

# White paper sobre telemetria orientada a modelos ASR9K

## Contents

[Introduction](#)

[Público-alvo](#)

[Uma breve introdução à telemetria](#)

[Por que telemetria](#)

[A necessidade de abandonar o SNMP](#)

[Vantagens da telemetria de transmissão](#)

[Especificações técnicas de telemetria orientada por modelo](#)

[Funções de telemetria](#)

[Componentes da telemetria](#)

[YANG](#)

[Codificação](#)

[Transporte](#)

[Telemetria baseada em cadência vs telemetria baseada em evento](#)

[Diretrizes de design de telemetria](#)

[Como selecionar o esquema de codificação](#)

[Consideração de projeto de rede de transporte](#)

[Avaliando opções de configuração de telemetria](#)

[Exemplos de configuração de telemetria](#)

[IOS-XR](#)

[Divisão da Configuração de Discagem Externa](#)

[Definir grupos de sensores](#)

[Definir grupos de destinos](#)

[Definir a Assinatura](#)

[Exemplo de configuração completa](#)

[Vantagens da discagem externa](#)

[Separação da Configuração de Discagem](#)

[Habilitar gRPC](#)

[Definir grupos de sensores](#)

[Definir a Assinatura](#)

[Modelo de configuração completo e exemplo](#)

[Vantagens da discagem](#)

[Telemetria orientada a eventos](#)

[Configuração de telemetria orientada a eventos](#)

[Modelo de Configuração Completo e Exemplo para DISCAGEM](#)

[Modelo de Configuração Completo e Exemplo para DISCAGEM](#)

[Validando a telemetria com comandos SHOW](#)

[Pilha de Coleção de Telemetria](#)

[Considerações de implantação para telemetria em uma rede](#)

[Dimensionamento](#)

[Transmitir apenas os dados necessários](#)

[Considere a quantidade de dados de transmissão](#)

[Referências](#)

## Introduction

### Público-alvo

Este white paper tem como objetivo ajudar os clientes a obter um rápido entendimento sobre o recurso de telemetria orientada a modelos (MDT) em geral e como ele foi implementado no Aggregation Services Router 9000 (ASR9K), incluindo algumas diretrizes de design e detalhes de configuração. Ele também inclui algumas considerações de implantação, que serão úteis durante a implantação desse recurso usando o ASR9K. Em geral, este white paper pode ser um guia de referência rápida para qualquer pessoa que trabalhe com esse recurso.

Embora a telemetria seja introduzida como um recurso genérico, o foco é na implementação do ASR9K; isto é, nem todos os recursos suportados por outras plataformas da Cisco são suportados pela plataforma ASR9K e algumas implementações de recursos podem ser específicas do ASR9K.

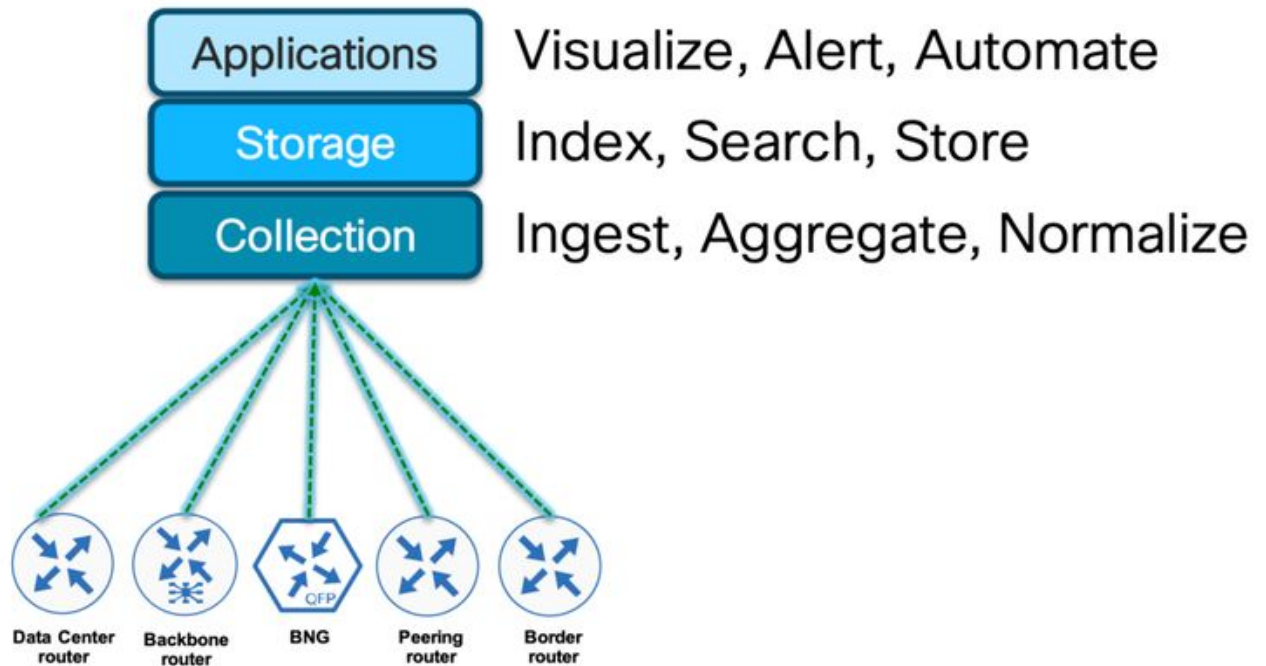
## Uma breve introdução à telemetria

Para começar, em termos simples, a telemetria é o processo de coleta de **dados operacionais úteis**. De acordo com a Wikipédia, a telemetria é um processo de comunicação automatizado pelo qual as medições e outros dados são coletados em pontos remotos ou inacessíveis e transmitidos para o equipamento receptor para monitoramento. A própria palavra de telemetria é derivada das raízes gregas: tele = remoto, e metron = medida.

Para o gerenciamento de rede, os operadores de rede têm um longo histórico de confiar no SNMP (Simple Network Management Protocol Protocolo de Gerenciamento de Rede Simples). Embora o SNMP seja amplamente adotado para monitoramento de rede, ele nunca foi usado para configuração, embora a capacidade de configuração com snmp estivesse sempre lá. Os operadores escreveram scripts de automação para lidar com tarefas de configuração diárias, mas os scripts são desafiadores para essas tarefas e difíceis de gerenciar.

Assim, os operadores passaram para o gerenciamento orientado por modelo de dados. A configuração de rede é baseada em modelos de dados YANG enviados por protocolos como o netconf, por exemplo. O simples envio da configuração não implica que o serviço configurado esteja em execução; é necessário que haja um mecanismo que possa monitorar os dados operacionais dos serviços ao mesmo tempo que a configuração. É aqui que os modelos de dados oper, que a telemetria usa para enviar informações para fora do dispositivo, ajudam. Portanto, a configuração é orientada pelo modelo de dados YANG, assim deve ser a verificação do serviço também; como no caso da telemetria, a fim de ter o mesmo objeto semântico. Assim, o termo é chamado **telemetria orientada a modelos** ou telemetria de fluxo contínuo.

A telemetria orientada a modelos (MDT - Model Driven Telemetry) foi introduzida no **cXR** (IOS XR de 32 bits) desde a versão **6.1.1** e permite a coleta e a medição de dados críticos em tempo quase real, fornecendo uma resposta rápida para a maioria dos problemas operacionais da rede moderna.



### Arquitetura de telemetria de alto nível

O MDT utiliza modelos de dados estruturados suportados pelo dispositivo de rede e fornece dados críticos definidos nesses modelos de dados. **A telemetria ajuda os clientes a gerenciar sua rede de vários fornecedores** usando um sistema de gerenciamento de rede, processo e aplicativos comuns, já que os dados coletados da rede são baseados em padrões e são uniformes em toda a implementação do fornecedor.

Em vez de esperar pela recuperação de dados (pull) de uma estação de gerenciamento centralizada (normalmente SNMP NMS); com MDT, os dispositivos de rede enviam proativamente (**push**) dados de desempenho relativos às funções vitais da rede, como informações de encaminhamento de pacotes, estatísticas de erro, status do sistema, recursos de CPU e memória, etc.

## Por que telemetria

A coleta de dados para fins analíticos e de solução de problemas sempre foi um aspecto importante no monitoramento da integridade de uma rede. Há vários mecanismos disponíveis, como SNMP, CLI e Syslog, para coletar dados de uma rede. Embora esses métodos tenham servido a rede por muito tempo, mas não se encaixem na rede moderna em que a demanda por automação, os serviços em escala são fundamentais. As informações de integridade da rede, estatísticas de tráfego e informações críticas de infraestrutura são enviadas para uma estação remota no NMS, onde são usadas para melhorar o desempenho operacional e reduzir o tempo de solução de problemas. Um modelo pull, como o snmp, no qual um cliente faz o poll de todos os nós de rede, não é eficiente. A carga de processamento nos nós de rede aumenta quando há mais clientes para pesquisar. Ao contrário, um modelo push tem a capacidade de transmitir dados continuamente para fora da rede e notificar o cliente. A telemetria permite o modelo push, que fornece acesso quase em tempo real aos dados de monitoramento.

A telemetria de fluxo fornece um mecanismo para selecionar dados de interesse dos roteadores e transmiti-los em um formato padrão para estações de gerenciamento remoto para monitoramento. Esse mecanismo permite o ajuste fino da rede com base em dados em tempo real, o que é crucial para sua operação contínua. A granularidade mais fina e a maior frequência dos dados disponíveis por telemetria permitem um melhor monitoramento do desempenho, resultando em

uma melhor solução de problemas.

Ele ajuda na utilização mais eficiente da largura de banda na rede, na utilização do link, na avaliação de riscos e na escalabilidade. Com a telemetria de fluxo contínuo, mais dados quase em tempo real estão disponíveis à disposição dos operadores de rede, o que ajuda a melhorar a tomada de decisões.

## A necessidade de abandonar o SNMP

O SNMP existe há três décadas e a maneira como ele opera não mudou para atender às necessidades de monitoramento das redes modernas. O problema real é a velocidade de execução do SNMP.

Os três principais desafios impostos pelo SNMP são, na verdade, parte de seu comportamento operacional fundamental e, portanto, o SNMP oferece pouco ou nenhum espaço para aprimoramento e a telemetria trata todos os três inerentemente.

- **Velocidade de execução e necessidade de monitoramento em tempo real**

O SNMP usa o modelo PULL - operações GetBulk / GetNext que funcionam de forma linear atravessando as tabelas de uma coluna para outra caixa. Além disso, várias solicitações são necessárias no caso de tabelas grandes que não se encaixam em um pacote. Esse é o maior gargalo que faz com que o SNMP fique lento e os dados enviados geralmente ficam desatualizados por um determinado fator de tempo em minutos. Esse atraso simplesmente não é aceitável para os requisitos modernos de monitoramento de rede.

MDT (Model Driven Telemetry) usa o modelo PUSH e está inerentemente livre da limitação listada acima, pois sabe quais dados devem ser enviados a quem e em que intervalo. Ele precisa apenas de uma consulta para coletar dados e usa modelos internos pré-desenvolvidos para velocidade ultrarrápida das operações internas, permitindo assim a entrega de muito mais dados em um tempo consideravelmente menor.

- **Despesas gerais adicionais e falta de opções de otimização**

Os dados que estão sendo extraídos pelo SNMP são, na verdade, armazenados como estruturas de dados internas e precisam ser convertidos internamente pelo nó. Esse é um trabalho adicional nos bastidores onde o nó de rede mapeia estruturas de dados internas no formato SNMP. Há otimizações internas executadas, mas elas ainda não são suficientes.

Por outro lado, a Telemetria retira diretamente as estruturas de dados internas e executa um processamento mínimo antes de enviar esses dados, fornecendo assim os dados mais atualizados com o menor tempo e esforço possível.

- **Natureza Linear da Carga de Trabalho**

Cada estação de polling adicional levará a carga de trabalho adicional no nó, mesmo se estivermos fazendo polling dos mesmos dados exatos ao mesmo tempo. O acesso paralelo do mesmo MIB a partir de várias estações de interrogação pode levar a uma resposta mais lenta e a uma maior utilização da CPU. Isso é evidente especialmente no caso de tabelas grandes, onde várias estações acessam partes diferentes da mesma tabela MIB.

A telemetria, por outro lado, precisa receber dados uma vez e replicar os pacotes se os mesmos dados forem exigidos por vários destinos. O modelo Push supera o SNMP Pull para velocidade e

escala.

Com o MDT, a abordagem da coleta de dados muda radicalmente e seus princípios fundamentais são listados na tabela abaixo e comparados com os pontos-chave da tecnologia SNMP.

#### Simple Network Management Protocol

Informações não em tempo real  
Pouco escalável  
Modelo de recepção  
Não automatizado

#### Telemetria orientada a modelo (MDT)

Informações em tempo real  
Altamente escalável  
Modelo de envio  
Automação pronta/orientada por modelo de dado

## Vantagens da telemetria de transmissão

Os dados de telemetria em tempo real transmitidos são úteis para:

**Planejamento de capacidade/Otimização de tráfego:** quando a utilização da largura de banda e as quedas de pacotes em uma rede são monitoradas com frequência, é mais fácil adicionar ou remover links, redirecionar o tráfego, modificar a vigilância e assim por diante. Com tecnologias como o redirecionamento rápido, a rede pode comutar para um novo caminho e redirecionar mais rápido do que o mecanismo de intervalo de poll SNMP. A transmissão contínua de dados de telemetria ajuda a fornecer tempo de resposta rápido para tráfego mais rápido.

**Melhor visibilidade:** ajuda a detectar e evitar rapidamente situações de falha que resultam após uma condição problemática na rede.

## Especificações técnicas de telemetria orientada por modelo

A seção a seguir trata das funções técnicas e dos principais componentes da Telemetria Acionada por Modelo IOS XR, também conhecida como MDT.

### Funções de telemetria

A estrutura de telemetria é organizada em três blocos funcionais separados e interligados.

O primeiro bloco trata da **representação de dados**, que é como a informação referente a análise ou medições é organizada a bordo.

O segundo bloco é sobre **codificação**. A cada intervalo de amostra, a telemetria converte os dados de medição acima em um formato que pode ser serializado através do fio. É claro que o controlador na outra extremidade deve ser capaz de decodificar os dados para ter uma cópia idêntica dos dados originais enviados pelo dispositivo.

O último bloco é sobre **transporte**. Essa é a pilha de protocolos usada para transferir dados entre dispositivos.

A tabela a seguir resume a estrutura principal dos blocos de construção de telemetria orientada a modelos:

#### Função

Representação de dados  
Codificação

#### Componentes

Modelos de dados YANG  
Autodescrição de GPB / GPB

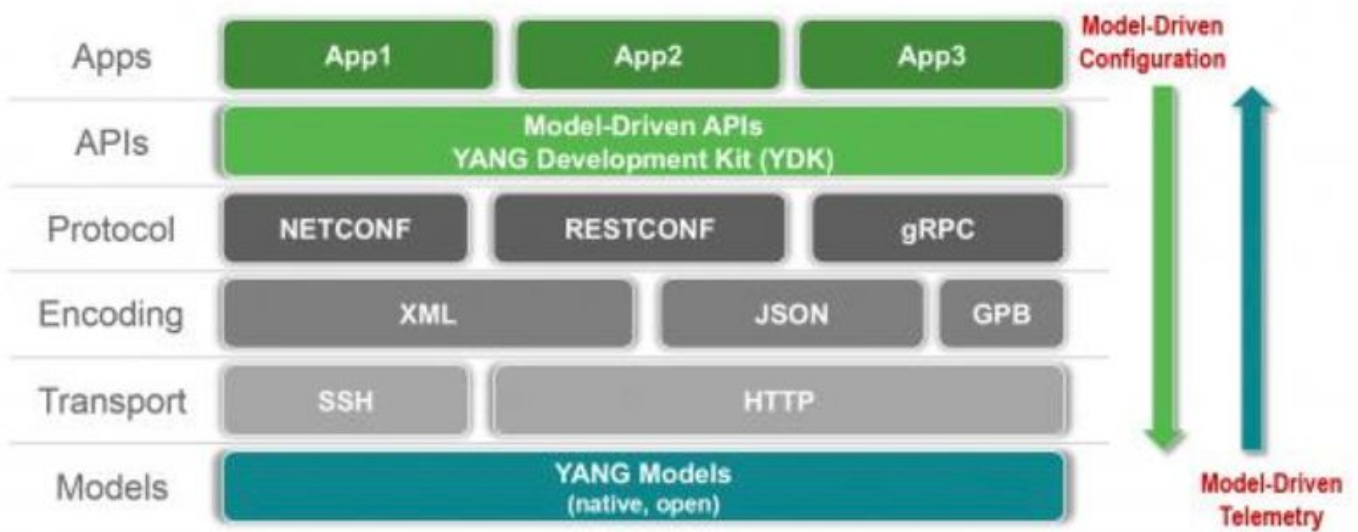
Tabela 3 Blocos de construção de telemetria

### Componentes da telemetria

Antes de entender como a telemetria e as partes de configuração subjacentes funcionam, é importante entender os diferentes componentes da telemetria para avaliar uma configuração ideal. A telemetria depende da pilha de programabilidade IOS XR, onde uma nova estrutura de infraestrutura fornece os recursos essenciais para a automação da rede.

A YANG recentemente se tornou um padrão para modelagem de dados, e isso é usado pela pilha de programabilidade da Cisco para formar um conjunto de dados estruturado que pode ser codificado e transportado o mais rápido possível pela rede. A flexibilidade da YANG oferece a grande vantagem de ser usada também como uma ferramenta de configuração para processos de automação. Esses modelos de dados são combinados com formatos de codificação específicos e protocolos de transporte para tornar o MDT uma solução completa para a análise de rede.

Para a configuração da telemetria orientada a modelos, o modelo de dados YANG se torna um componente crucial para permitir o fluxo de dados necessário para coleta e análise.



Pilha de programabilidade IOS XR

### YANG

Yang é definido como "linguagem de modelagem de dados usada para modelar dados de configuração, dados de estado e notificações para protocolos de gerenciamento de rede". Devido à sua natureza dissociada de uma arquitetura típica de linguagem de programação, o YANG pode ser implementado para interagir com uma grande variedade de ferramentas.

A estrutura de dados de modelagem YANG é construída em torno do conceito de módulos e submódulos que define uma hierarquia de dados em uma árvore como a moda, que pode ser usada para várias operações, incluindo ações de configuração e tratamento de notificação.

Há várias fontes de modelos YANG disponíveis para uso, das quais três abaixo são consideradas primárias:

- Modelos nativos/específicos da Cisco
- OpenConfig
- IETF

**Modelos específicos da Cisco:** Eles também são chamados de **modelo nativo** e são publicados por vários fornecedores de dispositivos, incluindo a Cisco. por exemplo, Cisco-IOS-XR-ptp-oper.yang

**Modelos OpenConfig:** O OpenConfig é um grupo de trabalho informal de operadores de rede. O OpenConfig define modelos YANG comuns que todos os fornecedores devem suportar para configurar recursos de missão crítica. por exemplo, openconfig-interfaces.yang

**modelos IETF:** O IETF também define alguns Módulos YANG comuns que descrevem a configuração básica para interfaces, QOS e definem outros tipos de dados comuns (como Ipv4, IPv6, etc.). por exemplo, ietf-syslog-types.yang

A Cisco oferece suporte aos modelos Openconfig disponíveis. Os fornecedores estão convergindo para uma forma padronizada de modelar dados para oferecer suporte a um ambiente de vários fornecedores.

Existem **três tipos** de modelos Yang:

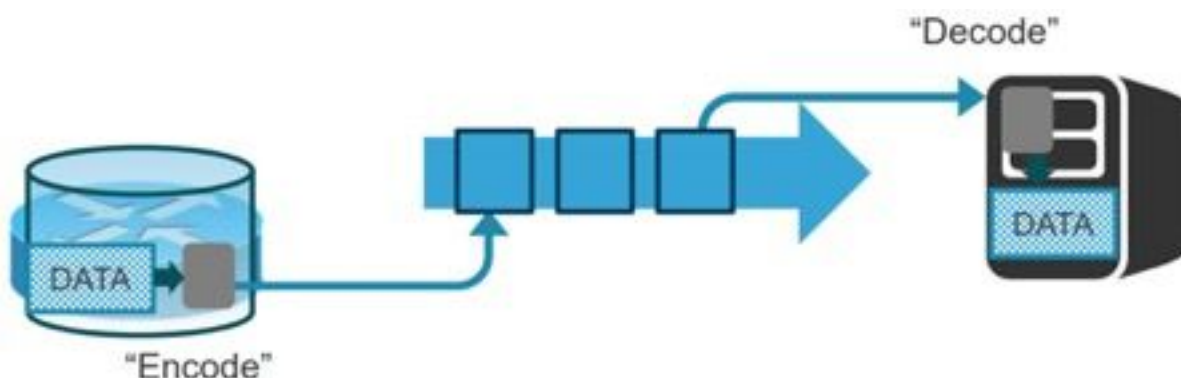
1. Operacional
2. Configuração
3. Ação

A telemetria só se preocupa com os **modelos Operacionais Yang** que podem ser identificados como **\*-oper-\*.yang**.

O YANG é definido no RFC 7950: <https://tools.ietf.org/html/rfc7950>.

## Codificação

A codificação (ou "serialização") converte dados (objetos, estado) em um formato que pode ser transmitido pela rede. Quando o receptor decodifica ("desserializa") os dados, ele tem uma cópia semanticamente idêntica dos dados originais.



Durante os estágios iniciais de desenvolvimento da telemetria, o XML foi inicialmente considerado como um formato de codificação de primeira escolha devido à sua estrutura baseada em tags. O



problema, no entanto, com o XML era sua estrutura de codificação não compacta. O GPB (Google Protocol Buffers) foi finalmente adotado pela Cisco porque melhora a eficiência e a velocidade nas operações de codificação.

Há dois tipos de GPB como opções de codificação para transmissão de telemetria:

1. GPB compacta
2. GPB de autodescrição

A principal diferença entre os dois formatos de telemetria GPB é como eles representam e codificam as chaves em um fluxo de dados de telemetria.

GPB – “compact”

```
1: GigabitEthernet0/0/0/0
50: 449825
51: 41624083
52: 360333
53: 29699362
54: 91299
<snip>
```

GPB – “self-describing”

```
{InterfaceName:
GigabitEthernet0/0/0/0
GenericCounters {
PacketsSent: 449825
BytesSent: 41624083
PacketsReceived: 360333
BytesReceived: 29699362
MulticastPacketsReceived: 91299
<snip>
```

JSON é outro esquema de codificação amigável disponível que é muito fácil de entender e quase qualquer aplicativo será capaz de decodificar.

Do ponto de vista da implantação, há alguns prós e contras de um esquema de codificação. A comparação sobre vários esquemas de codificação é fornecida na seção Telemetry Design Guidelines.

## Transporte

A telemetria oferece três opções possíveis para protocolos de transporte:

- TCP
- gRPC
- UDP

A telemetria também define dois modos de iniciação diferentes para iniciar uma sessão entre o nó e o coletor:

- Discagem
- Discagem

A diferença entre os dois modos consiste apenas em como a sessão de transporte é estabelecida.

Durante as sessões de discagem externa, o dispositivo inicia a conexão enviando um pacote syn para a porta pré-configurada do servidor. Depois que a conexão é estabelecida, os fluxos de dados são imediatamente empurrados para longe do dispositivo.

Para sessões de discagem, o roteador ouve passivamente uma porta tcp esperando por uma



conexão de servidor.

No entanto, uma vez estabelecida a sessão, o roteador não é interrogado pelo próprio servidor porque o dispositivo ainda é responsável pelas operações de envio de dados. No MDT, o conceito de pesquisa de dados nem mesmo existe.

O TCP é, por padrão, o método de transporte predefinido para Telemetria, pois é confiável e muito fácil de configurar como uma opção.

O gRPC é uma estrutura moderna de código aberto projetada para ser executada em qualquer ambiente. Ele é construído sobre HTTP/2 e fornece um conjunto avançado e rico de recursos.

## Telemetria baseada em cadência vs telemetria baseada em evento

Os dados do conjunto de dados inscrito são transmitidos para o destino em um intervalo periódico configurado ou somente quando ocorre um evento. Esse comportamento é baseado em se o MDT está configurado para telemetria baseada em cadência ou telemetria baseada em evento.

A configuração para a telemetria baseada em eventos é semelhante à telemetria baseada em cadência, com apenas o intervalo de amostra como o diferenciador. Configurar o valor do intervalo de amostra como zero define a assinatura para telemetria baseada em eventos, enquanto configurar o intervalo para qualquer valor diferente de zero define a assinatura para telemetria baseada em cadência.

É recomendável usar a telemetria orientada a eventos para eventos relacionados a alterações.

## Diretrizes de design de telemetria

Conforme explicado, há muitos componentes na pilha de telemetria. Aqui estão algumas diretrizes a serem consideradas durante a implementação da telemetria em dispositivos XR.

### Como selecionar o esquema de codificação

Conforme mencionado, a codificação ou a serialização converte dados (objetos, estado) em um formato que pode ser transmitido pela rede. Quando o receptor decodifica ou desserializa os dados, ele tem uma cópia semanticamente idêntica dos dados originais.

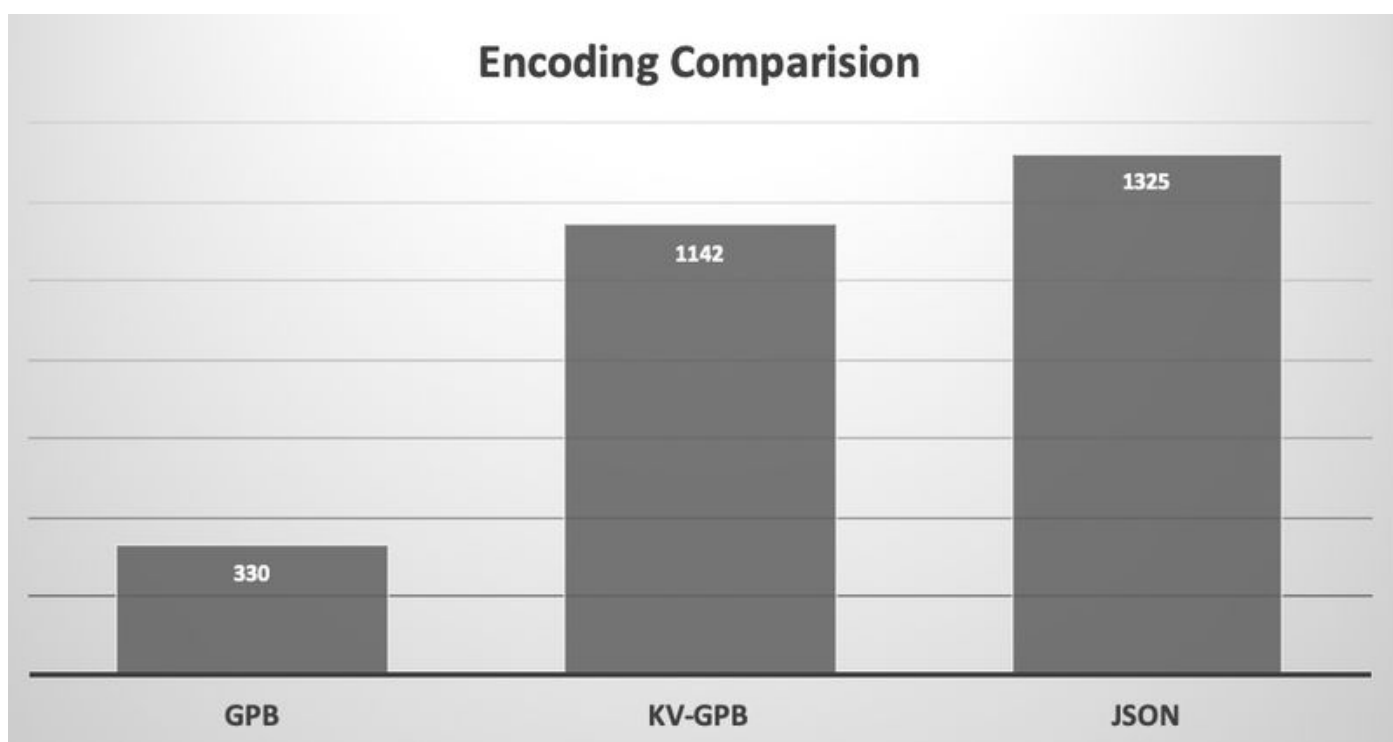
Várias opções de codificação variam em termos de eficiência de fio e facilidade de uso.

Codificação	Breve descrição	Eficiência com fio	Outra consideração
GPB (Compacto)	Tudo Binário (Exceto os valores que são strings) 2X mais rápido, operacionalmente mais complexo (mas não em relação ao SNMP)	Alto	Arquivo de proto por modelo

GPB - KV (Par chave-valor)	Chaves de string e valores binários (exceto valores que são strings) 3X Maior, Modelos nativos: ainda precisam de heurística para nomes de chave	Média a Baixa	Arquivo .proto único para decodificação
JSON	Cadeias de Caracteres Totais: Chaves e Valores	Baixa	Amigável. Legível por pessoas, amigável com aplicativos e fácil de analisar

O GPB-KV fornece um ponto médio bom e equilibrado para o esquema de codificação.

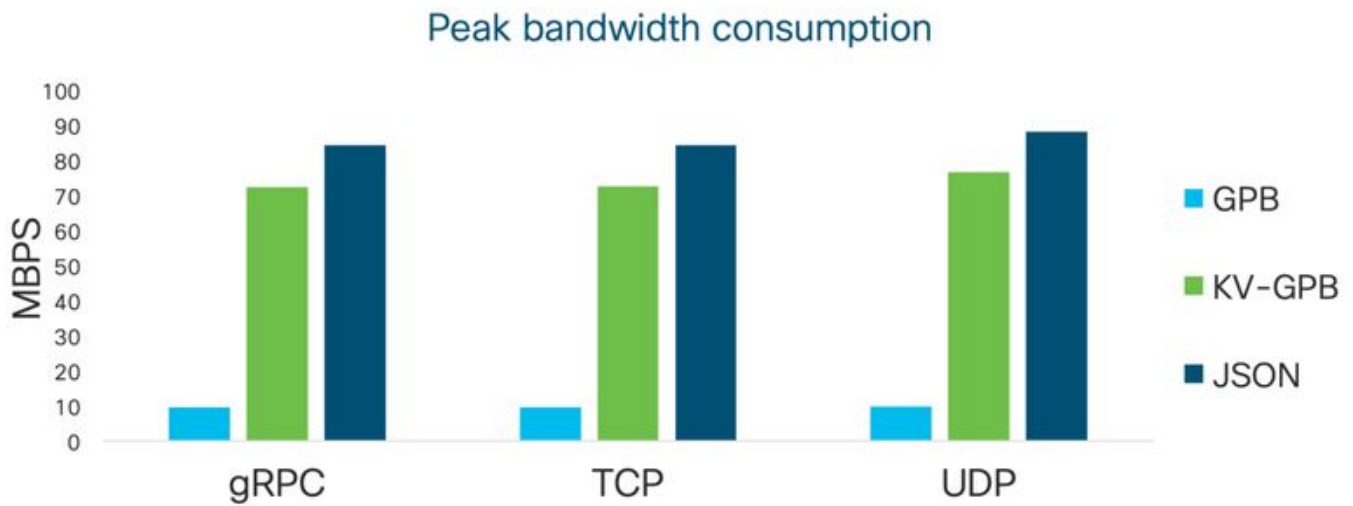
Com relação ao comprimento da mensagem para o esquema de codificação escolhido, abaixo está a comparação no fio.



Comparação de Codificação - Comprimento da Mensagem em bytes

## Consideração de projeto de rede de transporte

Opções de codificação diferentes representam requisitos de largura de banda diferentes. Ao considerar a telemetria, o operador de rede precisa cuidar do provisionamento de largura de banda suficiente de acordo com o esquema de codificação escolhido. Só para ter uma boa ideia, abaixo do consumo de largura de banda por comparação de esquema de codificação.



Comparação de largura de banda de rede

A Cisco recomenda o uso de KV-GPB. Ele funciona como um bom ponto de equilíbrio entre eficiência e conveniência.

## Avaliando opções de configuração de telemetria

Ao configurar a Telemetria orientada por modelo, o operador deve entender todos os diferentes componentes envolvidos na Telemetria. Com base nas opções disponíveis para transporte, codificação e direção de fluxo conforme detalhado acima, é possível escolher ainda mais a combinação mais adequada a um ambiente.

Os quatro componentes principais são:

1. Transporte
2. Codificação
3. Direção da sessão
4. Modelos de dados YANG

**Transporte:** Conforme mencionado, o nó pode fornecer dados de telemetria usando TCP, UDP ou gRPC sobre HTTP/2.

Embora o TCP seja a opção preferencial para a simplicidade, o gRPC oferece o recurso TLS opcional, que pode ser considerado como um benefício adicional do ponto de vista da segurança.

**Codificação:** O roteador pode fornecer dados de telemetria em dois tipos diferentes de Buffers de Protocolo do Google: Compacto e autodescritivo GPB.

Compact GPB é a codificação mais eficiente, mas requer um .proto exclusivo para cada modelo YANG que é transmitido. A autodescrição GPB é menos eficiente, mas usa um único arquivo .proto para decodificar todos os modelos YANG porque as chaves são passadas como strings no .proto.

**Direção da sessão:** Há duas opções para início de sessão na implantação de telemetria. O roteador pode discar para o coletor ou o coletor pode discar para o roteador.

**A YANG** é o padrão aceito pelo setor para modelagem de dados e a pilha de programabilidade da Cisco também a utiliza para formar um conjunto de dados estruturado que pode ser codificado e

transportado o mais rápido possível pela rede.

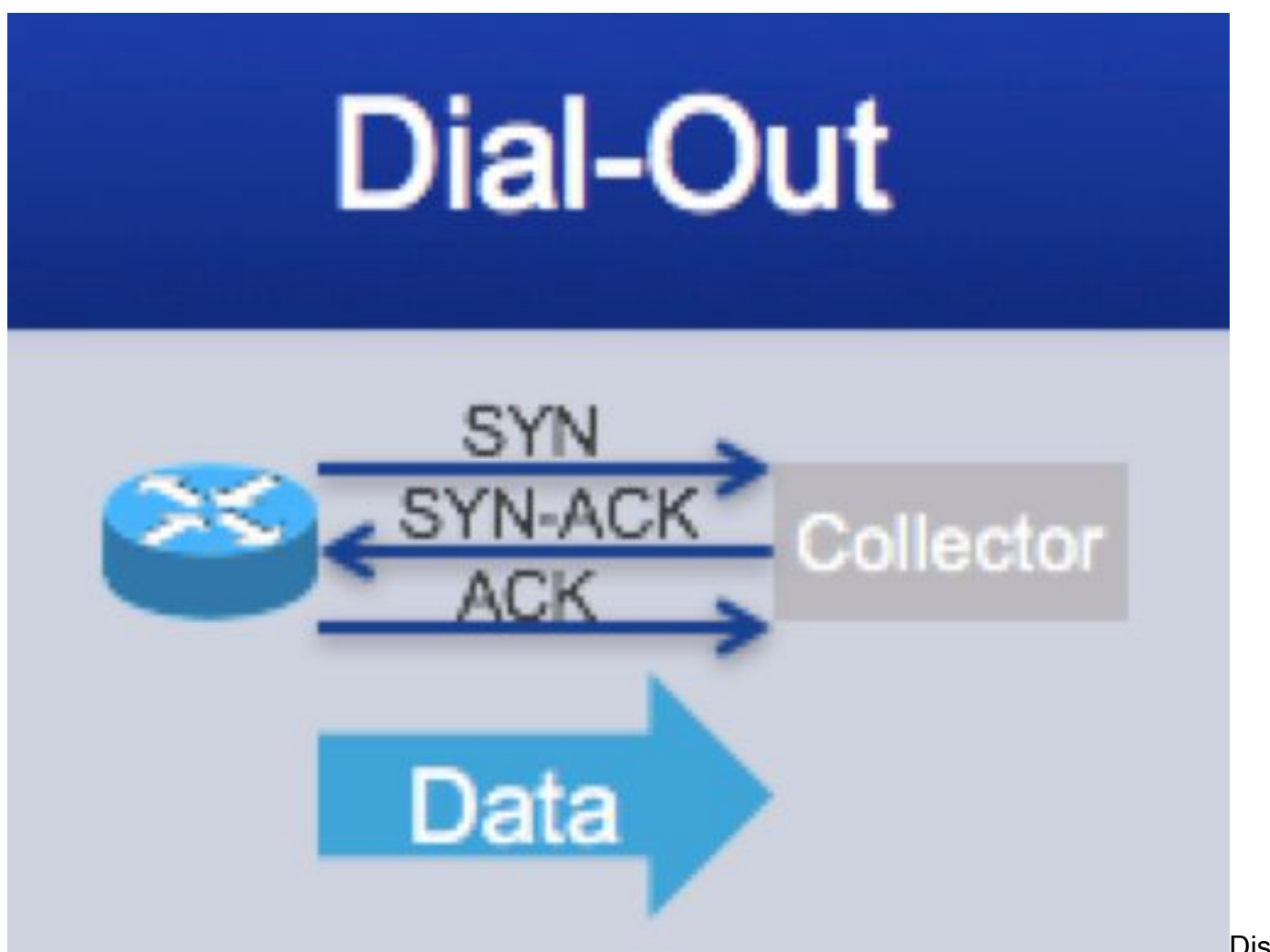
Esses modelos de dados, juntamente com formatos de codificação específicos e protocolos de transporte, conforme discutido acima, tornam a Telemetria Acionada por Modelos (MDT - Model Driven Telemetry) uma solução completa para Analíticos.

## Exemplos de configuração de telemetria

### IOS-XR

#### Divisão da Configuração de Discagem Externa

No modo de discagem externa, o roteador é responsável por iniciar uma sessão TCP para o coletor e envia dados que são especificados pelo grupo de sensores na assinatura.



Do ponto de vista da configuração, a configuração de telemetria é um processo de três etapas. Primeiro, identificamos as informações que queremos transmitir e as capturamos na configuração do Sensor Group. Em segundo lugar, identificamos o destino para o qual as informações devem ser transmitidas e as capturamos na configuração do grupo de destino. Em terceiro lugar, usamos as informações identificadas nas duas etapas anteriores para configurar a assinatura real.

1. Definir grupos de sensores
2. Definir grupos de destinos

### 3. Definir Assinatura

#### Definir grupos de sensores

A configuração do grupo Sensor identifica as informações que devem ser transmitidas. O modelo de configuração a seguir fornece a configuração necessária para configurar grupos de sensores.

```
RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)#telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#sensor-group <Sensor-Group-Name>
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# sensor-path <Sensor-Path>
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# commit
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# end
RP/0/RP0/CPU0:XR#
```

O exemplo a seguir mostra um exemplo real da CLI do roteador, onde um exemplo real da configuração do grupo de sensores:

```
RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)#telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#sensor-group SensorGroup101
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# sensor-path Cisco-IOS-XR-infra-statsd-
oper:infra-statistics/interfaces/interface/latest/generic-counters
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# commit
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# end
RP/0/RP0/CPU0:XR#
```

Podemos ter vários Caminhos de Sensor como parte da mesma definição de SensorGroup:

```
RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)#telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#sensor-group SensorGroup101
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# sensor-path sensor-path Cisco-IOS-XR-infra-
statsd-oper:infra-statistics/interfaces/interface/data-rate
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# sensor-path Cisco-IOS-XR-infra-statsd-
oper:infra-statistics/interfaces/interface/latest/generic-counters
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# commit
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# end
RP/0/RP0/CPU0:XR#
```

#### Definir grupos de destinos

A configuração do grupo Destino identifica o destino para o qual as informações devem ser transmitidas.

Ele tem três parâmetros principais

1. Direção da sessão
2. Codificação a ser usada
3. Protocolo de transporte a ser usado

A seguir, está um exemplo:

```
RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)# telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)# destination-group DestGroup101
RP/0/RP0/CPU0:XR(config-model-driven-dest)# address family ipv4 10.1.1.1 port 5432
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)# encoding self-describing-gpb
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)# protocol tcp
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)# commit
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# end
RP/0/RP0/CPU0:XR#
```

## Definir a Assinatura

A Inscrição vincula as informações do Sensor-Group e do Destination-Group como a parte final da configuração. O intervalo de exemplo é definido como parte da Assinatura.

A seguir, está um exemplo:

```
RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#subscription Subscription101
RP/0/RP0/CPU0:XR(config-model-driven-subs)#sensor-group-id SensorGroup101 sample-interval 30000
RP/0/RP0/CPU0:XR(config-model-driven-subs)#destination-id DestGroup101
RP/0/RP0/CPU0:XR(config-model-driven-subs)# commit
RP/0/RP0/CPU0:XR(config-model-driven-subs)# end
RP/0/RP0/CPU0:XR#
```

## Exemplo de configuração completa

```
RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#conf
RP/0/RP0/CPU0:XR(config)#
RP/0/RP0/CPU0:XR(config)#telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#sensor-group SensorGroup101
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# sensor-path Cisco-IOS-XR-infra-statsd-
oper:infra-statistics/interfaces/interface/latest/generic-counters
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)#destination-group DestGroup101
RP/0/RP0/CPU0:XR(config-model-driven-dest)#address family ipv4 10.1.1.2 port 5432
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)#encoding self-describing-gpb
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)#protocol tcp
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)#subscription Subscription101
RP/0/RP0/CPU0:XR(config-model-driven-subs)#sensor-group-id SensorGroup101 sample-interval 30000
RP/0/RP0/CPU0:XR(config-model-driven-subs)#destination-id DestGroup101
RP/0/RP0/CPU0:XR(config-model-driven-subs)#commit
RP/0/RP0/CPU0:XR(config-model-driven-subs)#end
RP/0/RP0/CPU0:XR#
```

## Vantagens da discagem externa

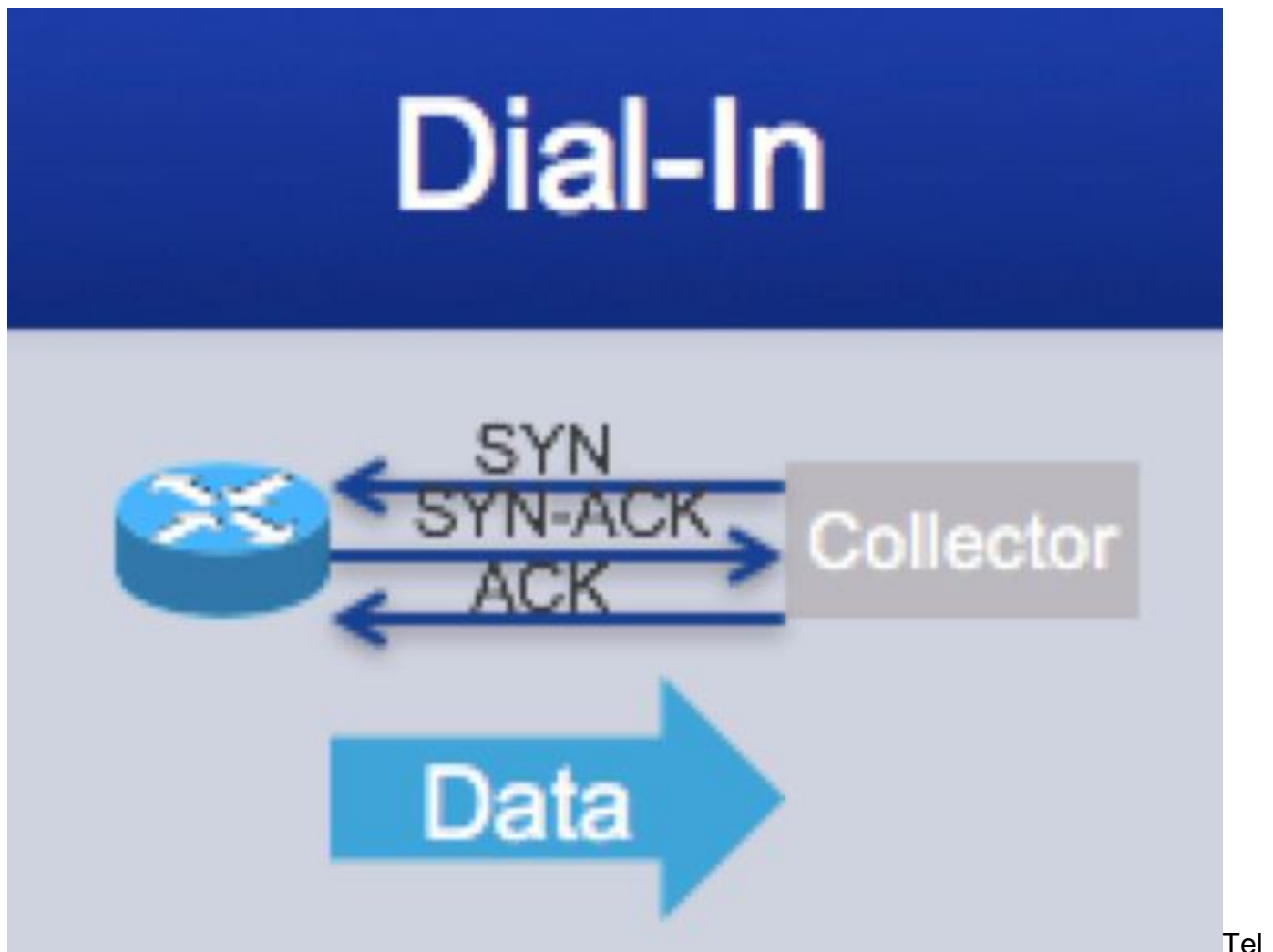
- Maior flexibilidade para opções de transporte.
- Não há necessidade de abrir portas para tráfego de gerenciamento de entrada.

- Anycast e balanceamento de carga.

## Separação da Configuração de Discagem

No modo de discagem, um coletor/receptor/orquestrador MDT discagem para o roteador e se inscreve dinamicamente em um ou mais caminhos ou assinaturas do sensor. O roteador atua como o servidor e o cliente como o receptor.

Apenas uma única sessão é formada e o roteador transmite dados de telemetria através da mesma sessão. Essa assinatura dinâmica é encerrada quando o receptor cancela a assinatura ou quando a sessão é encerrada.



emetry de discagem

Como o coletor "disca" para o roteador, não há necessidade de especificar cada destino MDT na configuração. Basta ativar o serviço gRPC no roteador, conectar o cliente e ativar dinamicamente a assinatura de telemetria desejada.

Do ponto de vista da configuração, a configuração de telemetria é um processo de três etapas semelhante ao descrito acima. Primeiro, precisamos habilitar o gRPC. Em segundo lugar, identificamos o destino para o qual as informações devem ser transmitidas e as capturamos na configuração do grupo de sensores. Em terceiro lugar, usamos as informações identificadas nas duas etapas anteriores para configurar a assinatura real.

1. Habilitar gRPC



2. Definir grupos de sensores

3. Definir Assinatura

[Clique para expandir](#)

O modo de discagem só tem suporte com gRPC

O modo de discagem só tem suporte com gRPC

## Habilitar gRPC

Primeiro, precisamos habilitar o servidor gRPC no roteador para aceitar conexões de entrada do coletor.

[Clique para expandir](#)

- O intervalo <port-number> vai de 57344 a 57999. Se um número de porta não estiver disponível, um erro será exibido.

O intervalo <port-number> vai de 57344 a 57999. Se um número de porta não estiver disponível, um erro será exibido.

```
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)# grpc
RP/0/RP0/CPU0:XR(config-grpc)#port 57890
RP/0/RP0/CPU0:XR(config-grpc)#commit
RP/0/RP0/CPU0:XR(config-grpc)#end
RP/0/RP0/CPU0:XR#
```

## Definir grupos de sensores

A configuração do grupo Sensor identifica as informações que devem ser transmitidas. O modelo de configuração a seguir fornece a configuração necessária para configurar grupos de sensores.

O exemplo a seguir mostra um exemplo real da CLI do roteador, onde um exemplo real da configuração do grupo de sensores

```
RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)#telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#sensor-group SensorGroup101
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# sensor-path openconfig-
interfaces:interfaces/interface
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# commit
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# end
RP/0/RP0/CPU0:XR#
```

## Definir a Assinatura

A Assinatura vincula o Sensor-Group e o gRPC como a parte final da configuração. O intervalo de exemplo é definido como parte da Assinatura.

O modelo de configuração a seguir fornece a configuração necessária para configurar a assinatura.

O exemplo a seguir mostra um exemplo real da CLI do roteador em que estamos criando uma assinatura e vinculando os grupos de sensor e destino juntos, além de definir a taxa de amostra.

```
RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#subscription Subscription101
RP/0/RP0/CPU0:XR(config-model-driven-subs)#sensor-group-id SensorGroup101 sample-interval 30000
RP/0/RP0/CPU0:XR(config-model-driven-subs)# commit
RP/0/RP0/CPU0:XR(config-model-driven-subs)# end
RP/0/RP0/CPU0:XR#
```

## Modelo de configuração completo e exemplo

```
RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)# grpc
RP/0/RP0/CPU0:XR(config-grpc)#port 57890
RP/0/RP0/CPU0:XR(config-grpc)telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#subscription Subscription101
RP/0/RP0/CPU0:XR(config-model-driven-subs)#sensor-group-id SensorGroup101 sample-interval 30000
RP/0/RP0/CPU0:XR(config-model-driven-subs)# commit
RP/0/RP0/CPU0:XR(config-model-driven-subs)# end
RP/0/RP0/CPU0:XR#
```

## Vantagens da discagem

- Um único canal para configuração e transmissão
- Porta de escuta no roteador/dispositivo
- Conexão Transitória
- Atualmente, somente gRPC/gNMI está disponível

## Telemetria orientada a eventos

Na telemetria orientada a eventos, os dados do conjunto de dados inscrito são transmitidos somente quando ocorre um evento.

## Configuração de telemetria orientada a eventos

A configuração da telemetria baseada em eventos é semelhante à telemetria baseada em cadência, com a única diferença na configuração da telemetria orientada a eventos é a do intervalo de amostra. Configurar o valor do intervalo de amostra como zero define a assinatura para telemetria baseada em eventos.

## Modelo de Configuração Completo e Exemplo para DISCAGEM

```
RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#conf
RP/0/RP0/CPU0:XR(config)#
RP/0/RP0/CPU0:XR(config)#telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#sensor-group <Sensor-Group-Name>
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# sensor-path <Sensor-Path>
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)#destination-group <Destination-Group-Name>
RP/0/RP0/CPU0:XR(config-model-driven-dest)#address family ipv4 <Destination-IP> port
```

```

<Destination-Port>
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)#encoding <Encoding-Type>
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)#protocol <Transport-Protocol>
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)#subscription <Subscription-Name>
RP/0/RP0/CPU0:XR(config-model-driven-subs)#sensor-group-id <Sensor-Group-Name> sample-interval
<0>
RP/0/RP0/CPU0:XR(config-model-driven-subs)#destination-id <Destination-Group-Name>
RP/0/RP0/CPU0:XR(config-model-driven-subs)#commit
RP/0/RP0/CPU0:XR(config-model-driven-subs)#end
RP/0/RP0/CPU0:XR#

```

O exemplo a seguir mostra um exemplo real da CLI do roteador.

```

RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#conf
RP/0/RP0/CPU0:XR(config)#
RP/0/RP0/CPU0:XR(config)#telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#sensor-group SensorGroup101
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)# sensor-path Cisco-IOS-XR-infra-statsd-
oper:infra-statistics/interfaces/interface/latest/generic-counters
RP/0/RP0/CPU0:XR(config-model-driven-snsr-grp)#destination-group DestGroup101
RP/0/RP0/CPU0:XR(config-model-driven-dest)#address family ipv4 10.1.1.2 port 5432
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)#encoding self-describing-gpb
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)#protocol tcp
RP/0/RP0/CPU0:XR(config-model-driven-dest-addr)#subscription Subscription101
RP/0/RP0/CPU0:XR(config-model-driven-subs)#sensor-group-id SensorGroup101 sample-interval 0
RP/0/RP0/CPU0:XR(config-model-driven-subs)#destination-id DestGroup101
RP/0/RP0/CPU0:XR(config-model-driven-subs)#commit
RP/0/RP0/CPU0:XR(config-model-driven-subs)#end
RP/0/RP0/CPU0:XR#

```

## Modelo de Configuração Completo e Exemplo para DISCAGEM

```

RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)# grpc
RP/0/RP0/CPU0:XR(config-grpc)#port <port-number>
RP/0/RP0/CPU0:XR(config-grpc)telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#subscription <Subscription-Name>
RP/0/RP0/CPU0:XR(config-model-driven-subs)#sensor-group-id <Sensor-Group-Name> sample-interval
<0>
RP/0/RP0/CPU0:XR(config-model-driven-subs)#commit
RP/0/RP0/CPU0:XR(config-model-driven-subs)#end
RP/0/RP0/CPU0:XR#

```

O exemplo a seguir mostra um exemplo real da CLI do roteador.

```

RP/0/RP0/CPU0:XR#
RP/0/RP0/CPU0:XR#config
RP/0/RP0/CPU0:XR(config)# grpc
RP/0/RP0/CPU0:XR(config-grpc)#port 57890
RP/0/RP0/CPU0:XR(config-grpc)telemetry model-driven
RP/0/RP0/CPU0:XR(config-model-driven)#subscription Subscription101
RP/0/RP0/CPU0:XR(config-model-driven-subs)#sensor-group-id SensorGroup101 sample-interval 0
RP/0/RP0/CPU0:XR(config-model-driven-subs)# commit
RP/0/RP0/CPU0:XR(config-model-driven-subs)# end
RP/0/RP0/CPU0:XR#

```

# Validando a telemetria com comandos SHOW

Do ponto de vista do roteador, podemos verificar os parâmetros configurados para cada grupo de sensores, grupo de destino e assinatura

```
// ALL CONFIGURED SUBSCRIPTIONS
```

```
RP/0/RP0/CPU0:XR#show telemetry model-driven subscription
```

```
Subscription: Subscription101          State: ACTIVE
```

```
-----
```

```
Sensor groups:
```

Id	Interval (ms)	State
SensorGroup101	30000	<b>Resolved</b>

```
Destination Groups:
```

Id	Encoding	Transport	State	Port	IP
DestGroup101	self-describing-gpb	tcp	Active	5432	172.16.128.3

```
// DETAILS ON A PARTICULAR SUBSCRIPTION
```

```
RP/0/RP0/CPU0:XR#show telemetry model-driven subscription Subscription101
```

```
Subscription: Subscription101
```

```
-----
```

```
State: ACTIVE
```

```
Sensor groups:
```

```
Id: SensorGroup101
```

```
Sample Interval: 30000 ms
```

```
Sensor Path: Cisco-IOS-XR-infra-statsd-oper:infra-  
statistics/interfaces/interface/latest/generic-counters
```

```
Sensor Path State: Resolved
```

```
Destination Groups:
```

```
Group Id: DestGroup101
```

```
Destination IP: 172.16.128.3
```

```
Destination Port: 5432
```

```
Encoding: self-describing-gpb
```

```
Transport: tcp
```

```
State: Active
```

```
Total bytes sent: 4893
```

```
Total packets sent: 1
```

```
Last Sent time: 2019-11-01 10:04:11.2378949664 +0000
```

```
Collection Groups:
```

```
-----
```

```
Id: 1
```

```
Sample Interval: 30000 ms
```

```
Encoding: self-describing-gpb
```

```
Num of collection: 5
```

```
Collection time: Min: 6 ms Max: 29 ms
```

```
Total time: Min: 6 ms Avg: 12 ms Max: 29 ms
```

```
Total Deferred: 0
```

```
Total Send Errors: 0
```

```
Total Send Drops: 0
```

```
Total Other Errors: 0
```

```
Last Collection Start: 2019-11-01 10:06:11.2499006664 +0000
```

```
Last Collection End: 2019-11-01 10:06:11.2499006664 +0000
```

```
Sensor Path: Cisco-IOS-XR-infra-statsd-oper:infra-
```

statistics/interfaces/interface/latest/generic-counters

RP/0/RP0/CPU0:XR#

// ALL CONFIGURED DESTINATIONS

RP/0/RP0/CPU0:XR#show telemetry model-driven destination

Group Id	IP	Port	Encoding	Transport	State
DestGroup101	172.16.128.3	5432	self-describing-gpb	tcp	Active

RP/0/RP0/CPU0:XR#

// PARTICULAR DESTINATION

RP/0/RP0/CPU0:XR#show telemetry model-driven destination DestGroup101

Destination Group: DestGroup101

-----

Destination IP: 172.16.128.3  
Destination Port: 5432  
State: Active  
Encoding: self-describing-gpb  
Transport: tcp  
**Total bytes sent: 83181**  
**Total packets sent: 17**  
Last Sent time: 2019-11-01 10:12:11.2859133664 +0000

Collection Groups:

-----

Id: 1  
Sample Interval: 30000 ms  
Encoding: self-describing-gpb  
**Num of collection: 17**  
Collection time: Min: 5 ms Max: 29 ms  
Total time: Min: 6 ms Max: 29 ms Avg: 10 ms  
Total Deferred: 0  
Total Send Errors: 0  
Total Send Drops: 0  
Total Other Errors: 0  
Last Collection Start: 2019-11-01 10:12:11.2859128664 +0000  
Last Collection End: 2019-11-01 10:12:11.2859134664 +0000  
Sensor Path: Cisco-IOS-XR-infra-statsd-oper:infra-  
statistics/interfaces/interface/latest/generic-counters

RP/0/RP0/CPU0:XR#

// ALL CONFIGURED SENSOR GROUPS

RP/0/RP0/CPU0:XR#show telemetry model-driven sensor-group

Sensor Group Id: SensorGroup101

Sensor Path: Cisco-IOS-XR-infra-statsd-oper:infra-  
statistics/interfaces/interface/latest/generic-counters  
Sensor Path State: Resolved

// PARTICULAR SENSOR GROUPS

RP/0/RP0/CPU0:XR#show telemetry model-driven sensor-group SensorGroup101

Sensor Group Id: SensorGroup101

Sensor Path: Cisco-IOS-XR-infra-statsd-oper:infra-  
statistics/interfaces/interface/latest/generic-counters  
Sensor Path State: Resolved

RP/0/RP0/CPU0:XR#

## Pilha de Coleção de Telemetria

Além da configuração do roteador, uma solução baseada em telemetria exigia vários componentes, como coletor, banco de dados e software de monitoramento/análise. Esses componentes podem ser configurados separadamente ou podem fazer parte de um único produto abrangente.

- Um decodificador deve ser instalado para receber pacotes de entrada e passá-los adiante para armazenamento adicional.
- É necessário um banco de dados Time Series (também conhecido como TSDB) para armazenar informações transmitidas.
- Uma ferramenta gráfica também é necessária para visualizar dados extraídos do banco de dados interno.

Está além do escopo descrever a pilha de coleta em detalhes. O Cisco Crossworks Health Insights permite telemetria automatizada, na qual os dispositivos são provisionados automaticamente com a configuração de telemetria e as tabelas/esquemas são criados em um TSDB (Time Series Database, banco de dados de série temporal). Ele simplifica a sobrecarga operacional e de gerenciamento de rede da coleta e limpeza de dados, permitindo que as operadoras se concentrem em seus objetivos comerciais. A utilização de um coletor comum para coletar dados do dispositivo de rede por SNMP, CLI e telemetria orientada por modelo permite evitar a duplicação de dados e também reduz a carga nos dispositivos e na rede.

## Considerações de implantação para telemetria em uma rede

Veja a seguir várias considerações a serem feitas durante a análise da implantação de telemetria em uma rede.

### Dimensionamento

A telemetria pode transmitir uma quantidade considerável de dados e recomenda-se uma consideração cuidadosa dos aspectos de escalabilidade.

### Transmitir apenas os dados necessários

Cada modelo Yang terá vários nós de folha. É aconselhável ser específico sobre as informações que são necessárias e as informações que não são necessárias. Recomenda-se explorar os modelos Yang e identificar o caminho de dados exigido pelos casos de uso de telemetria.

### Considere a quantidade de dados de transmissão

A quantidade total de dados de telemetria sendo transmitida precisará considerar os seguintes pontos:

1. Alocação de largura de banda dentro da rede
2. qos
3. Desempenho de aplicativos/software adicionais, como Software coletor Banco de dados de Série Temporal Software de análise/visualização
4. A eficiência da codificação (discutida na Seção "Diretrizes de design da telemetria") afeta diretamente a quantidade de dados que está sendo transmitida. A GPB compacta é recomendada sempre que possível.
5. O intervalo de coleta afeta diretamente vários aspectos, incluindo, mas não se limitando a

seguir. Utilização da largura de banda na redeRequisito de armazenamento no banco de dadosDesempenho do dispositivo que transmite os dados

Recomenda-se avaliar a frequência da coleta com base nos requisitos do aplicativo.

Em geral, recomenda-se considerar a filtragem de dados indesejados na origem ou no destino como possível. Temos a opção de filtrar dados indesejados. A filtragem pode ser executada em dois níveis: -

1. Na origem - Os dispositivos que transmitem os dados.
2. No destino - O coletor coleta e normaliza dados. (A filtragem no coletor está além do escopo deste documento)

O exemplo a seguir mostra dados sendo filtrados somente para interfaces Hundred Gig dentro dos caminhos do sensor, aplicando curingas.

```
sensor-path Cisco-IOS-XR-infra-statsd-oper:infra-statistics/interfaces/interface[interface-name='HundredGigE*']/latest/generic-counters
```

## Referências

<https://blogs.cisco.com/sp/the-limits-of-snmp>

<https://blogs.cisco.com/sp/why-you-should-care-about-model-driven-telemetry>

<https://www.cisco.com/c/en/us/td/docs/iosxr/asr9000/telemetry/b-telemetry-cg-asr9000-61x.html>



## Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.