

# **Documento de Especificação Técnica - API APEX JSON**

## **Parâmetros Técnicos**

Versão 3.0

Data: 2023-02-10

## » Controlo do Documento

### HISTORIAL DAS REVISÕES

Versão	Data	Descrição das alterações
2.1	2022-09-26	Atualização para formato 2.1. Adicionado campo LastStopDenyEntry na configuração dos operadores. Adicionado suporte a configuração de permissões de acesso a funcionalidades extra da API APEX.
3.0	2023-02-10	Alteração de versão de formato de 2.1 para 3.0.

## » Índice

<b>1</b>	<b>Introdução.....</b>	<b>6</b>
1.1	Enquadramento do documento .....	6
1.2	Referências a outros documentos.....	6
1.3	Glossário .....	6
1.4	Simbologia .....	7
<b>2</b>	<b>Descrição Geral.....</b>	<b>8</b>
2.1	Observações e Codificação .....	8
2.1.1	Nomenclatura.....	8
2.1.2	Cardinalidade .....	9
2.1.3	Codificação do ficheiro .....	9
2.2	Tipo de Dados .....	10
<b>3</b>	<b>Estrutura do JSON.....</b>	<b>11</b>
3.1	Versionamento .....	11
3.2	Objetos .....	12
3.2.1	Document.....	12
3.2.2	General .....	13
3.2.3	CardType .....	13
3.2.4	SamType .....	15
3.2.5	Network.....	16
3.2.6	Operator .....	17
3.2.7	Calendar .....	19
3.2.8	InfractionAttribute .....	21
3.2.9	Infraction .....	21
3.2.10	FineAttribute .....	22
3.2.11	Fine .....	23
3.2.12	AccessPermission .....	23
3.3	Ficheiro de Exemplo .....	24

---

## » Índice de Figuras

---

Figura 1 – Estrutura do documento JSON .....	11
--	----

---

## » Índice de tabelas

---

Tabela 1 – Glossário (definições, termos e acrónimos) .....	6
Tabela 2 – Simbologia de notas do documento .....	7

# 1 Introdução

Este documento apresenta a especificação do Ficheiro de configuração de parâmetros técnicos no formato JSON, que será utilizado pelo Gestor de Configurações.

## 1.1 Enquadramento do documento

Pretende-se com este documento descrever um ficheiro em formato JSON correspondente à configuração de parâmetros técnicos utilizada pela API APEX. O ficheiro JSON é utilizado pelo sistema Gestor de Configurações, onde é processado e convertido no formato utilizado pela API APEX, sendo neste documento identificada a estrutura JSON e os seus respetivos elementos.

A estrutura e conteúdo apresentados neste documento têm o objetivo de permitir a representação de todos os campos e configurações de parâmetros técnicos necessárias para a API APEX e identificados no documento (3).

## 1.2 Referências a outros documentos

Documentos de referência com relevância para este documento:

- (1) OTLIS, API-VIVA External Specifications, v1.54, 28 janeiro 2021.
- (2) OTLIS, Viva Data Model Functional Specification, v1.3, 29 dezembro 2017.
- (3) Link Consulting, Documento de Especificação Técnica - API APEX Ficheiro de Configuração Parâmetros Técnicos, v3.0, 10 fevereiro 2023.
- (4) Link Consulting, Documento de Especificação Técnica - API APEX Segurança e Algoritmos, v1.1, 26 setembro 2022.

## 1.3 Glossário

Esta secção fornece um glossário detalhado ordenado alfabeticamente do domínio de negócio, definindo todos e cada um dos conceitos de negócio relevantes para o projeto.




A tabela abaixo identifica as definições, termos e acrónimos utilizados ao longo deste documento.

Item	Descrição

Tabela 1 – Glossário (definições, termos e acrónimos)

## 1.4 Simbologia

No presente documento foi adotada a seguinte simbologia:

Item	Descrição
	Representa uma informação importante.
	Representa uma dica ou boa prática.
	Representa uma chamada de atenção para um (eventual) problema ou questão existente.

**Tabela 2 – Simbologia de notas do documento**

## 2 Descrição Geral

Esta secção apresenta a sintaxe e os tipos de dados utilizados no ficheiro JSON da configuração de topologia de rede.

### 2.1 Observações e Codificação

#### 2.1.1 Nomenclatura

Formato	Descrição
<b>Int8</b>	Valor representado em decimal inteiro de tamanho de 8 bits. Aceita valores entre -127 a 127.
<b>UInt8</b>	Valor representado em decimal inteiro positivo de tamanho de 8 bits. Aceita valores entre 0 a 255.
<b>Int16</b>	Valor representado em decimal inteiro de tamanho de 16 bits. Aceita valores entre -32767 a 32767.
<b>UInt16</b>	Valor representado em decimal inteiro positivo de tamanho de 16 bits. Aceita valores entre 0 a 65535.
<b>Int32</b>	Valor representado em decimal inteiro de tamanho de 32 bits. Aceita valores entre -2147483647 a 2147483647.
<b>UInt32</b>	Valor representado em decimal inteiro positivo de tamanho de 32 bits. Aceita valores entre 0 a 4294967295.
<b>Enum</b>	Valor de um enumerado representado em decimal inteiro.
<b>String</b>	Valor representado em texto de tamanho variável.
<b>String(XX)</b>	Valor representado em texto, com um <u>tamanho máximo</u> de XX caracteres.
<b>StringHex</b>	Representação de um valor através de uma string hexadecimal, com tamanho variável, mas sempre de tamanho par (padding zero à esquerda).
<b>StringHex(XX)</b>	Representação de um valor através de uma string hexadecimal, com <u>tamanho fixo</u> de XX caracteres (padding zero à esquerda).
<b>StringB64</b>	Representação em string Base64 de um campo binário (exemplo: array), com tamanho variável.
<b>DateTime</b>	Representação de uma data e hora através de texto, seguindo o formato: aaaa-mm-ddThh:mm:ss  Data: aaaa – Valor do ano composto por 4 dígitos mm – Valor do mês composto por 2 dígitos dd – Valor do dia composto por 2 dígitos  Hora: hh – Valor do hora composto por 2 dígitos e em representação 24H. mm – Valor dos minutos composto por 2 dígitos ss – Valor dos segundos composto por 2 dígitos
<b>DateTimeMask</b>	Representação de uma máscara de data e hora através de texto, seguindo o formato:



	<p>yyyy-MM-ddThh:mm:ss</p> <p>Exemplos:          yyyy-01-ddThh:mm:ss – Representa todos os dias, anos e horas do mês de janeiro.          yyyy-MM-ddT10:30:00 – Representa a hora 10:30 de todos os dias do ano.</p>
<b>Date</b>	<p>Representação de uma data através de texto, seguindo o formato:          aaaa-mm-dd</p> <p>aaaa – Valor do ano composto por 4 dígitos          mm – Valor do mês composto por 2 dígitos          dd – Valor do dia composto por 2 dígitos</p>
<b>DateMask</b>	<p>Representação de uma máscara de data através de texto, seguindo o formato:          yyyy-MM-dd</p> <p>Exemplos:          yyyy-01-dd – Representa todos os dias, anos de janeiro.</p>
<b>Time</b>	<p>Representação de uma hora através de texto, seguindo o formato:          hh:mm:ss</p> <p>hh – Valor do hora composto por 2 dígitos e em representação 24H.          mm – Valor dos minutos composto por 2 dígitos          ss – Valor dos segundos composto por 2 dígitos</p>
<b>VersionLibrary</b>	<p>Representação de uma versão de uma biblioteca, definida através das componentes Major, Minor e Revision.</p> <p>Formato: MMM.mmm.rrr (major.minor.revision)</p> <p>O valor não tem de ter obrigatoriamente 3 dígitos em cada componente, mas as 3 componentes devem estar presentes.</p> <p>Exemplo 1: “2.1.3”          Exemplo 2: “21.0.256”</p>

## 2.1.2 Cardinalidade

A cardinalidade é uma propriedade que descreve o número de ocorrências do elemento JSON, bem como se o elemento é ou não necessário:

- 0,1 - Opcional - Nenhuma, ou um máximo de 1 ocorrência permitida.
- 0,N - Opcional - Nenhuma, ou um máximo de 1 ou mais ocorrências permitidas.
- 1,1 - Obrigatório - Pelo menos uma ocorrência é exigida, mas não mais do que uma.
- 1,N - Obrigatório - É necessária pelo menos uma ou mais ocorrências.

## 2.1.3 Codificação do ficheiro

Todos os ficheiros devem ser codificados em UTF-8.

## 2.2 Tipo de Dados

### ENUMERADOS API APEX

Os enumerados Apex**XXX** (exemplo: *ApexInterchangeMode*) estão definidos no documento (1).

### ENUMERADOS API VIVA

Os enumerados T\_**XXX** (exemplo: *T\_CardDataModel*) estão definidos no documento (1).

## 3 Estrutura do JSON

Esta secção apresenta a estrutura do ficheiro JSON para a configuração de parâmetros técnicos, seguindo a seguinte estrutura:

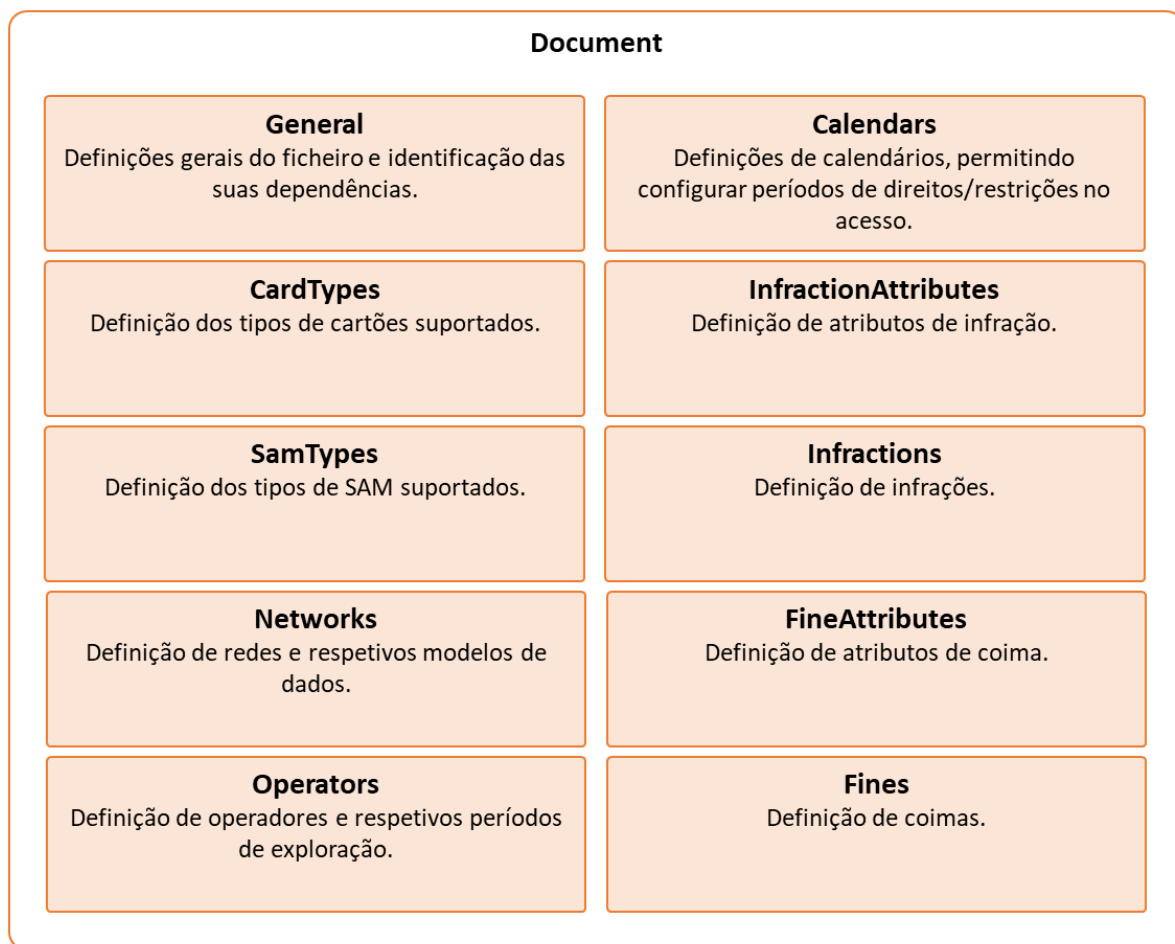


Figura 1 – Estrutura do documento JSON

### 3.1 Versionamento

A versão do ficheiro (General.FileVersion) deve ser incremental e representada por um valor inteiro. Não deve em circunstância alguma existir dois ficheiros com conteúdo diferente com versões iguais.



De acordo com as configurações embedded, o valor máximo suportado será de 999.

## 3.2 Objetos

Esta seção descreve todos os objetos JSON utilizados no documento. Após cada definição de objeto JSON é apresentado um breve exemplo ilustrativo do formato do mesmo. Os seus valores são apresentados no formato correto, no entanto, não devem ser considerados como valores reais.

### 3.2.1 Document



Campo	Tipo	Ocorr.	Descrição
<b>formatVersion</b>	UInt32	1,1	Versão do formato do documento. Valor fixo: 2
<b>securityType</b>	UInt16	1,1	Tipo de segurança a ser aplicado ao ficheiro embedded.  Valores possíveis: 1 – MD5
<b>general</b>	General	1,1	Definições gerais do ficheiro e identificação das suas dependências.
<b>cardTypes</b>	List<CardType>	1,N	Lista de cartões suportados.
<b>samTypes</b>	List<SamType>	1,N	Lista de SAM suportados.
<b>networks</b>	List<Network>	1,N	Lista de redes suportadas, identificado a versão do modelo de dados utilizada.
<b>operators</b>	List<Operator>	1,N	Lista de operadores, com a configuração dos seus períodos de exploração.
<b>calendars</b>	List<Calendar>	0,N	Lista de definições de calendário, onde se configuram os períodos de direitos/restrições no acesso.
<b>infractionAttributes</b>	List<InfractionAttribute>	0,N	Lista de atributos de infração.
<b>infractions</b>	List<Infraction>	0,N	Lista de infrações.
<b>fineAttributes</b>	List<FineAttribute>	0,N	Lista de atributos de coima.
<b>fines</b>	List<Fine>	0,N	Lista de coimas.
<b>accessPermissions</b>	List<AccessPermission>	0,N	Lista de permissões de acesso.

#### EXEMPLO

```
{
  "formatVersion": 2,
  "securityType": 1,
  "general": {<object-content-here>},
  "cardTypes": [<card-types-here>],
  "samTypes": [<sam-types-here>],
  "networks": [<networks-here>],
  "operators": [<operators-here>],
  "calendars": [<calendars-here>],
  "infractionAttributes": [<infraction-attributes-here>],
  "infractions": [<infractions-here>],
  "fineAttributes": [<fine-attributes-here>],
}
```

```
"fines": [<fines-here>],
"accessPermissions": [<access-permissions-here>]
}
```

### 3.2.2 General




Campo	Tipo	Ocorr.	Descrição
<b>fileVersion</b>	UInt16	1,1	Versão do ficheiro de configuração de Parâmetros Técnicos. Valor sequencial e número inteiro.   Este valor é mapeado na versão <i>major</i> do ficheiro <i>embedded</i> , sendo a versão <i>minor</i> gerida internamente pelo Gestor de Configurações.
<b>fileDate</b>	DateTime	1,1	Data de geração do ficheiro.
<b>fileStartDate</b>	DateTime	1,1	Data de início de validade do ficheiro de configuração.
<b>fileEndDate</b>	DateTime	0,1	Data de fim de validade do ficheiro de configuração.   Se omissa, significa que não tem data final de validade.
<b>apexMinVersion</b>	VersionLibrary	1,1	Versão mínima necessária da API APEX.
<b>backendURL</b>	String	1,1	URL do back-end (exemplo: verificação on-line de vouchers).

#### EXEMPLO

```
{
  "fileVersion": 1,
  "fileDate": "2021-04-08T09:30:00",
  "fileStartDate": "2021-04-08T09:30:00",
  "apexMinVersion": "1.0.0",
  "backendURL": "https://backend-tml.pt"
}
```

### 3.2.3 CardType

Campo	Tipo	Ocorr.	Descrição
<b>cardTypeID</b>	String(36)	1,1	Identificador único do tipo de cartão.
<b>cardTypeName</b>	String(20)	0,1	Descrição da configuração do cartão.

<b>protocol</b>	Enum	1,1	Identifica o protocolo de comunicação utilizado pelo cartão. Enumerado: T_CardFamily
<b>physicalType</b>	Enum	1,1	Identifica o tipo físico do cartão. Enumerado: T_CardPhysicalType
<b>networkID</b>	String(36)	1,1	Identificador da rede de transportes ( <i>NetworkId</i> ) ao qual pertence o cartão.
<b>filterType</b>	Enum	1,1	Identifica o tipo de filtro em utilização. Enumerado: ApexCardFilterType
<b>filter</b>	StringHex	0,1	Filtro a ser aplicado sobre os dados do CardInfo devolvido na deteção do cartão pela API VIVA. Utilização do carácter 'X' para ser aceite qualquer valor.  Exemplo: "XX0F836201"   O campo pode ser obrigatório dependendo do valor do campo 'filterType'.
<b>aid</b>	StringHex	1,1	Identificador da aplicação.   Apenas utilizado em cartões personalizáveis (exemplo: Lisboa Viva).
<b>cardDataModel</b>	Enum	1,1	Identifica o modelo de dados utilizado pelo cartão. Enumerado: T_CardDataModel
<b>kvc</b>	UInt8	0,1	Identifica o KVC do cartão de modo a permitir saber, para tipos de cartões iguais, e com o mesmo AID, qual o modelo de dados presente (e.g. distinguir um cartão de Lisboa de um cartão do Porto).   A configuração com valor 0 é diferente de ausência de configuração, pois um implica valor 0 e outro implica 'não definido', sendo aceite qualquer valor.
<b>pin</b>	StringHex(8)	0,1	PIN de acesso ao cartão por omissão. Exemplo: "30303030"   Apenas utilizado em cartões personalizáveis (exemplo: Lisboa Viva).
<b>securityLevel</b>	Enum	1,1	Identifica o nível de segurança a ser utilizado numa escrita no cartão independentemente do valor lido, permitindo assim um upgrade.  Enumerado T_VivaSecurityLevel.   Apenas utilizado em cartões personalizáveis e MDv2 (exemplo: VivaCard).

<b>operations</b>	List<Enum>	1,N	<p>Lista das operações permitidas sobre o suporte.</p> <p>Enumerado: ApexOperationType</p> <p>Cada elemento da lista é representado pelo valor do enumerado.</p>
-------------------	------------	-----	--

#### EXEMPLO

```
{
  "cardTypeID": "id-cardtype-lisboaviva",
  "cardTypeName": "LisboaViva",
  "protocol": 2,
  "physicalType": 3,
  "networkID": "id-network-lisboa-v1",
  "filterType": 2,
  "filter": "",
  "aid": "315449432E494341",
  "cardDataModel": 1,
  "kvc": 66,
  "pin": "30303030",
  "securityLevel": 0,
  "operations": [ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
}
```

### 3.2.4 SamType

Campo	Tipo	Ocorr.	Descrição
<b>samTypeID</b>	String(36)	1,1	Identificador único do tipo de SAM.
<b>samTypeName</b>	String(20)	0,1	Nome do SAM.
<b>samType</b>	Enum	1,1	Identifica o tipo de SAM. Enumerado: T_SamType
<b>cards</b>	List<String(36)>	1,N	<p>Lista de cartões suportados por este SAM.</p> <p>Cada elemento da lista deve ter o valor CardTypeID.</p>
<b>signatures</b>	List<Signature>	1,N	Lista de tipos de assinaturas suportadas por este SAM.
<b>heartbeat</b>	UInt32	1,1	Múltiplo de referência a partir do qual é realizada uma transação de heartbeat. O valor aqui definido será utilizado contra o valor do contador de assinatura presente no MAC.

#### EXEMPLO

```
{
  "samTypeID": "id-sam-lisboa-valid",
  "samTypeName": "Lisboa Valid",

```

```
"samType": 3,
"cards": [
  "id-cardtype-lisboaviva",
  "id-cardtype-vivacard-v1",
  "id-cardtype-vivacard-v2",
  "id-cardtype-vivaviagem-cts",
  "id-cardtype-vivaviagem-srt"
],
"signatures": [<signatures-here>],
"heartbeat": 0
}
```

### 3.2.4.1 Signature

Campo	Tipo	Ocorr.	Descrição
<b>kif</b>	UInt8	1,1	Chave do SAM ( <i>Key Identifier</i> )
<b>kvc</b>	UInt8	1,1	Versão da chave do SAM ( <i>Key Version Control</i> )
<b>operationTypes</b>	List<Enum>	1,N	<p>Lista com os tipos de operação suportados pelo SAM.</p> <p>Enumerado: ApexOperationType</p> <p>Cada elemento da lista é representado pelo valor do enumerado.</p>

#### EXEMPLO

```
{
  "kif": 0,
  "kvc": 0,
  "operationTypes": [11, 13, 15, 16]
}
```

### 3.2.5 Network

Campo	Tipo	Ocorr.	Descrição
<b>networkID</b>	String(36)	1,1	Identificador único da rede.
<b>networkCodes</b>	List<NetworkCode>	1,N	Lista de configurações de códigos de rede, associados a modelo de dados.

#### EXEMPLO

```
{
  "networkID": "id-network-lisboa-v1",
  "networkCodes": [
    {
      "cardDataModel": 1,
      "networkCode": 2
    }
  ]
}
```



```

    },
    {
      "cardDataModel": 2,
      "networkCode": 2
    }
  ]
}

```

### 3.2.5.1 NetworkCode

Campo	Tipo	Ocorr.	Descrição
<b>cardDataModel</b>	Enum	1,1	Identifica o modelo de dados utilizado pelo suporte.  Enumerado T_CardDataModel
<b>networkCode</b>	UInt16	1,1	Representa o código da network de acordo com o modelo de dado, sendo este escrito no cartão.


#### EXEMPLO



```

{
  "cardDataModel": 1,
  "networkCode": 2
}

```

### 3.2.6 Operator


Campo	Tipo	Ocorr.	Descrição
<b>operatorLongID</b>	String(36)	1,1	Identificador único do operador.
<b>networkID</b>	String(36)	1,1	Identificador da rede a que pertence.
<b>operatorShortID</b>	UInt16	1,1	Identificador do operador a ser escrito no cartão.
<b>operatorShortName</b>	String(10)	0,1	Nome curto do operador até 10 caracteres (informativo).
<b>operatorLongName</b>	String(50)	0,1	Nome do operador (informativo).
<b>nextStopcheck</b>	Boolean	0,1	Obriga que seja verificada se a próxima paragem é válida na validação.   Se omissa, é equivalente ao valor "false".

<b>lastStopDenyEntry</b>	Boolean	0,1	Indica se o operador rejeita validações de entrada na última paragem do percurso.   Se omissa, é equivalente ao valor "false".   Suportado no formato v3.0 e APEX v3.0.0
<b>explorationPeriods</b>	List<ExplorationPeriod>	0,N	Lista dos períodos de exploração do operador.

#### EXEMPLO

```
{
  "operatorLongID": "id-operator-ca",
  "networkID": "id-network-lisboa-v1",
  "operatorShortID": 1,
  "operatorShortName": "CA",
  "operatorLongName": "Carris",
  "nextStopCheck": false,
  "explorationPeriods": [<exploration-periods-here>]
}
```

### 3.2.6.1 ExplorationPeriod

Campo	Tipo	Ocorr.	Descrição
<b>startDate</b>	DateTime	1,1	Data e hora de início da validade da configuração.
<b>endDate</b>	DateTime	0,1	Data e hora de fim da validade da configuração.   Se omissa significa que não tem data final de validade.
<b>explorationStartTime</b>	Time	1,1	Hora de início de exploração.   Granularidade em minutos.
<b>explorationEndTime</b>	Time	1,1	Hora de fim de exploração.   Granularidade em minutos.
<b>Tolerance</b>	UInt16	1,1	Tempo após a hora de fim de exploração em que os títulos ainda são válidos.
<b>antipassbackTime</b>	UInt16	1,1	Valor em minutos, a ser aplicado tanto nas saídas como entradas.

#### EXEMPLO

```
{
  "startDate": "2021-01-01T00:00:00",
}
```

```
{
  "explorationStartTime": "04:00:00",
  "explorationEndTime": "03:00:00",
  "tolerance": 15,
  "antipassbackTime": 1
}
```



### 3.2.7 Calendar

Campo	Tipo	Ocorr.	Descrição
<b>calendarID</b>	String(36)	1,1	Identificador único do calendário.
<b>calendarShortID</b>	Enum	1,1	Identificador do calendário a ser escrito no cartão.  Enumerado: T_VivaContractRestrictTime
<b>calendarConfigs</b>	List<CalendarConfig>	0,N	Lista de configurações do calendário, definindo os seus períodos de validade.

#### EXEMPLO

```
{
  "calendarID": "id-calendar-weekend",
  "calendarShortID": 3,
  "calendarConfigs": [<calendar-configs-here>]
}
```



#### 3.2.7.1 CalendarConfig

Campo	Tipo	Ocorr.	Descrição
<b>startDate</b>	DateTime	1,1	Data e hora de início de validade da configuração.
<b>endDate</b>	DateTime	0,1	Data e hora de fim de validade da configuração.   Se omissa, significa que não tem data final de validade.
<b>operators</b>	List<String(36)>	0,N	Lista de operadores com produtos que envolvem o calendário em causa.   Se omissa ou lista vazia, significa que o calendário é válido para todos os operadores.
<b>validityPeriods</b>	List<ValidityPeriod>	1,N	Definição do período de validade.  Um título de transporte é válido caso seja verificado pelo menos um dos períodos.

#### EXEMPLO

```
{
  "startDate": "2021-01-01T00:00:00",
  "endDate": "2999-01-01T00:00:00",
  "operators": [
    "id-operator-cp",
    "id-operator-ml",
    "id-operator-ca",
    "id-operator-ft"
  ],
  "validityPeriods": [<validity-periods-here>]
}
```

#### 3.2.7.1.1 ValidityPeriod

Campo	Tipo	Ocorr.	Descrição
<b>validityPeriodID</b>	String(36)	1,1	Identificador único do período de validade.
<b>periodStartDateMask</b>	DateMask	1,1	Máscara de data inicial.
<b>periodEndDateMask</b>	DateMask	1,1	Máscara de data final.
<b>periodStartTime</b>	Time	0,1	Hora inicial do período.   Se omissa, significa que não é efetuada nenhuma verificação sobre o início do período horário.
<b>periodEndTime</b>	Time	0,1	Hora final do período.   Se omissa, significa que não é efetuada nenhuma verificação sobre o fim do período horário.
<b>weekDayMask</b>	StringHex(2)	1,1	Máscara dos dias da semana em que o período se aplica, em hexadecimal.  Segue a especificação do enumerado T_UtilWeekDay, sendo o K_SUNDAY o bit mais significativo. O bit menos significativo indica se é válido ou não em feriados. 'FF' representa todos os dias.  Representação lógica: 'DSTQQSSF'

#### EXEMPLO

```
{
  "validityPeriodID": "bda65e26-c41a-4481-9aec-575758437a73",
  "periodStartDateMask": "yyyy-MM-dd",
  "periodEndDateMask": "yyyy-MM-dd",
  "periodStartTime": "00:00:00",
  "periodEndTime": "23:59:59",
  "weekDayMask": "82"
}
```

```
}
```



### 3.2.8 InfractionAttribute


Campo	Tipo	Ocorr.	Descrição
<b>infractionAttributeLongID</b>	String(36)	1,1	Identificador único do atributo.
<b>description</b>	String(255)	1,1	Descrição do atributo (ex: 1ª vez, reincidente)

#### EXEMPLO

```
{
  "infractionAttributeLongID": "id-infraction-attribute-first-time",
  "description": "Primeira vez"
}
```

### 3.2.9 Infraction

Campo	Tipo	Ocorr.	Descrição
<b>infractionLongID</b>	String(36)	1,1	Identificador único da infração.
<b>description</b>	String(255)	1,1	Descrição da infração.
<b>infractionAttributes</b>	List<String(36)>	0,N	<p>Lista de atributos da infração.</p> <p> Se omissa ou lista vazia, significa que a infração é independente dos atributos; caso contrário, apresenta esta lista para o fiscal escolher o atributo a ser utilizado.</p>
<b>controlContractStatus</b>	List<Enum>	0,N	<p>Lista de estados da fiscalização do contrato aos quais a infração se aplica.</p> <p>Enumerado ApexControlContractStatus</p> <p> Se omissa ou lista vazia, significa que a infração é independente do estado da fiscalização; caso contrário, será utilizado pela APEX para filtrar as infrações aplicáveis de acordo com o estado da fiscalização do contrato.</p>

<b>controlEnvironmentStatus</b>	List<Enum>	0,N	<p>Lista de estados da fiscalização do <i>Environment</i> aos quais a infração se aplica.</p> <p>Enumerado ApexControlEnvironmentStatus</p> <p> Se omissa ou lista vazia, significa que a infração é independente do estado da fiscalização; caso contrário, será utilizado pela APEX para filtrar as infrações aplicáveis de acordo com o estado da fiscalização do contrato.</p>
<b>infractionProcedure</b>	String(255)	0,1	Descrição do procedimento a ser realizado pelo fiscal.

#### EXEMPLO

```
{
  "infractionLongID": "id-infraction_without-contract",
  "description": "Sem título válido",
  "infractionAttributes": [
    "id-infraction-attribute-first-time",
    "id-infraction-attribute-recidivist"
  ],
  "controlContractStatus": [
    1,
    2,
    4,
    5,
    6
  ],
  "infractionProcedure": "Emitir coima."
}
```



### 3.2.10 FineAttribute

Campo	Tipo	Ocorr.	Descrição
<b>fineAttributeLongID</b>	String(36)	1,1	Identificador único do atributo.
<b>description</b>	String(255)	1,1	Descrição do atributo (ex: comboios > 50km, outros casos)

#### EXEMPLO

```
{
  "fineAttributeLongID": "id-fine-attribute-train-50km",
  "description": "Comboios >50KM"
}
```

### 3.2.11 Fine

Campo	Tipo	Ocorr.	Descrição
<b>fineLongID</b>	String(36)	1,1	Identificador único da coima.
<b>description</b>	String(255)	1,1	Descrição da coima.
<b>fineAttributes</b>	List<String(36)>	0,N	<p>Lista de atributos da coima.</p> <p> Se omissa ou lista vazia, significa que a coima é independente dos atributos; caso contrário, apresenta esta lista para o fiscal escolher o atributo a ser utilizado.</p>
<b>infractions</b>	List<String(36)>	1,N	<p>Lista de infrações às quais esta coima é aplicável.</p> <p> Obrigatório ter pelo menos 1 elemento.</p>
<b>minAmount</b>	Int32	1,1	Valor mínimo da coima (em cêntimos).
<b>maxAmount</b>	Int32	1,1	Valor máximo da coima (em cêntimos).
<b>promptAmount</b>	Int32	1,1	Valor da coima para pagamento imediato (em cêntimos).

#### EXEMPLO

```
{
  "fineLongID": "id-fine-long-distance",
  "description": "Coima longa distância",
  "fineAttributes": [
    "id-fine-attribute-train"
  ],
  "infractions": [
    "id-infraction_without-contract",
    "id-infraction_card-expired"
  ],
  "minAmount": 6000,
  "maxAmount": 15000,
  "promptAmount": 3000
},
```

### 3.2.12 AccessPermission

Campo	Tipo	Ocorr.	Descrição
<b>accessPermissionLongID</b>	String(36)	1,1	Identificador único da permissão de acesso.
<b>accessKeyType</b>	Enum	1,1	<p>Permite identificar o tipo de chave e algoritmo a ser aplicado.</p> <p>Enumerado ApexAccessKeyType</p>



<b>accessKeyID</b>	UInt16	1,1	Identificador do IV utilizado no algoritmo de segurança.
<b>accessKeyVersion</b>	UInt16	1,1	Versão do IV utilizado no algoritmo de segurança.
<b>accessKeyHash</b>	StringHex(64)	1,1	Hash da chave.
<b>operatorLongID</b>	String(36)	1,1	Operador ao qual a chave se destina.
<b>permissions</b>	List<Enum>	0,N	<p>Lista de permissões atribuídas. Permissões não presentes na lista são consideradas como não atribuídas.</p> <p>Enumerado: ApexAccessPermissionType</p> <p>Cada elemento da lista é representado pelo valor do enumerado.</p>

#### EXEMPLO

```
{
  "accessPermissionLongID": "d6f07aa3-b41f-4fd4-ba92-f28e32cc8ff4",
  "accessKeyType": 1,
  "accessKeyID": 102,
  "accessKeyVersion": 1,
  "accessKeyHash": "5b11a257b8d9451fc05b66fe94ff2ffa6addf7146275653e9cb9d23ab767a86a",
  "operatorLongID": "id-operator-cp",
  "permissions": [
    1
  ]
}
```

## 3.3 Ficheiro de Exemplo

Nesta secção encontram-se anexados dois ficheiros referentes à configuração de parâmetros técnicos, um no formato JSON e o seu correspondente em formato embedded.

Formato JSON	Formato Embedded
 apx_tech.json	 apx_tech.cfg