



TransinfoService

Définition de l'API « TimeTables »

Référence	TransinfoService API « TimeTables »
Date	02/12/2013
Version	1.1
Pages	12

Documents de référence :

Document	Description
TransinfoModel	Docx ou pdf décrivant les objets retournés par l'API

HISTORIQUE DES VERSIONS

Version	Chapitres	Date	Etablie par	Description modification
1.0	Sans Objet	19/02/2013	Sébastien BARTOLI	Création
1.1	Sans Objet	02/12/2013	Sébastien BARTOLI	MAJ

Le contenu du présent document demeure propriété de la société CITYWAY. Il ne peut être ni diffusé, ni reproduit sans un accord écrit de la société CITYWAY. De même toutes les informations contenues dans ce document ne seront ni divulguées, ni révélées, sans l'accord préalable de la société CITYWAY.

SOMMAIRE

1	INTRODUCTION	3
1.1	Objet du document.....	3
1.2	API.....	3
1.3	Principe de communication	3
1.4	Type de retour	4
2	L'API « TIMETABLES »	5
2.1	Définition	5
2.2	Méthodes de l'API v1	5
2.2.1	GetLineHours.....	5
2.2.2	GetStopHours	6
2.2.3	GetVehicleJourneyHours	7
2.2.4	GetJourneyLineHours	8
2.2.5	GetNextStopHours.....	9
2.2.6	GetStopsHours.....	9
2.2.7	GetNextStopPassingTime.....	10
2.2.8	GetNextStationPassingTimes	11
2.2.9	GetNextVehicleJourneyHoursFromPoint	11

1 Introduction

1.1 Objet du document

Ce document spécifie l'API « NOMDELAPI » et les méthodes qu'elle expose par le Web Service.

1.2 API

Une API est l'ensemble de méthodes qui seront exposées par le Web Service.

Les APIs représentent des périmètres fonctionnels différents :

1.3 Principe de communication

Le service expose les fonctionnalités du SIV à travers des API constituées de méthodes.

Pour interroger le service, il suffit simplement d'appeler une URL selon le format suivante :

`http://[host]/api/NOMDELAPI/[version]/[méthode]/[format]?key=[clef utilisateur]&([param]=[valeur])*`

La sémantique des différents composants de l'URL étant la suivante :

Composant	Description
host	Url de base
api	Nom de l'API sollicitée
version	Version de l'API sollicitée
méthode	Méthode de l'API invoquée
format	Format de la réponse souhaité. Suivant la méthode invoquée, les trois formats suivants peuvent être reconnus : xml pour un format XML, json pour un format JSON. Kml pour un format KML (Si utilisable)
clef	Un identifiant unique qualifiant l'utilisateur ou l'application.
param/valeur	Un paramètre et sa valeur reconnus par la méthode. Le nombre et la teneur de ces paramètres varient d'une méthode à une autre.

Pour accéder à l'ensemble des méthodes de l'API, la clef doit être obligatoirement spécifiée dans l'url. Cette clef, selon le contexte, représente soit un utilisateur, soit une application. Cette clef est associée à une configuration permettant de définir les accès de l'API et de ses méthodes, ainsi que d'appliquer des limitations d'usage.

1.4 Type de retour

L' API retourne toujours un objet de type « **Response** ».

Cet objet contient toujours les informations suivantes :

Paramètre	Type	Description
StatusCode	Entier	Le StatusCode le status de l'appel au web service. Il peut prendre 4 différentes valeurs : <ul style="list-style-type: none">- 200 : La requête a été exécutée avec succès et renvoi un résultat non vide- 300 : La requête a été exécutée avec succès mais ne renvoi aucune solution- 400 : La requête n'a pas été exécutée car certains paramètres sont incorrects ou manquants- 500 : La requête a échoué
Message	Chaine de caractères	Le message peut indiquer une information en adéquation avec le « StatusCode » avec par exemple l'erreur qu'il y a eu lors d'un « StatusCode 500 »
Data	Objet ou liste d'objet	« Data » est le résultat proprement dit que la requête renvoi. Il peut contenir un objet ou une liste d'objet suivant la requête passée.

Les objets sont détaillés dans le document **TransinfoModel**

2 L'API « TimeTables »

2.1 Définition

Cette API permet d'obtenir les données relatives aux horaires

Url d'appel

`http://[host]/api/TimeTables/v[version]/[méthode]/[format]?key=[clef utilisateur]&([param]=[valeur])*`

2.2 Méthodes de l'API v1

2.2.1 GetLineHours

Permet d'obtenir les horaires d'une ligne

Appel de la méthode

`http://[host]/api/TimeTables/v1/GetLineHours/[format]?key=[clef utilisateur]&([param]=[valeur])*`

Formats de sortie possibles :

- XML
- JSON

Paramètres

Paramètre	Multiplicité	Type	Description
LineId	1 :1	Int	ID de la ligne
LineDirection	1 :1	Int	Direction de la ligne (1 ou 2)
Date	1 :1	String	Date (yyyy-MM-dd)
DepartureTime	0 :1	String	Heure de départ (HH-mm)
ArrivalTime	0 :1	String	Heure d'arrivée (HH-mm)
MaxItems	0 :1	Int	Nombre d'horaire retournés au maximum
IndexItems	0 :1	Int	Index du 1 ^{er} item de la liste des horaires retournés (pour la pagination)
UserRequestId	0 :1	string	Id de la requête qui servira pour les stats

Réponse

Le type de retour est une liste de LineTimeTable (Voir [LINETIME TABLE SUR LE DOCUMENT TRANSINFO MODEL](#))

Exemple d'interrogation

`http://host/TimeTables/v1/GetLineHours/xml?key=DEMO&LineId=3&LineDirection=1&Date=2013-01-22`

```

- <Response>
- <Data>
  <IdList i:nil="true"/>
- <Line>
  - <AccessibilityStatus>
    <BlindAccess>0</BlindAccess>
    <DeafAccess>0</DeafAccess>
    <MentalIllnessAccess>0</MentalIllnessAccess>
    <WheelChairAccess>0</WheelChairAccess>
  <AccessibilityStatus>
  - <Company>
    <Id>1</Id>
    <Name>ST2N</Name>
  <Company>
    <CompanyId>1</CompanyId>
    <Deleted>>false</Deleted>
  - <DirectionList>
    - <LineDirection>
      <Direction>1</Direction>
      <Name>Ariane / Général Saramito</Name>
    <LineDirection>
  <DirectionList>
    <Id>7</Id>
    <Name>Promenade des Arts - Ariane / Général Saramito</Name>
  - <Network>
    <Id>1</Id>
    <Name>Lignes d'Azur</Name>
  <Network>
    <NetworkId>1</NetworkId>
    <Number>16</Number>
  - <Operator>
    <Id>1</Id>
    <Name>LIGNES d'AZUR - NCA</Name>
  <Operator>
    <OperatorId>1</OperatorId>
    <Order>16</Order>
    <Published>>true</Published>
    <TransportMode>Bus</TransportMode>
  <Line>
    - <StopHoursList>
      - <StopHours>
        - <StopPassingTimeList>
          - <StopPassingTime>
            <JourneyPatternId>0</JourneyPatternId>
            <PassingTime>727</PassingTime>
            <VehicleJourneyId>209620</VehicleJourneyId>
          <StopPassingTime>
          - <StopPoint>
            <Id>3476</Id>
            <Latitude>43.70117349</Latitude>
          <Locality>
            <Id>6088</Id>
            <InseeCode>06088</InseeCode>
            <Latitude>0</Latitude>
            <Longitude>0</Longitude>
            <Name>NICE</Name>
          <Locality>
            <LocalityId>6088</LocalityId>
            <Longitude>7.27890949</Longitude>
            <Name>Promenade des Arts</Name>
            <PointType>Boarding_Position</PointType>
          <AccessibilityStatus>
            <BlindAccess>0</BlindAccess>
            <DeafAccess>0</DeafAccess>
            <MentalIllnessAccess>0</MentalIllnessAccess>
            <WheelChairAccess>0</WheelChairAccess>
          <AccessibilityStatus>
          - <LogicalStop>
            <Id>28053</Id>
            <Locality i:nil="true"/>
            <LocalityId>0</LocalityId>
            <Name>Promenade des Arts</Name>
          <LogicalStop>
            <LogicalStopId>28053</LogicalStopId>
          <StopPoint>
          <StopHours>
          + <StopHours></StopHours>
          + <StopHours></StopHours>
          + <StopHours></StopHours>
          </StopHoursList>
          <VehicleJourneyCount>77</VehicleJourneyCount>
          <VehicleJourneyIndex>33</VehicleJourneyIndex>
        - <VehicleJourneyList>
          - <VehicleJourney>
            <Code>11496</Code>
            <Id>209620</Id>
            <JourneyPatternId>14675</JourneyPatternId>
            <TransportMode>Bus</TransportMode>
          <VehicleJourney>
            <VehicleJourneyList>
          </Data>
          <Message>
          <StatusCode>200</StatusCode>
        </Response>

```

2.2.2 GetStopHours

Permet d'obtenir les horaires d'un arrêt

Appel de la méthode

http://[host]/api/TimeTables/v1/GetStopHours/[format]?key=[clef utilisateur]&([param]=[valeur])*

Formats de sortie possibles :

- XML
- JSON

Paramètres

Paramètre	Multiplicité	Type	Description
StopId	1 :1	Int	Identifiant du point d'arrêt
Date	1:1	String	Une date yyyy-MM-dd
LineId	1 :1	String	Identifiant de la ligne
LineDirection	1 :1	Int	Snes de la ligne (1 ou 2)
UserRequestId	0 :1	Short	Id de la requête qui servira pour les stats
CalcMode	0 :1	string	Mode de calcul pour les horaires (par défaut théorique sinon PREDICTED ou REALTIME)

Réponse

Le type de retour est une liste de StopTimeTable (Voir [STOPTIME TABLE SUR LE DOCUMENT TRANSINFOMODEL](#))

Exemple d'interrogation

```
http://host/TimeTables/v1/GetStopHours/xml?key=DEMO&StopId=221461&LineId=913&LineDirection=2&Date=2013-01-22
```

```
<Response>
  <Data>
    <EndHour>1226</EndHour>
    <JourneyPatternList>
      <JourneyPattern>
        <Direction>
          <Destination>Blache</Destination>
          <Direction>2</Direction>
          <Name>Blache</Name>
          <Origin>La Baume</Origin>
        </Direction>
        <JourneyPatternId>6294</JourneyPatternId>
      </JourneyPattern>
    </JourneyPatternList>
  </Data>
  + <Line></Line>
  <StartHour>394</StartHour>
  + <StopPassingTimeList></StopPassingTimeList>
  + <StopPoint></StopPoint>
  + <VehicleJourneyList></VehicleJourneyList>
</Data>
<Message>
  <StatusCode>200</StatusCode>
</Response>
```

2.2.3 GetVehicleJourneyHours

Permet d'obtenir les horaires d'une ligne pour une course précise

Appel de la méthode

```
http://[host]/api/TimeTables/v1/GetVehicleJourneyHours/[?mobile?]0,1/[format]?key=[clef utilisateur]&([param]=[valeur])*
```

Formats de sortie possibles :

- XML + version mobile
- JSON + version mobile

Paramètres

Paramètre	Multiplicité	Type	Description
StopId	1 :1	Int	Identifiant du point d'arrêt
Date	1:1	String	Une date yyyy-MM-dd
Time	1 :1	String	Une heure en HH-mm
LineId	1 :1	String	Identifiant de la ligne
LineDirection	1 :1	Int	Snes de la ligne (1 ou 2)
UserRequestId	0 :1	Short	Id de la requête qui servira pour les stats
VehicleJourneyId	0 :1 (n'existe pas sur la version mobile)	Int	Id de la course

Réponse

Le type de retour est une liste de JourneyTimeTable (Voir [JOURNEYTIME TABLE SUR LE DOCUMENT TRANSINFOMODEL](#))

Exemple d'interrogation

http://host/TimeTables/v1/GetVehicleJourneyHours/xml?key=DEMO&StopId=221461&LineId=913&LineDirection=2&Date=2013-01-22&Time=10-15

```
<ResponseOfJourneyTimeTable>
  <Data>
    <StopPassingTimeList>
      + <StopPassingTime></StopPassingTime>
      + <StopPassingTime></StopPassingTime>
      + <StopPassingTime></StopPassingTime>
      + <StopPassingTime></StopPassingTime>
    </StopPassingTimeList>
    <StopPassingTime>
      <StopId>220675</StopId>
      <PassingTime>604</PassingTime>
      <RealTime xsi:nil="true"/>
      <PredictedTime xsi:nil="true"/>
      <VehicleJourneyId>120418</VehicleJourneyId>
      <JourneyPatternId>0</JourneyPatternId>
    </StopPassingTime>
  </Data>
  <StatusCode>200</StatusCode>
  <Message/>
</ResponseOfJourneyTimeTable>
```

2.2.4 GetJourneyLineHours

Permet d'obtenir les horaires d'une ligne pour un itinéraire

Appel de la méthode

http://[host]/api/TimeTables/v1/GetJourneyLineHours/[format]?key=[clef utilisateur]&([param]=[valeur])*

Formats de sortie possibles :

- XML
- JSON

Paramètres

Paramètre	Multiplicité	Type	Description
JourneyId	1 :1	Int	ID de l'itinéraire
Date	1:1	String	Une date yyyy-MM-dd
LineId	1 :1	String	Identifiant de la ligne
LineDirection	1 :1	Int	Snes de la ligne (1 ou 2)
UserRequestId	0 :1	Short	Id de la requête qui servira pour les stats
CalcMode	0 :1	string	Mode de calcul pour les horaires (par défaut théorique sinon PREDICTED ou REALTIME)

Réponse

Le type de retour est une liste de LineTimeTable (Voir [LINETIME TABLE SUR LE DOCUMENT TRANSINFOMODEL](#))

Exemple d'interrogation

```
http://host/TimeTables/v1/GetJourneyLineHours/xml?key=DEMO&JourneyId=221461&LineId=913&LineDirection=2&Date=2013-01-22
```

2.2.5 GetNextStopHours

Permet d'obtenir les prochains horaires à l'arrêt

Appel de la méthode

```
http://[host]/api/TimeTables/v1/GetNextStopHours/[format]?key=[clef utilisateur]&([param]=[valeur])*
```

Formats de sortie possibles :

- XML
- JSON

Paramètres

Paramètre	Multiplicité	Type	Description
StopId	1 :1	Int	ID de l'arrêt
DateTime	1:1	String	Une date yyyy-MM-dd_HH-mm
CalcMode	1 :1	String	Mode de calcul (TimeComputeMode du TransinfoModel)
UserRequestId	0 :1	Short	Id de la requête qui servira pour les stats

Réponse

Le type de retour est une liste de StopTimeTable (Voir [STOPTIME TABLE SUR LE DOCUMENT TRANSINFOMODEL](#))

Exemple d'interrogation

```
http://host/TimeTables/v1/GetNextStopHours/xml?key=DEMO&StopId=221461&DateTime=2013-01-22_15-44
```

2.2.6 GetStopsHours

Permet d'obtenir les horaires à l'arrêt (logique)

Appel de la méthode

```
http://[host]/api/TimeTables/v1/GetStopsHours/[format]?key=[clef utilisateur]&([param]=[valeur])*
```

Formats de sortie possibles :

- XML
- JSON

Paramètres

Paramètre	Multiplicité	Type	Description
LogicalStopId	1 :1	Int	ID de l'arrêtlogique
Time	1 :1	String	Heure HH-mm
Date	1:1	String	Une date yyyy-MM-dd
CalcMode	1 :1	String	Mode de calcul (TimeComputeMode du TransinfoModel)
UserRequestId	0 :1	Short	Id de la requête qui servira pour les stats

Réponse

Le type de retour est une liste de StopTimeTable (Voir [STOPTIME TABLE SUR LE DOCUMENT TRANSINFOMODEL](#))

Exemple d'interrogation

```
http://host/TimeTables/v1/GetStopsHours/xml?key=DEMO&LogicalStopId=221461&Date=2013-01-22&Time=15-44
```

2.2.7 GetNextStopPassingTime

Permet d'obtenir les prochains horaires à l'arrêt pour une ligne et

Appel de la méthode

```
http://[host]/api/TimeTables/v1/GetNextStopPassingTime/[format]?key=[clef utilisateur]&([param]=[valeur])*
```

Formats de sortie possibles :

- XML
- JSON

Paramètres

Paramètre	Multiplicité	Type	Description
StopId	1 :1	Int	ID de l'arrêt
LineId	1 :1	Int	ID de la ligne
LineDirection	1 :1	Int	Sens de la ligne
Time	1 :1	String	Heure HH-mm
Date	1:1	String	Une date yyyy-MM-dd
RealTimeOptions	1 :1	String	Options pour le temps réel (voir RealTimeOption sur le TransinfoModel)

Réponse

Le type de retour est une liste de StopTimeTable (Voir [STOPTIME TABLE SUR LE DOCUMENT TRANSINFOMODEL](#))

Exemple d'interrogation

http://host/TimeTables/v1/GetNextStopPassingTime/xml?key=DEMO&StopId=221461&LineId=1&LineDirection=1&Date=2013-01-22&Time=15-44&realTimeOptions=SIRIONLY

2.2.8 GetNextStationPassingTimes

Permet d'obtenir les N prochains horaires au départ et à l'arrivée pour les gares SNCF

Appel de la méthode

http://[host]/api/TimeTables/v1/GetNextStationPassingTimes/[format]?key=[clef utilisateur]&([param]=[valeur])*

Formats de sortie possibles :

- XML
- JSON

Paramètres

Paramètre	Multiplicité	Type	Description
StopId	1 :1	Int	ID de l'arrêt
DateTime	1:1	String	Une date yyyy-MM-dd_HH-mm
number	1 :1	int	Nombre de retours souhaités

Réponse

Le type de retour est une liste de StationPassingTime (Voir [STATIONPASSINGTIME SUR LE DOCUMENT TRANSINFOMODEL](#))

Exemple d'interrogation

http://host/TimeTables/v1/GetNextStationPassingTimes/xml?key=DEMO&StopId=221461&DateTime=2013-01-22_15-44&number=10

2.2.9 GetNextVehicleJourneyHoursFromPoint

Permet d'obtenir les horaires de la prochaine course pour un point à une date

Appel de la méthode

http://[host]/api/TimeTables/v1/GetNextVehicleJourneyHoursFromPoint/[format]?key=[clef utilisateur]&([param]=[valeur])*

Formats de sortie possibles :

- XML
- JSON

Paramètres

Paramètre	Multiplicité	Type	Description
JourneyId	1 :1	Int	ID de la course

Réponse

Le type de retour est une liste de StationPassingTime (Voir [STATIONPASSINGTIME SUR LE DOCUMENT TRANSINFOMODEL](#))

Exemple d'interrogation

<http://host/TimeTables/v1/GetNextVehicleJourneyHoursFromPoint/xml?key=DEMO&JourneyId=221461>