(s) = 0, n(ps) = 1 *Mar
21 03:18:47.968: SSCOP(ATM3/0): i Stat pdu, Active state, length = 12 *Mar 21 03:18:47.968:
SSCOP(ATM3/0): Rcv Stat in Active State *Mar 21 03:18:47.968: SSCOP(ATM3/0): processStatPdu: ps
1, nmr 30, nr 0 *Mar 21 03:18:47.968: SSCOP(ATM3/0): processStatPdu: vtPa 1, vps 1 *Mar 21
03:18:47.968: SSCOP(ATM3/0): processStatPdu: vta 0, vts 0 *Mar 21 03:18:47.968: SSCOP(ATM3/0):
processStatPdu: listCount = 0 - normal !--- This is the first outbound POLL PDU and inbound STAT
PDU. *Mar 21 03:18:48.040: SSCOP(ATM3/0): * Poll pdu, ns = 0, nps = 1 *Mar 21 03:18:48.040:
SSCOP(ATM3/0): o Stat pdu, n(r) = 0, n(mr) = 30, n(ps) = 1 !--- The "*" indicates an inbound
POLL PDU from the attached ATM switch. !--- The router responds with an outbound STAT PDU. *Mar
21 03:18:57.292: SSCOP(ATM3/0): o Poll pdu, state = Active, n(s) = 0, n(ps) = 2 *Mar 21
03:18:57.292: SSCOP(ATM3/0): i Stat pdu, Active state, length = 12 *Mar 21 03:18:57.292:
SSCOP(ATM3/0): Rcv Stat in Active State *Mar 21 03:18:57.292: SSCOP(ATM3/0): processStatPdu: ps
2, nmr 30, nr 0 *Mar 21 03:18:57.292: SSCOP(ATM3/0): processStatPdu: vtPa 1, vps 2 *Mar 21
03:18:57.292: SSCOP(ATM3/0): processStatPdu: vta 0, vts 0 *Mar 21 03:18:57.292: SSCOP(ATM3/0):
processStatPdu: listCount = 0 - normal !--- This is the second outbound POLL PDU and inbound
STAT PDU. N(PS) and V(PS) !--- increment to 2. *Mar 21 03:18:58.004: SSCOP(ATM3/0): * Poll pdu,
ns = 0, nps = 2 *Mar 21 03:18:58.004: SSCOP(ATM3/0): o Stat pdu, n(r) = 0, n(mr) = 30, n(ps) = 2
*Mar 21 03:19:06.812: SSCOP(ATM3/0): o Poll pdu, state = Active, n(s) = 0, n(ps) = 3 *Mar 21
```



```
03:19:06.812: SSCOP(ATM3/0): i Stat pdu, Active state, length = 12 *Mar 21 03:19:06.812:
SSCOP(ATM3/0): Rcv Stat in Active State *Mar 21 03:19:06.812: SSCOP(ATM3/0): processStatPdu: ps
3, nmr 30, nr 0 *Mar 21 03:19:06.812: SSCOP(ATM3/0): processStatPdu: vtPa 2, vps 3 *Mar 21
03:19:06.812: SSCOP(ATM3/0): processStatPdu: vta 0, vts 0 *Mar 21 03:19:06.812: SSCOP(ATM3/0):
processStatPdu: listCount = 0 - normal *Mar 21 03:19:07.228: SSCOP(ATM3/0): * Poll pdu, ns = 0,
nps = 3 *Mar 21 03:19:07.228: SSCOP(ATM3/0): o Stat pdu, n(r) = 0, n(mr) = 30, n(ps) = 3 !---
This is the third outbound POLL PDU and inbound STAT PDU. N(PS) and V(PS) !--- increment to
3. N(MR) remains at 30. N(S), VT(S), and VT(A) remain at 0 since !--- no sequenced Q.2931 "user"
data is being transmitted.
```

作为保活机制一部分，**debug**输出捕获SSCOP发送的消息在连接建立期间和。一个同时捕获**show sscop atm**命令，当调试指令运行时显示增加值Pdu's和Pdu's，以及和Stat。

```
7500# show sscop atm 3/0 SSCOP details for interface ATM3/0 Current State = Active, Uni
version = 4.0 Send Sequence Number: Current = 0, Maximum = 30 Send Sequence Number Acked
= 0 Rcv Sequence Number: Lower Edge = 0, Upper Edge = 0, Max = 30 Poll Sequence Number =
6, Poll Ack Sequence Number = 6 Vt(Pd) = 0 Vt(Sq) = 1 Timer_IDLE = 10 - Active
Timer_CC = 1 - Inactive Timer_POLL = 1000 - Inactive Timer_KEEPALIVE = 5 - Inactive
Timer_NO-RESPONSE = 45 - Inactive Current Retry Count = 0, Maximum Retry Count = 10 AckQ
count = 0, RcvQ count = 0, TxQ count = 0 AckQ HWM = 0, RcvQ HWM = 0, TxQ HWM = 0 Local
connections currently pending = 0 Max local connections allowed pending = 0 Statistics -
Pdu's Sent = 9, Pdu's Received = 9, Pdu's Ignored = 0 Begin = 1/1, Begin Ack = 1/1,
Begin Reject = 0/0 End = 1/0, End Ack = 0/1 Resync = 0/0, Resync Ack = 0/0
Sequenced Data = 0/0, Sequenced Poll Data = 0/0 Poll = 6/6, Stat = 6/6, Unsolicited Stat =
0/0 Unassured Data = 0/0, Mgmt Data = 0/0, Unknown Pdu's = 0 Error Recovery/Ack =
0/0, lack of credit 0 7500# show sscop atm 3/0 SSCOP details for interface ATM3/0 Current
State = Active, Uni version = 4.0 Send Sequence Number: Current = 0, Maximum = 30 Send
Sequence Number Acked = 0 Rcv Sequence Number: Lower Edge = 0, Upper Edge = 0, Max = 30
Poll Sequence Number = 7, Poll Ack Sequence Number = 7 Vt(Pd) = 0 Vt(Sq) = 1 Timer_IDLE
= 10 - Active Timer_CC = 1 - Inactive Timer_POLL = 1000 - Inactive Timer_KEEPALIVE = 5
- Inactive Timer_NO-RESPONSE = 45 - Inactive Current Retry Count = 0, Maximum Retry Count
= 10 AckQ count = 0, RcvQ count = 0, TxQ count = 0 AckQ HWM = 0, RcvQ HWM = 0, TxQ HWM =
0 Local connections currently pending = 0 Max local connections allowed pending = 0
Statistics - Pdu's Sent = 10, Pdu's Received = 10, Pdu's Ignored = 0 Begin = 1/1,
Begin Ack = 1/1, Begin Reject = 0/0 End = 1/0, End Ack = 0/1 Resync = 0/0, Resync
Ack = 0/0 Sequenced Data = 0/0, Sequenced Poll Data = 0/0 Poll = 7/7, Stat = 7/7,
Unsolicited Stat = 0/0 Unassured Data = 0/0, Mgmt Data = 0/0, Unknown Pdu's = 0
Error Recovery/Ack = 0/0, lack of credit 0
```

相关信息

- [ITU-T用户网络接口\(UNI\)规格](#)
- [ATM论坛UNI规格](#)
- [ATM技术支持页](#)
- [技术支持 - Cisco Systems](#)