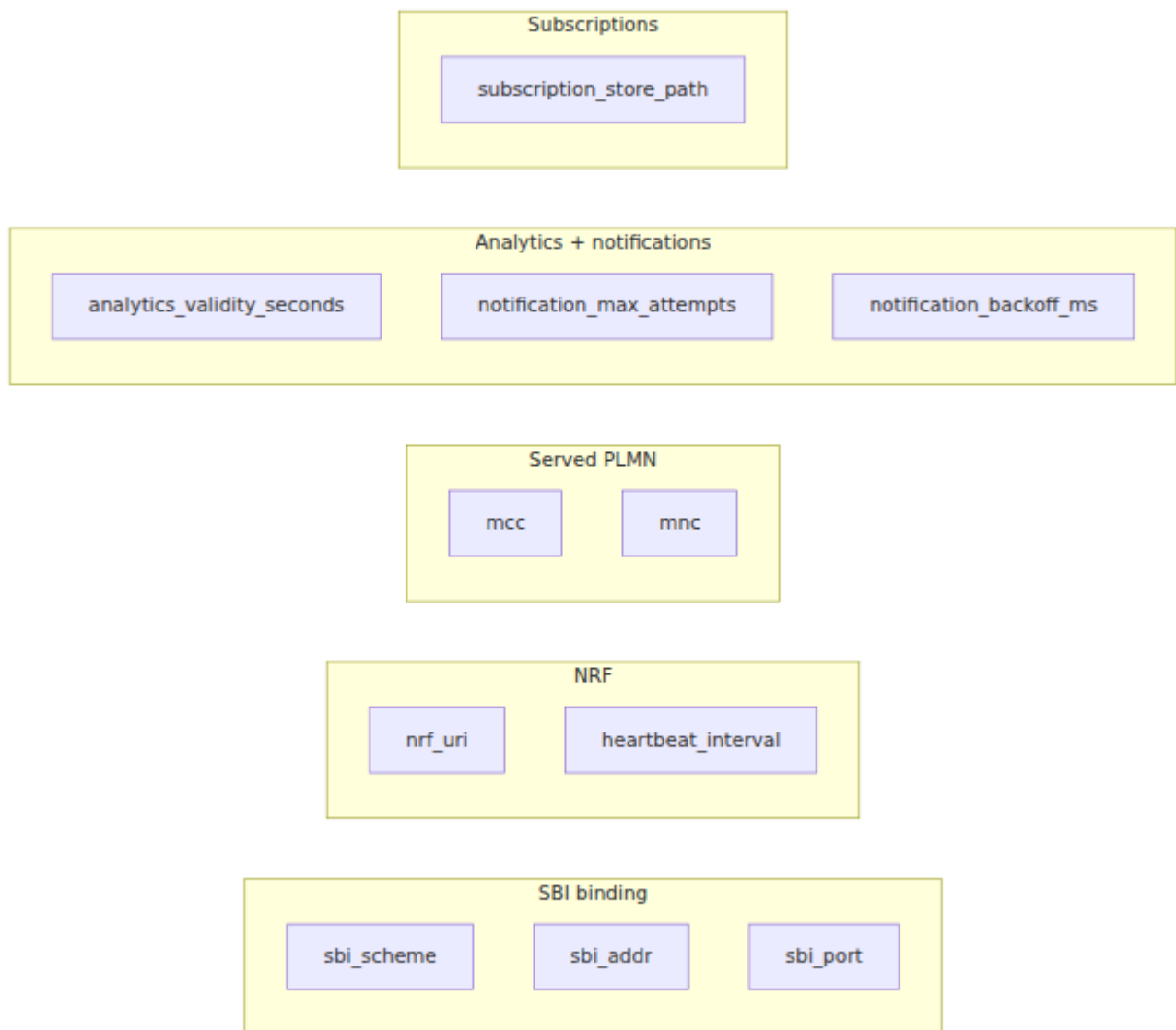


Referência de Configuração do OmniNWDAF

[Visão Geral](#) | [Operações](#) | [Métricas](#) | [Solução de Problemas](#)

Esta referência documenta cada parâmetro de configuração do OmniNWDAF. Todos os parâmetros estão sob o escopo da aplicação `:omninwdaf`. Variáveis de ambiente se aplicam apenas na configuração de tempo de execução de produção (`runtime.exs`); os ajustes que são apenas de arquivo não têm variável de ambiente e devem ser definidos no arquivo de configuração da aplicação.

Estrutura da Configuração



Exemplo Completo de Configuração

```
config :omninwdaf,  
  # SBI binding e endpoint anunciado  
  sbi_scheme: "http",  
  sbi_addr: "127.0.0.23",  
  sbi_port: 7777,  
  
  # Alvo de registro do NRF e cadência de heartbeat  
  nrf_uri: "http://127.0.0.10:7777",  
  heartbeat_interval: 10_000,  
  
  # PLMN servido (substitua o padrão de teste 999-70 por  
  # implantação)  
  mcc: "999",  
  mnc: "70",  
  
  # Janela de validade aplicada à expiração de analytics  
  # retornados (segundos)  
  analytics_validity_seconds: 3600,  
  
  # Ajuste de tentativas de entrega de notificação  
  notification_max_attempts: 4,  
  notification_backoff_ms: 500,  
  
  # Arquivo de armazenamento de assinatura durável (padrões sob o  
  # diretório temporário do sistema)  
  subscription_store_path: "/var/lib/omninwdaf/subscriptions.dets"
```

O `runtime.exe` de produção lê o subconjunto baseado em ambiente das variáveis de ambiente, revertendo para os padrões abaixo:

```
config :omninwdaf,  
  sbi_scheme: System.get_env("NWDAF_SBI_SCHEME", "http"),  
  sbi_addr: System.get_env("NWDAF_SBI_ADDR", "127.0.0.23"),  
  sbi_port: String.to_integer(System.get_env("NWDAF_SBI_PORT",  
"7777")),  
  nrf_uri: System.get_env("NWDAF_NRF_URI",  
"http://127.0.0.10:7777"),  
  mcc: System.get_env("NWDAF_MCC", "999"),  
  mnc: System.get_env("NWDAF_MNC", "70")
```

Parâmetros de Binding do SBI

Estes controlam o endereço e a porta que o OmniNWDAF utiliza para seus serviços SBI, e o esquema que ele anuncia ao NRF e aos consumidores.

Parâmetro	Tipo	Requerido	Padrão	Var. Env	
sbi_scheme	String	Não	http	NWDAF_SBI_SCHEME	E U e e a
sbi_addr	String	Não	127.0.0.23	NWDAF_SBI_ADDR	E q O u s S a N c
sbi_port	Integer	Não	7777	NWDAF_SBI_PORT	P o O e s S

Parâmetros do NRF

Estes controlam o registro com o NRF e a cadência de heartbeat. Veja [Operações: Registro e Heartbeat do NRF](#). A interação com o NRF segue [TS 29.510](#).

Parâmetro	Tipo	Requerido	Padrão	
nrf_uri	String	Não	http://127.0.0.10:7777	(
heartbeat_interval	Integer (ms)	Não	10000	(

Parâmetros do PLMN Servido

Estes identificam o PLMN que o OmniNWDAF anuncia em seu perfil NF.

Parâmetro	Tipo	Requerido	Padrão	Var. Env	Descrição
mcc	String	Não	999	NWDAF_MCC	Código do País Móvel do PLMN servido. O padrão 999 é um valor de teste: substitua por implantação.
mnc	String	Não	70	NWDAF_MNC	Código da Rede Móvel do PLMN servido. O padrão 70 é um valor de teste: substitua por implantação.

Parâmetros de Analytics

Parâmetro	Tipo	Requerido	Padrão	Var. Env
<code>analytics_validity_seconds</code>	Integer (s)	Não	<code>3600</code>	(nenhum)

Parâmetros de Tentativa de Notificação

Estes ajustam a entrega de melhor esforço de cada notificação de assinatura de eventos (criação, recorrente e pós-modificação). Veja [Operações: Entrega e Tentativa de Notificação](#).

Parâmetro	Tipo	Requerido	Padrão	Var. Env
<code>notification_max_attempts</code>	Integer	Não	4	(nenhum)
<code>notification_backoff_ms</code>	Integer (ms)	Não	500	(nenhum)

Parâmetros do Armazenamento de Assinaturas

Estes controlam onde as assinaturas de eventos são persistidas de forma durável. As assinaturas são mantidas em uma tabela ETS em memória e espelhadas em um arquivo DETS, sendo recarregadas na reinicialização. Veja [Operações: Nnwdaf_EventsSubscription](#).

Parâmetro	Tipo	Requerido	Padrão
<code>subscription_store_path</code>	String (caminho)	Não	<code>omninwdaf_subscrip</code> sob o diretório tempo sistema

Notas sobre Parâmetros Apenas de Arquivo

`heartbeat_interval`, `analytics_validity_seconds`,
`notification_max_attempts`, `notification_backoff_ms`, e

`subscription_store_path` **não têm variável de ambiente** e são definidos no arquivo de configuração da aplicação.

Métricas e Monitoramento do OmniNWDAF

[Visão Geral](#) | [Operações](#) | [Configuração](#) | [Solução de Problemas](#)

Observabilidade: Logs Estruturados

A observabilidade é fornecida através de **logs de aplicação estruturados** e registro de solicitações padrão. Os operadores devem monitorar os logs do OmniNWDAF em busca dos seguintes sinais.

Ciclo de Vida da Assinatura

As entradas de log cobrem o ciclo de vida da assinatura de eventos:

- Criação de uma assinatura e seu id de assinatura atribuído.
- Recarga de assinaturas duráveis na inicialização, incluindo quantas foram restauradas do arquivo de armazenamento.
- Rejeição de uma assinatura inválida (falta de `notificationURI` ou evento não suportado).
- Exclusão de uma assinatura, incluindo tentativas de exclusão para um id desconhecido.

Entrega de Notificações

As entradas de log cobrem a entrega de cada notificação (tempo de criação, recorrente e pós-modificação; veja [Operações: Entrega de Notificações e Retentativa](#)):

- Cada tentativa de entrega para o `notificationURI` de um assinante.

- Cada tentativa, seguindo o cronograma de retrocesso exponencial limitado.
- A desistência final após a última tentativa, registrada como um **aviso**.

Registro de Solicitações

O registro padrão de solicitações HTTP cobre os endpoints SBI, incluindo as recuperações de AnalyticsInfo, as chamadas de criação, leitura, modificação e exclusão de assinaturas, e o endpoint de status de vivacidade. Use isso para confirmar que os consumidores estão alcançando o OmniNWDAF e para observar códigos de status de resposta, como `400`

`MANDATORY_QUERY_PARAM_MISSING`, `404 UNSUPPORTED_EVENT` e `400 MANDATORY_IE_MISSING`.

Abordagem de Monitoramento Recomendada

Até que métricas específicas de NF sejam adicionadas, monitore o OmniNWDAF através de seus logs e seu endpoint de vivacidade:

Sinal	Onde	O que isso informa
Vivacidade	GET /nnwdaf-analyticinfo/v1/status retornando 200 {"status": "UP"}	O serviço está ativo e atendendo solicitações SBI.
Registro NRF	Logs na inicialização	Se o registro com o NRF foi bem-sucedido.
Desistências de Notificações	Logs de aviso após a última tentativa de entrega	Assinantes cujo notificationURI é inacessível.
Erros de Solicitação	Logs de solicitação com status 4xx	Solicitações e assinaturas de análise malformadas ou não suportadas.

Verificação de Saúde

O endpoint de vivacidade é o alvo recomendado para a verificação de prontidão e vivacidade:

- **Endpoint:** GET /nnwdaf-analyticinfo/v1/status
- **Resposta saudável:** 200 com corpo {"status": "UP"}

Um status saudável confirma que o ouvinte SBI está atendendo solicitações. Isso não confirma, por si só, que o registro NRF está atualizado: verifique os logs de inicialização e heartbeat para isso.

Guia de Operações do OmniNWDAF

[Visão Geral](#) | [Configuração](#) | [Métricas](#) | [Solução de Problemas](#)

Este guia descreve como o OmniNWDAF se comporta em tempo de execução e como operá-lo. O OmniNWDAF é a **Função de Análise de Dados de Rede** do núcleo 5G (3GPP TS 23.501 §6.2.18, TS 23.288), expondo os serviços Nnwdaf de TS 29.520.

Índice

- [Cobertura de Análise](#)
- [Visão Geral da Arquitetura](#)
- [Endpoints de Serviço](#)
- [Registro e Heartbeat do NRF](#)
- [Nnwdaf_AnalyticsInfo: Solicitação / Resposta](#)
- [Nnwdaf_EventsSubscription: Inscrever / Notificar](#)
- [Entrega de Notificação e Retentativa](#)
- [Referência de ID de Evento de Análise](#)
- [Resumo do Comportamento Operacional](#)

Cobertura de Análise

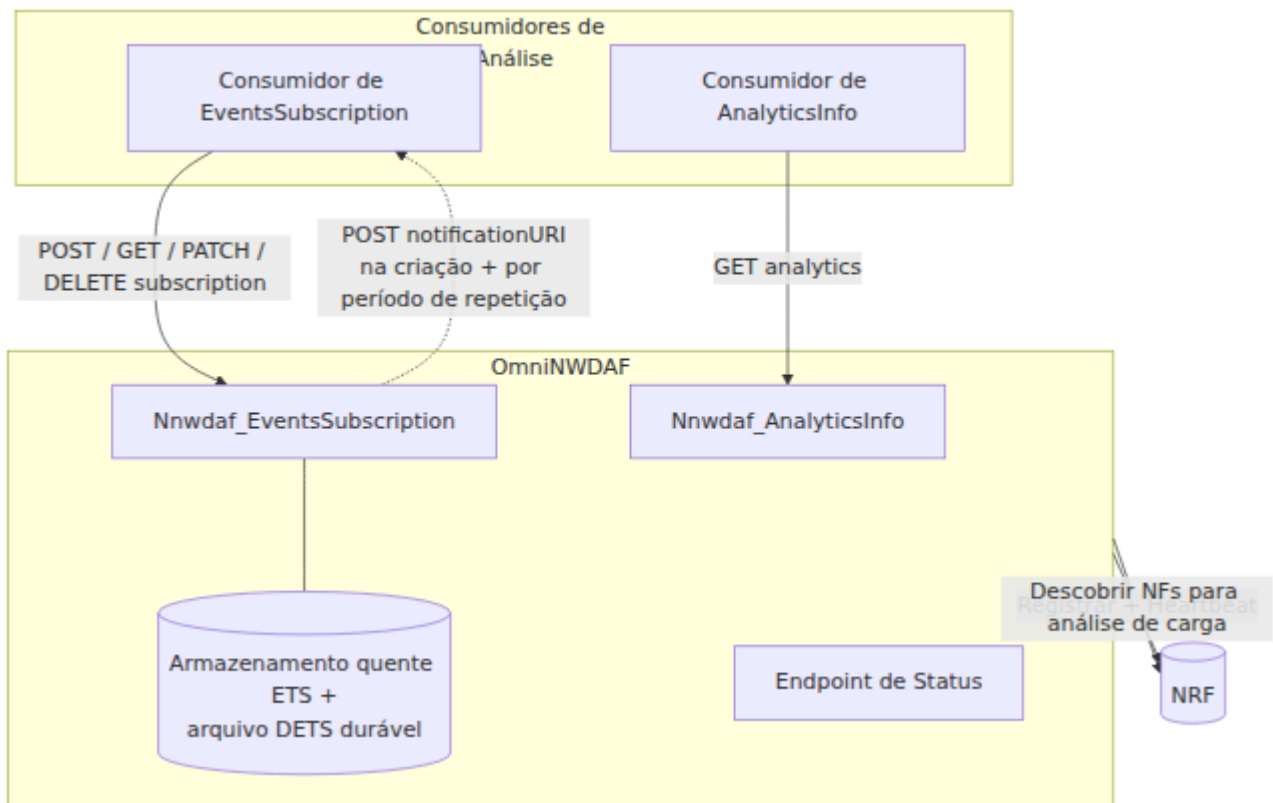
Os eventos orientados a carga são calculados a partir de dados ao vivo que o OmniNWDAF descobre do NRF, nomeadamente o status de registro de cada NF e sua `load` reportada (0-100, TS 29.510 NFProfile):

- `NF_LOAD` - uma entrada `NfLoadLevelInformation` por NF descoberta, carregando sua carga média e de pico reportadas e seu status. Quando um único filtro `nfType` é fornecido, apenas aquele tipo é descoberto.

- `SLICE_LOAD_LEVEL` e `NSI_LOAD_LEVEL` - a carga média reportada entre os NFs que atendem ao S-NSSAI solicitado.
- `NETWORK_PERFORMANCE` - agregados em toda a rede: a proporção de NFs registrados (`GNB_ACTIVE_RATIO`) e a carga média reportada (`GNB_COMPUTING_USAGE`, `GNB_MEMORY_USAGE`).

Para esses eventos, `confidence` reflete quanto dado ao vivo sustentou o resultado (a contagem de NFs com uma `load` utilizável) e é 0 quando nenhum dado estava disponível.

Visão Geral da Arquitetura



O armazenamento de assinaturas mantém uma tabela ETS quente em memória, apoiada por um **arquivo DETS durável**. As assinaturas sobrevivem a uma reinicialização: elas são recarregadas na inicialização e suas notificações recorrentes são rearmadas.

Endpoints de Serviço

O OmniNWDAF expõe os seguintes endpoints HTTP SBI. Todos os caminhos são servidos sob o vínculo SBI configurado por `sbi_scheme`, `sbi_addr` e `sbi_port` (veja [Configuração](#)).

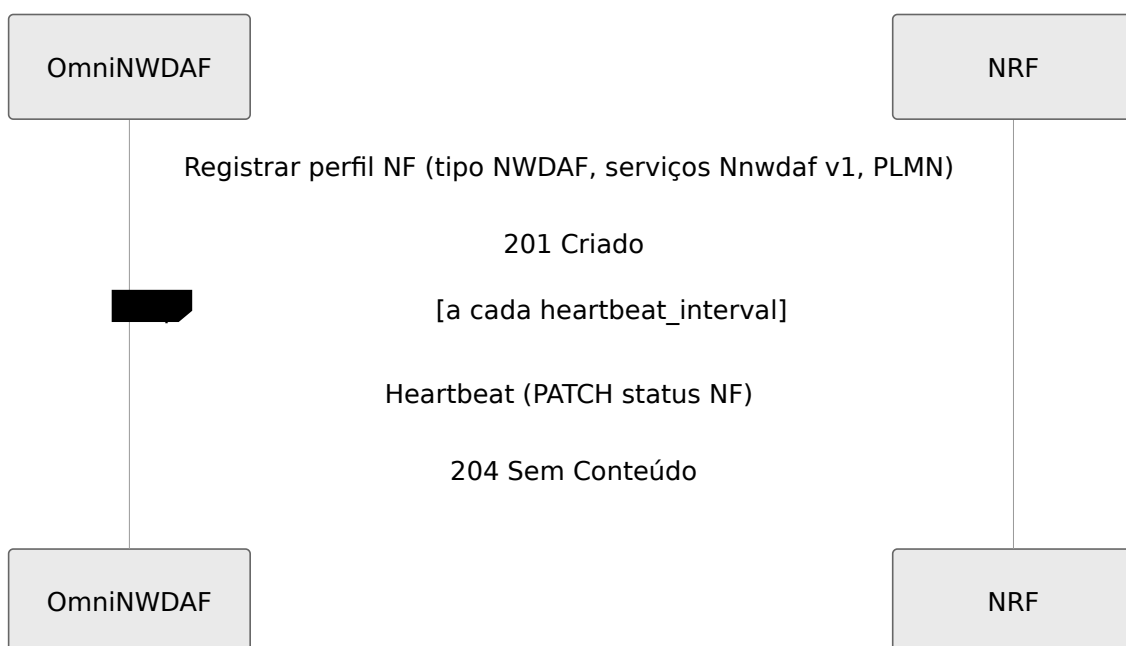
Método	Caminho	
GET	/nnwdaf-analyticsinfo/v1/analytics	Recu para sele ever parâ cons filtre
POST	/nnwdaf-eventssubscription/v1/subscriptions	Cria O co noti ever
GET	/nnwdaf-eventssubscription/v1/subscriptions/{subscriptionId}	Ler u arma
PATCH	/nnwdaf-eventssubscription/v1/subscriptions/{subscriptionId}	Mod assin mes assin arma
DELETE	/nnwdaf-eventssubscription/v1/subscriptions/{subscriptionId}	Dele assin
GET	/nnwdaf-analyticsinfo/v1/status	Verif viva

Esses mapeiam para as operações de serviço Nnwdaf em [TS 29.520](#). Criar, ler, modificar (PATCH) e deletar uma assinatura individual são todos fornecidos.

Registro e Heartbeat do NRF

Na inicialização, o OmniNWDAF se registra no NRF em `nrf_uri` como tipo de NF `NWDAF`, anunciando os dois serviços `Nnwdaf` (`nnwdaf-analyticsinfo` e `nnwdaf-eventsubscription`, ambos `v1`) e o PLMN atendido (`mcc/mnc`). Em seguida, envia um heartbeat periódico a cada `heartbeat_interval` milissegundos para manter o perfil ativo. O registro e o heartbeat do NRF seguem [TS 29.510](#).

Além do registro, o OmniNWDAF também usa o NRF como uma **fonte de dados**: quando uma solicitação ou notificação de análise orientada a carga é atendida, ele descobre as instâncias NF relevantes do NRF e lê sua `load` e status reportados para calcular o resultado (veja [Cobertura de Análise](#)). A descoberta é feita com o melhor esforço: uma falha de descoberta resulta em um conjunto NF vazio e as análises afetadas retornam um resultado `confidence 0` em vez de gerar um erro.



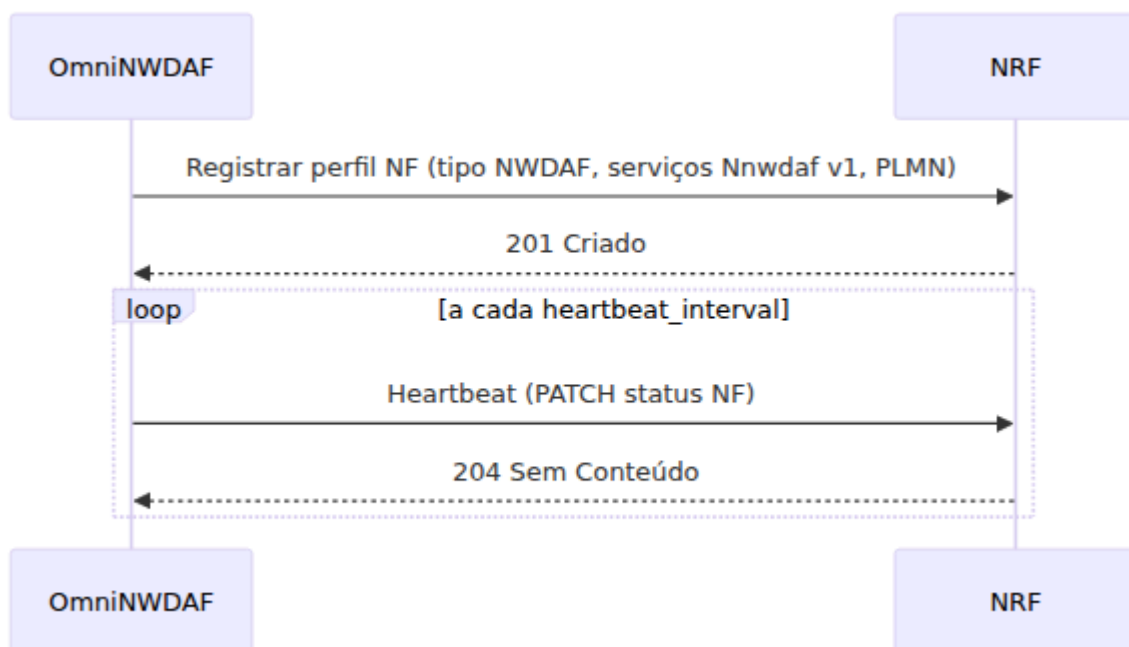
Além do registro e heartbeat, o OmniNWDAF realiza a descoberta do NRF para ler a carga e o status de outros NFs para as análises orientadas a carga. Ele não consome nenhuma outra interface de produtor.

Nnwdaf_AnalyticsInfo: Solicitação / Resposta

Um consumidor solicita análises para um único evento chamando `GET /nnwdaf-analyticsinfo/v1/analytics` com um parâmetro de consulta `event-id`. Parâmetros de consulta adicionais (por exemplo, S-NSSAI, DNN, SUPI, tipo de NF) atuam como filtros.

O OmniNWDAF retorna um objeto `AnalyticsData`:

- `start` e `timeStampGen` são definidos para o horário atual.
- `expiry` é definido para o horário atual mais `analytics_validity_seconds`.
- Para os quatro eventos orientados a carga (`NF_LOAD`, `SLICE_LOAD_LEVEL`, `NSI_LOAD_LEVEL`, `NETWORK_PERFORMANCE`), o OmniNWDAF descobre instâncias NF do NRF e calcula as análises a partir de sua carga e status reportados. `confidence` reflete o número de NFs com dados de carga utilizáveis (0 quando nenhum).
- Para os eventos restantes, os valores de filtro do chamador são ecoados de volta dentro da resposta.



Como funciona: os eventos orientados a carga refletem a população NF ao vivo reportada pelo NRF. Para os eventos restantes, a resposta carrega os timestamps padrão e ecoa os valores de filtro do chamador.

Nnwdaf_EventsSubscription: Inscrever / Notificar

Um consumidor cria uma assinatura fazendo um POST para `/nnwdaf-eventssubscription/v1/subscriptions` com um corpo contendo um `notificationURI` e uma ou mais `eventSubscriptions`.

Comportamento na criação:

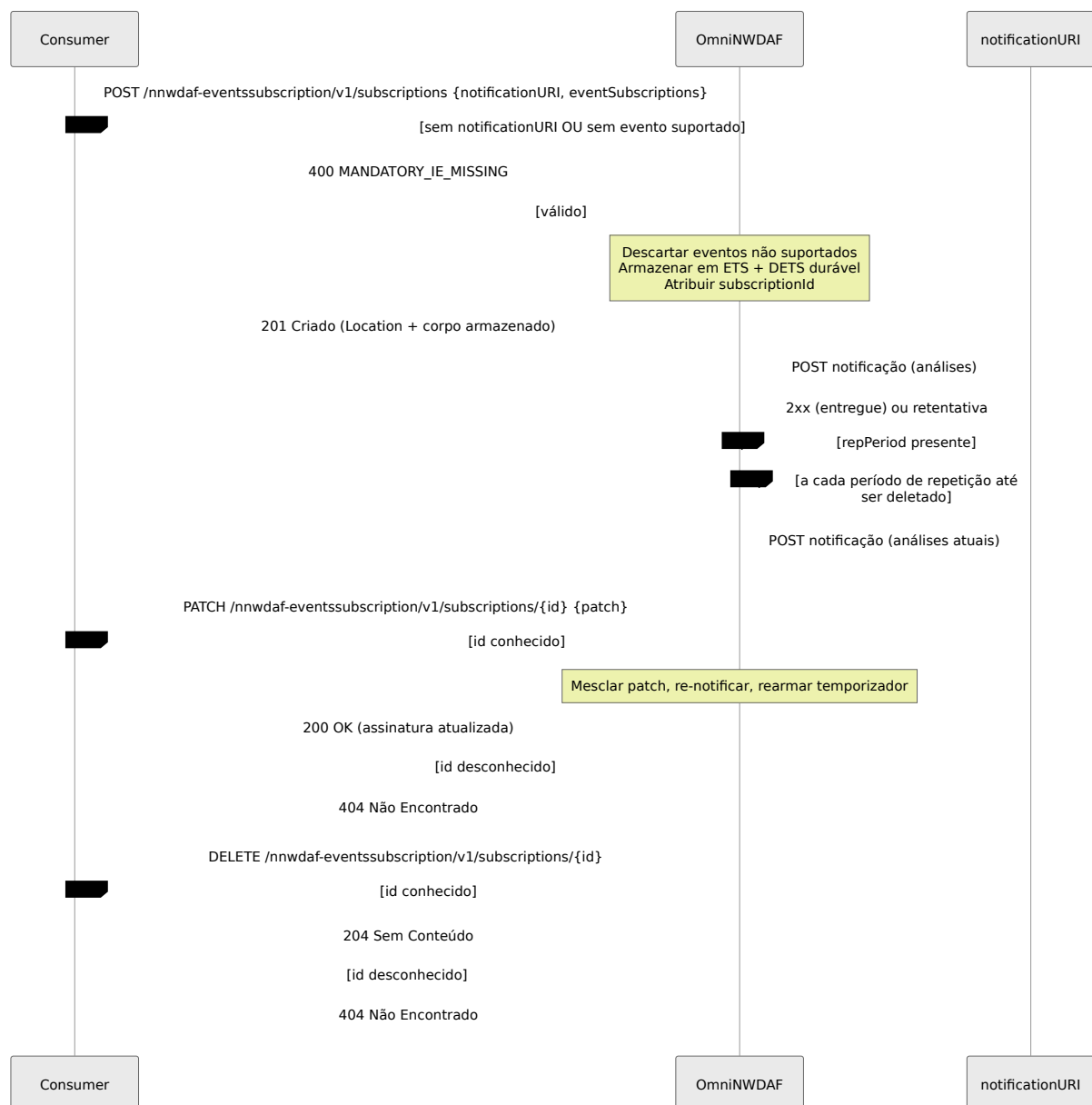
- A assinatura é armazenada na tabela ETS quente e espelhada para o **arquivo DETS durável**, e recebe um id de assinatura.
- Um `201 Criado` é retornado com um cabeçalho `Location` apontando para a nova assinatura e o corpo armazenado na resposta.
- **Eventos não suportados na assinatura são descartados silenciosamente.** Uma assinatura que contém **nenhum evento suportado** é rejeitada com `400 MANDATORY_IE_MISSING`. Um `notificationURI` ausente também é rejeitado com `400 MANDATORY_IE_MISSING`.
- Uma **notificação imediata** é entregue ao `notificationURI` na criação (uma `eventNotification` por evento suportado inscrito).

Notificação recorrente. Quando uma assinatura carrega um requisito de relatório com um `repPeriod` (segundos), seja no nível da assinatura (`evtReq`) ou por evento (`eventReq`), o OmniNWDAF reentrega a notificação nesse período até que a assinatura seja removida. O menor `repPeriod` positivo encontrado é utilizado. Sem um `repPeriod`, a assinatura é única: apenas a notificação do momento da criação é enviada. Não há limite ou gatilho de mudança de dados.

Ler e modificar. Uma assinatura armazenada pode ser lida com `GET .../subscriptions/{id}` e modificada com `PATCH .../subscriptions/{id}`. Uma modificação mescla superficialmente o corpo do patch na assinatura armazenada (preservando o `subscriptionId`), re-notifica imediatamente com a definição atualizada e rearma o temporizador de repetição.

Durabilidade. Assinaturas e suas notificações recorrentes sobrevivem a uma reinicialização: o armazenamento é recarregado do arquivo DETS na

inicialização e os temporizadores de repetição são rearmados.



Entrega de Notificação e Retentativa

Toda notificação (a do momento da criação, cada uma recorrente e a que segue uma modificação) é entregue **com o melhor esforço** com uma retentativa exponencial com limite:

- Até `notification_max_attempts` tentativas de entrega são feitas.

- O atraso antes de cada tentativa começa em `notification_backoff_ms` e cresce exponencialmente por tentativa.
- Se o assinante não aceitou a notificação após a tentativa final, o OmniNWDAF desiste e registra um aviso. A assinatura permanece armazenada; não é removida automaticamente.

Veja [Configuração](#) para os parâmetros de tentativa, e [Métricas](#) para como as tentativas de entrega, tentativas e desistências aparecem nos logs.

Referência de ID de Evento de Análise

O OmniNWDAF anuncia e aceita todos os 16 IDs de evento de análise padronizados da enumeração `NwdafEvent` em [TS 29.520](#).

ID do Evento	Análise	Referência
NF_LOAD	Análise de carga para um NF específico	TS 29.520
SLICE_LOAD_LEVEL	Informações sobre o nível de carga para um slice de rede	TS 29.520
NSI_LOAD_LEVEL	Nível de carga da instância do slice de rede	TS 29.520
NETWORK_PERFORMANCE	Análise de desempenho da rede	TS 29.520
SERVICE_EXPERIENCE	Análise da experiência do serviço	TS 29.520
UE_MOBILITY	Análise de mobilidade do UE	TS 29.520
UE_COMMUNICATION	Análise de comunicação do UE	TS 29.520
QOS_SUSTAINABILITY	Análise de sustentabilidade de QoS	TS 29.520
ABNORMAL_BEHAVIOUR	Análise de comportamento anômalo	TS 29.520
USER_DATA_CONGESTION	Análise de congestionamento de dados do usuário	TS 29.520
SM_CONGESTION	Análise de congestionamento de gerenciamento de sessão	TS 29.520
DISPERSION	Análise de dispersão de dados e mobilidade	TS 29.520
RED_TRANS_EXP	Análise de experiência de transmissão redundante	TS 29.520

ID do Evento	Análise	Referência
WLAN_PERFORMANCE	Análise de desempenho de WLAN	TS 29.520
DN_PERFORMANCE	Análise de desempenho da Rede de Dados	TS 29.520
PDU_SESSION_TRAFFIC	Análise de tráfego da sessão PDU	TS 29.520

Um `event-id` fora desse conjunto retorna `404 UNSUPPORTED_EVENT` em AnalyticsInfo e é descartado silenciosamente quando aparece dentro de uma assinatura de eventos.

Resumo do Comportamento

Operacional

Comportamento	Detalhe
Análises orientadas a carga	<code>NF_LOAD</code> , <code>SLICE_LOAD_LEVEL</code> , <code>NSI_LOAD_LEVEL</code> , <code>NETWORK_PERFORMANCE</code> calculados a partir de dados ao vivo do NRF; <code>confidence</code> = contagem de NFs com carga utilizável
Conteúdo de análises outras	Filtros ecoados de volta
Timestamps de análises	<code>start/timeStampGen</code> = agora; <code>expiry</code> = agora + <code>analytics_validity_seconds</code>
Armazenamento de assinaturas	Tabela ETS quente espelhada para um arquivo DETS durável; recarregada na reinicialização
Operações de assinatura	Criar, ler (GET), modificar (PATCH), deletar
Notificações de assinatura	Notificação imediata na criação; notificação recorrente quando um <code>repPeriod</code> está presente; re-notificar na modificação; sem limite ou gatilho de mudança de dados
Eventos não suportados	Descartados silenciosamente em uma assinatura; <code>404 UNSUPPORTED_EVENT</code> em AnalyticsInfo
Assinatura vazia	Rejeitada com <code>400 MANDATORY_IE_MISSING</code>
Entrega de notificação	Melhor esforço, retentativa exponencial com limite; desistência registrada após a tentativa final
NRF	Registro, heartbeat periódico e descoberta de NF para as análises orientadas a carga

Comportamento	Detalhe
Coleta de dados	A descoberta do NRF (carga + status por NF) alimenta as análises orientadas a carga

Solução de Problemas do OmniNWDAF

[Visão Geral](#) | [Operações](#) | [Configuração](#) | [Métricas](#)

Este guia cobre problemas comuns e suas resoluções.

Problemas Comuns

AnalyticsInfo retorna 400 MANDATORY_QUERY_PARAM_MISSING

Sintomas: GET /nnwdaf-analyticsinfo/v1/analytics retorna 400 com MANDATORY_QUERY_PARAM_MISSING.

Causas possíveis:

- O parâmetro de consulta event-id está ausente. É obrigatório: as análises são sempre recuperadas para um único evento.

Resolução:

- Adicione um parâmetro de consulta event-id definido como um dos IDs de evento suportados (veja [Operações: Referência de ID de Evento de Análise](#)).

AnalyticsInfo retorna 404 UNSUPPORTED_EVENT

Sintomas: GET /nnwdaf-analyticsinfo/v1/analytics retorna 404 com UNSUPPORTED_EVENT.

Causas possíveis:

- O valor de `event-id` não é um dos 16 IDs de evento de análise padronizados.

Resolução:

1. Verifique o valor em relação à [Referência de ID de Evento de Análise](#). Os IDs de evento diferenciam maiúsculas de minúsculas e devem corresponder exatamente aos nomes `NwdafEvent` do [TS 29.520](#).

A criação da assinatura retorna 400 MANDATORY_IE_MISSING

Sintomas: `POST /nnwdaf-eventsubscription/v1/subscriptions` retorna `400` com `MANDATORY_IE_MISSING`.

Causas possíveis:

- O corpo não possui `notificationURI`.
- O `eventSubscriptions` do corpo contém **nenhum evento suportado**. Eventos não suportados são silenciosamente descartados, então um corpo que lista apenas eventos não suportados é tratado como vazio e rejeitado.

Resolução:

1. Certifique-se de que um `notificationURI` esteja presente.
2. Certifique-se de que pelo menos uma entrada de `eventSubscriptions` faça referência a um ID de evento suportado.

O assinante nunca recebe a notificação

Sintomas: Uma assinatura é criada (`201`), mas o `notificationURI` do assinante nunca recebe o callback.

Causas possíveis:

- O `notificationURI` é inacessível a partir do OmniNWDAF, ou o assinante retornou um código não `2xx` em cada tentativa.

- Todas as tentativas de entrega (até `notification_max_attempts`) foram esgotadas; o OmniNWDAF desistiu e registrou um aviso.

Resolução:

1. Confirme que o `notificationURI` é acessível a partir do OmniNWDAF e que o assinante retorna um `2xx` ao receber.
2. Verifique os logs do OmniNWDAF para as tentativas de entrega e o aviso final de desistência (veja [Métricas](#)).
3. Se necessário, aumente `notification_max_attempts` ou `notification_backoff_ms` (veja [Configuração](#)). Uma notificação perdida no momento da criação é reattemptada de acordo com o cronograma de entrega; se a assinatura tiver um `repPeriod`, ela também será reattemptada em cada ciclo recorrente. Caso contrário, modifique (`PATCH`) ou recrie a assinatura para acionar uma nova entrega.

Assinaturas desapareceram após uma reinicialização

Sintomas: Assinaturas criadas anteriormente retornam `404` ao serem lidas ou excluídas, ou não existem mais após uma reinicialização.

Causas possíveis:

- As assinaturas são persistidas em um arquivo DETS durável e recarregadas na inicialização. Se desapareceram, o arquivo de armazenamento não foi preservado: pode ter estado em armazenamento efêmero (o caminho padrão está sob o diretório temporário do sistema) ou o `subscription_store_path` configurado mudou entre execuções.

Resolução:

1. Aponte `subscription_store_path` para um armazenamento durável que persista entre reinicializações e mantenha o caminho estável entre execuções (veja [Configuração](#)).
2. Se o arquivo de armazenamento foi perdido, recrie as assinaturas afetadas.

OmniNWDAF não aparece no NRF

Sintomas: Consumidores não conseguem encontrar o OmniNWDAF, ou o NRF não mostra nenhum perfil NWDAF.

Causas possíveis:

- `nrf_uri` aponta para o NRF errado ou um endereço inacessível.
- O registro falhou na inicialização (verifique os logs).
- O heartbeat expirou e o NRF expirou o perfil.

Resolução:

1. Verifique se `nrf_uri` está correto e acessível (veja [Configuração](#)).
2. Verifique os logs de inicialização para um registro bem-sucedido.
3. Confirme se o heartbeat está em execução; se `heartbeat_interval` for maior do que o período de validade do NRF, encurte-o.

DELETE retorna 404

Sintomas: `DELETE /nnwdaf-eventssubscription/v1/subscriptions/{id}` retorna `404`.

Causas possíveis:

- O ID da assinatura é desconhecido, seja porque nunca existiu, foi excluído ou o arquivo de armazenamento durável não foi preservado entre uma reinicialização.

Resolução:

1. Confirme se o ID corresponde a um retornado no cabeçalho `Location` de uma criação. Se existiu antes de uma reinicialização, verifique se `subscription_store_path` aponta para um armazenamento durável (veja [Assinaturas desapareceram após uma reinicialização](#)).

