Contents

Legal Notices .......................................................................................................................... 4
Document Information ......................................................................................................... 5

Chapter 1: Overview ........................................................................................................... 6
  What is the Map Tile API? ................................................................................................. 7
  Why use the Map Tile API? ............................................................................................ 7

Chapter 2: Quick Start ....................................................................................................... 8
  Making the First Request ............................................................................................... 9

Chapter 3: User Guide ....................................................................................................... 11
  Acquiring Credentials .................................................................................................. 12
  Constructing a Request ................................................................................................. 12
    Customer Integration Testing .................................................................................... 14
  Key Concepts ................................................................................................................ 14
    Mercator Projection ..................................................................................................... 15
    Using a Tile Address ................................................................................................... 17
  Requesting the Right Map Version ............................................................................. 17
  Traffic ............................................................................................................................ 19
  Examples ......................................................................................................................... 23
    Normal Day Tile ......................................................................................................... 23
    Normal Day Grey Tile ............................................................................................... 24
    Terrain Map Tile ....................................................................................................... 26
    Satellite Map Tile ...................................................................................................... 27
    Hybrid Map Tile ........................................................................................................ 29
    Normal Day Transit Map Tile ................................................................................... 30
    Fleet Tile ................................................................................................................... 32
    Truck Tile .................................................................................................................. 34
Chapter 4: API Reference

Resources.........................................................................................................................44
Maptile............................................................................................................................44
Info....................................................................................................................................50
Copyright.........................................................................................................................54
Version..............................................................................................................................61
HTTP Status Codes..........................................................................................................62
Legal Notices

© 2013 HERE. All rights reserved.

This material, including documentation and any related computer programs, is protected by copyright controlled by HERE. All rights are reserved. Copying, including reproducing, storing, adapting or translating, any or all of this material requires the prior written consent of HERE. This material also contains confidential information, which may not be disclosed to others without the prior written consent of HERE.

Trademark Acknowledgements

HERE and Nokia are trademarks or registered trademarks of Nokia Corporation in the United States and other countries.

Other trade names are trademarks or registered trademarks of their owners.

Disclaimer

This content is provided “as-is” and without warranties of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability, fitness for a particular purpose, satisfactory quality and non-infringement. Nokia does not warrant that the content is error free and Nokia does not warrant or make any representations regarding the quality, correctness, accuracy, or reliability of the content. you should therefore verify any information contained in the content before acting on it.

To the furthest extent permitted by law, under no circumstances, including without limitation Nokia’s negligence, shall Nokia be liable for any damages, including, without limitation, direct, special, indirect, punitive, consequential, exemplary and/or incidental damages that result from the use or application of this content, even if Nokia or an authorized representative has been advised of the possibility of such damages.
## Document Information

<table>
<thead>
<tr>
<th>Product</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name:</td>
<td>Map Tile API</td>
</tr>
<tr>
<td>Version:</td>
<td>Version 2.1.32.1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Document</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name:</td>
<td>Map Tile API Developer's Guide</td>
</tr>
<tr>
<td>Id:</td>
<td>c3ec9b4-1383581503</td>
</tr>
<tr>
<td>Status:</td>
<td>FINAL</td>
</tr>
<tr>
<td>Date:</td>
<td>2013-Nov-04, 16:12 (GMT)</td>
</tr>
</tbody>
</table>
Overview

Topics:
• What is the Map Tile API?
• Why use the Map Tile API?

This document introduces the Map Tile API and:
• explains key concepts
• provides examples
• documents resources and query parameters
• documents response structures and data types
What is the Map Tile API?

HERE Map Tile API is a RESTful API that retrieves map images for all regions of the world.

To get a map image, formulate a request that combines the URL and a set of parameters to specify details such as position, format, zoom level, map type of the map image. You can embed the resulting map image in web pages and applications.

Why use the Map Tile API?

Map Tile API addresses for the following high level use cases.

- get a map tile image
- specify the viewing scheme of a map tile image
- specify the zoom level of a map tile image
- specify the resolution of a map tile image
- identify different map versions
- retrieve map copyright information
Chapter 2

Quick Start

Topics:

- Making the First Request

This article demonstrates the simplest way to use the Map Tile API.
Making the First Request

The simplest request scenario is to obtain the most up-to-date day-time map image that includes a particular location.

The Map Tile API uses a grid of map tiles that, combined, form a complete map of the world reflecting the Normalized Mercator projection.

Thus, a request corresponding to the simple scenario above must provide:

- the unique URL of the API resource that is to process the request for a map image
- path variables that specify the details of the request (zoom level, map tile location in the grid, etc.)
- authentication parameters

The authentication parameters are `app_id` and `app_code`. Their values are the authentication credentials required for access to the API resources. Note that demo credentials are available and can be used for testing, but application-specific credentials are required in commercial applications – see Acquiring Credentials on page 12 for further information.

The example below shows a request for a map tile at zoom level 13, located in 4400th column and 2686th row in the tile grid (to understand how these tile coordinates were obtained, see Mercator Projection on page 15). The tile resolution is 256x256 pixels, and the tile format PNG8.

http://1.base.maps.cit.api.here.com/maptile/2.1/maptile/newest/normal.day/13/4400/2686/256/png8?
app_id=DemoAppId01082013GAL
app_code=AJKnXv84fjrb0KIHawS0Tg
The response delivers a map tile image of part of Berlin, Germany.

Figure 1: A location in Berlin, Germany
Chapter 3

User Guide

Topics:

- Acquiring Credentials
- Constructing a Request
- Key Concepts
- Requesting the Right Map Version
- Traffic
- Examples
- Service Support

The articles in this section provide a guide to using the Map Tile API.
Acquiring Credentials

All users of HERE APIs must obtain authentication and authorization credentials and provide them as values for the parameters `app_id` and `app_code`. The credentials are assigned per application.

To obtain the credentials for an application, please visit [http://developer.here.com/get-started](http://developer.here.com/get-started) for more details.

Constructing a Request

A request to the Map Tile API includes the basic elements shown in the following table and, in addition, it may contain resource-specific parameters.

### Table 1: Basic request elements

<table>
<thead>
<tr>
<th>Element</th>
<th>Value/Example</th>
<th>Note</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base URL</td>
<td><strong>Map Tiles</strong></td>
<td></td>
<td>Production environment</td>
</tr>
<tr>
<td></td>
<td>http://{1-4}.base.maps.api.here.com</td>
<td></td>
<td>See also Load Balancing and URLs on page 13</td>
</tr>
<tr>
<td></td>
<td><strong>Aerial Tiles</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>http://{1-4}.aerial.maps.api.here.com</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Traffic Tiles</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>http://{1-4}.traffic.maps.api.here.com</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Pano Tiles</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>http://{1-4}.pano.maps.api.here.com</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Map Tiles</td>
<td>http://{1-4}.base.maps.cit.api.here.com</td>
<td></td>
<td>See also Load Balancing and URLs on page 13</td>
</tr>
<tr>
<td>Aerial Tiles</td>
<td>http://{1-4}.aerial.maps.cit.api.here.com</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Traffic Tiles</td>
<td>http://{1-4}.traffic.maps.cit.api.here.com</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

CIT environment: see Customer Integration Testing on page 14
## Pano Tiles

http://{1-4}.pano.maps.cit.api.here.com

### Path

```
/maptile/2.1/
```

### Resource

<table>
<thead>
<tr>
<th>Element</th>
<th>Value/Example</th>
<th>Note</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path</td>
<td>/maptile/2.1/</td>
<td>[1]</td>
<td>GET only, specify request details via query parameters. See Maptile on page 44, Info on page 50 or Copyright on page 54 for a more detailed parameter list for each resource.</td>
</tr>
<tr>
<td>Maptile</td>
<td>http://{1-4}.pano.maps.cit.api.here.com</td>
<td>[1]</td>
<td>Please use normal tile base URL for this request. Keep in mind that for some schemes you should use the aerial tile base URLs, check in the API definition.</td>
</tr>
<tr>
<td>Base Tile</td>
<td>basetile/</td>
<td>[1]</td>
<td>Please use pano tile base URL for this request.</td>
</tr>
<tr>
<td>Label Tile</td>
<td>labeltile/</td>
<td>[1]</td>
<td>Please use the right base URL depending on which environment do you want to get information from.</td>
</tr>
<tr>
<td>Rctile</td>
<td>rctile/</td>
<td>[2]</td>
<td>Please use traffic tile base URL for this request.</td>
</tr>
<tr>
<td>Rconlytile</td>
<td>rconlytile/</td>
<td>[2]</td>
<td></td>
</tr>
<tr>
<td>Rcdistonlytile</td>
<td>rcdistonlytile/</td>
<td>[2]</td>
<td></td>
</tr>
<tr>
<td>Truck Tile</td>
<td>trucktile/</td>
<td>[1]</td>
<td></td>
</tr>
<tr>
<td>Truckonlytile</td>
<td>truckonlytile/</td>
<td>[1]</td>
<td></td>
</tr>
<tr>
<td>Info</td>
<td>info/</td>
<td>[3]</td>
<td></td>
</tr>
<tr>
<td>Copyright</td>
<td>copyright/</td>
<td>[3]</td>
<td></td>
</tr>
<tr>
<td>Version</td>
<td>version/</td>
<td>[3]</td>
<td></td>
</tr>
<tr>
<td>Traffictile</td>
<td>traffictile/</td>
<td>[4]</td>
<td></td>
</tr>
<tr>
<td>Flowtile</td>
<td>flowtile/</td>
<td>[4]</td>
<td></td>
</tr>
</tbody>
</table>

### Application Code

```
?app_code=AJKnXv84fjrb0KIHawS0Tg
```

**Notes:**

[1] Please use normal tile base URL for this request. Keep in mind that for some schemes you should use the aerial tile base URLs, check in the API definition.

[2] Please use pano tile base URL for this request.

[3] Please use the right base URL depending on which environment do you want to get information from.

[4] Please use traffic tile base URL for this request.

## Load Balancing and URLs

The servers that process requests against the Map Tile API resources are distributed for optimized load balancing. Clients can take advantage of this and improve response times, especially when submitting multiple simultaneous requests, by using URL prefixes (1 - 4). The numeric prefix must be the first part of the request URL after "http://".
There are different ways to take advantage of these mechanism, no method is enforced from the server, although the method used should ensure that your requests are distributed among the different prefixes and do not use only one. We propose the following methods:

- Choose the prefix randomly.
- Choose the prefix depending on the parity of the row and column number:
  1. when both the row and the column number are even
  2. when the row number is even, the column number is odd
  3. when the row number is odd, the column number is even
  4. when both the row and the column number are odd
- Use a simple formula such as:
  \[ 1 + ((\text{row\_number} + \text{column\_number}) \mod 4) \]

In this formula, \( \mod \) indicates modulo division.

**Customer Integration Testing**

HERE is committed to maintain the best possible production service for all customers. Given that the production environment is live and common to all API users, we request that you use the alternative Customer Integration Testing (CIT) environment when evaluating our products, running tests, making changes in your code and altering the way you access our APIs.

The CIT environment also allows you to test your software against a newer version of the service before HERE brings that version into production. CIT offers a fully functional environment for customers to use for development and testing, but it does not support high loads or performance testing in general.

Note that the same application id can be used in both environments, but CIT may require a dedicated application code. If this is the case, please contact us as described under *Service Support* on page 42.

The CIT environment is not intended for general production use.

**Key Concepts**

This section provides insights into the key concepts used throughout the Map Tile API.
Mercator Projection

The Map Tile API serves map tiles obtained by mapping points on the surface of a sphere (the globe) to points on a plane, using the normalized Mercator projection.

The basic Mercator projection formula is this:

$\{\lambda, \varphi\} \rightarrow x[-1, 1] \ y\ [-1, 1]$

In this formula:

$\lambda = \text{longitude}$
$\varphi = \text{latitude}$

$x = \lambda / \pi$

$y = \ln(\tan(\pi/4 + \varphi/2)) / \pi$

The plane represents the globe as a square grid of map tiles. The size of the grid depends on the map zoom level. At the lowest zoom level, the world is contained in one map tile. At the next higher zoom level, the world is two tiles wide and two tiles high (2x2), at the next level above that, the grid is 4x4, then 8x8, 16x16, and so on up to the maximum zoom for a particular region. In other words, at each zoom level the tiles that make up the complete map of the world form a grid in which the number of tiles is equal to the zoom level raised to the power of two ($zoom^2$).

The relationship between tiles at two consecutive zoom levels can be expressed as follows:

$col_{z+1} = (2*col_z) + 1$
$row_{z+1} = (2*row_z) + 1$

In this formula:

$col = \text{column number in the tile grid}$
$row = \text{row number in the tile grid}$
$z = \text{zoom level}$
The diagram below demonstrates this graphically:

**Figure 2: Tiles at different zoom levels**

You can use this information to obtain the grid coordinates (row and column) of the map tile for a particular geographic location in your application. The following pseudo code contains the complete algorithm:

```javascript
--- javascript ---
var lat = 52.525439, // Latitude
    lon = 13.38727,      // Longitude
    z = 12,              // Zoom level
    latRad,
    n,
    xTile,
    yTile;

latRad = lat * Math.PI / 180;
    n = Math.pow(2, z);
    xTile = n * ((lon + 180) / 360);
    yTile = n * (1-(Math.log(Math.tan(latRad) + 1/Math.cos(latRad)) / Math.PI)) / 2;

--- output ---
    lat_rad = 0.916
    n = 4096
    xTile = 2200.31 // Column
    yTile = 1343.20 // Row
```

The zoom level and tile row and column can be used as URL variables separated by the '/' character in map tile requests. Note that they must be provided in this order: zoom/column/row. This is the [Z]/[X]/[Y] addressing scheme.
The map tile specification is typically preceded by other path variables and may be followed either by further path variables and/or query parameters.

### Using a Tile Address

A request that includes a tile address in the format `zoom/column/row` can be formulated as follows:

```plaintext
normal.day/12/2200/1343/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```

In this case, the request obtains a map tile at zoom level 12 for part of Berlin, Germany:

**Figure 3: A map tile for part of Berlin**

---

### Requesting the Right Map Version

The Map Tile API offers:

- regularly updated maps
• different map versions for certain regions of the world to meet the legal requirements in specific countries

The response to a metadata request lists all the available map versions as elements named map, grouped by the attribute region. The response identifies the most recent map version in each region with the attribute id set to the value "newest". The following excerpt shows an example of a map version list:

```xml
<maps>
  <map region="all" id="newest" />
  <map region="all"
       newest="true"
       id="0.2.45.110"
       hash="c376276c01" />
</maps>
```

When requesting map tiles, use URL variables to indicate the most recent map or a specific map version:

• **newest** – to request the most recent map version available, for example:

```
  normal.day/11/1202/843/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```

• a hash value – to request a specific map version; the actual URL variable is the value of the attribute hash from the response to a metadata request (see the response excerpt above), for example; here is a map tile request that uses a hash value instead of /newest/.

Note that the hash values identifying map versions change as a result of software updates, even if the map ids remain the same. To get a valid response to the query, make a metadata request request first, and substitute a current map hash value for c376276c01 in the following URL:

```
http://2.base.maps.cit.api.here.com/maptile/2.1/maptile/c376276c01/
  normal.day/11/1202/843/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```

**Note:** Metadata requests retrieve information specific to the base URLs they target. This information cannot be used in subsequent requests that use other base URLs. For example, to obtain a traffic map version, use metadata from the response to a request addressed to the Traffic Tiles base URL, but for a version of aerial map, use metadata from the response to a request addressed to Aerial Tiles base URL.
Traffic

This article provides basic information about traffic tiles, explains how to access them and the available options. It also demonstrates the visual styles used by the different types of traffic tiles.

Traffic tiles display current (always up-to-date) traffic information. The information reflects congestion and is presented to the user by the lines drawn over or along the affected streets and roads. The line color depends on the information to be conveyed (colors are configured using the scheme attribute on the request). For example in the normal.day scheme, the meaning of colors is as follows:

- **green**: no congestion or very light traffic congestion
- **yellow**: medium traffic congestion
- **red**: heavy traffic congestion
- **black**: road blockage

It is important to bear in mind that traffic information remains valid only for a short period of time (approximately 15 minutes). For this reason, client applications should cache traffic tiles only briefly or should not use caching at all.

As with map data, traffic information is not available in all countries and in some countries availability is limited to certain areas. The table below provides the details of coverage.

Table 2: Traffic data availability

<table>
<thead>
<tr>
<th>Country name</th>
<th>Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>all areas</td>
</tr>
<tr>
<td>Belgium</td>
<td>all areas</td>
</tr>
<tr>
<td>Denmark</td>
<td>all areas</td>
</tr>
<tr>
<td>Finland</td>
<td>all areas</td>
</tr>
<tr>
<td>France</td>
<td>all areas</td>
</tr>
<tr>
<td>Germany</td>
<td>all areas</td>
</tr>
<tr>
<td>Greece</td>
<td>all areas</td>
</tr>
<tr>
<td>Ireland</td>
<td>all areas</td>
</tr>
<tr>
<td>Italy</td>
<td>all areas</td>
</tr>
<tr>
<td>Luxembourg</td>
<td>all areas</td>
</tr>
<tr>
<td>Country name</td>
<td>Availability</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Netherlands</td>
<td>all areas</td>
</tr>
<tr>
<td>Norway</td>
<td>all areas</td>
</tr>
<tr>
<td>Poland</td>
<td>all areas</td>
</tr>
<tr>
<td>Portugal</td>
<td>all areas</td>
</tr>
<tr>
<td>Russia</td>
<td>all areas</td>
</tr>
<tr>
<td>Spain</td>
<td>all areas</td>
</tr>
<tr>
<td>Sweden</td>
<td>all areas</td>
</tr>
<tr>
<td>Switzerland</td>
<td>all areas</td>
</tr>
<tr>
<td>Turkey</td>
<td>all areas</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>all areas</td>
</tr>
<tr>
<td>Ukraine</td>
<td>all areas</td>
</tr>
<tr>
<td>Saudi Arabia</td>
<td>all areas</td>
</tr>
<tr>
<td>South Africa</td>
<td>all areas</td>
</tr>
<tr>
<td>UAE</td>
<td>all areas</td>
</tr>
<tr>
<td>Argentina</td>
<td>all areas</td>
</tr>
<tr>
<td>Brazil</td>
<td>all areas</td>
</tr>
</tbody>
</table>
| Canada       | Hamilton-Burlington - ON  
                Kitchener - ON  
                London - ON  
                Windsor - ON  
                St. Catharines-Niagara Falls-Welland - ON  
                Oshawa-Whitby-Clarington - ON  
                Toronto-Mississauga - ON  
                Ottawa-Gatineau - ON-QC  
                Montreal-Laval - QC  
                Quebec - QC  
                Vancouver-Surrey-Burnaby - BC  
                Edmonton - AB  
                Calgary - AB |
| Mexico       | Mexico City  
                Guadalajara  
                Monterrey |
<table>
<thead>
<tr>
<th>Country name</th>
<th>Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td>USA</td>
<td>Detailed coverage info can be found <a href="#">here</a></td>
</tr>
<tr>
<td>Australia</td>
<td>Canberra Metro</td>
</tr>
<tr>
<td></td>
<td>Sydney Metro</td>
</tr>
<tr>
<td></td>
<td>Melbourne Metro</td>
</tr>
<tr>
<td></td>
<td>Brisbane Metro</td>
</tr>
<tr>
<td></td>
<td>Adelaide Metro</td>
</tr>
<tr>
<td></td>
<td>Perth Metro</td>
</tr>
<tr>
<td></td>
<td>Hobart</td>
</tr>
<tr>
<td></td>
<td>Darwin</td>
</tr>
<tr>
<td>China</td>
<td>Beijing</td>
</tr>
<tr>
<td></td>
<td>Changchun</td>
</tr>
<tr>
<td></td>
<td>Changsha</td>
</tr>
<tr>
<td></td>
<td>Chengdu</td>
</tr>
<tr>
<td></td>
<td>Chongqing</td>
</tr>
<tr>
<td></td>
<td>Dongguan</td>
</tr>
<tr>
<td></td>
<td>Foshan</td>
</tr>
<tr>
<td></td>
<td>Fuzhou</td>
</tr>
<tr>
<td></td>
<td>Guangzhou</td>
</tr>
<tr>
<td></td>
<td>Hangzhou</td>
</tr>
<tr>
<td></td>
<td>Nanjing</td>
</tr>
<tr>
<td></td>
<td>Ningbo</td>
</tr>
<tr>
<td></td>
<td>Qingdao</td>
</tr>
<tr>
<td></td>
<td>Shanghai</td>
</tr>
<tr>
<td></td>
<td>Shenyang</td>
</tr>
<tr>
<td></td>
<td>Shenzhen</td>
</tr>
<tr>
<td></td>
<td>Shijiazhuang</td>
</tr>
<tr>
<td></td>
<td>Tianjin</td>
</tr>
<tr>
<td></td>
<td>Wuhan</td>
</tr>
<tr>
<td>India</td>
<td>Delhi NCR</td>
</tr>
<tr>
<td></td>
<td>Greater Mumbai</td>
</tr>
<tr>
<td></td>
<td>Bangalore</td>
</tr>
<tr>
<td>Malaysia</td>
<td>all areas</td>
</tr>
<tr>
<td>New Zealand</td>
<td>all areas</td>
</tr>
<tr>
<td>Thailand</td>
<td>all areas</td>
</tr>
</tbody>
</table>
It is important to note that even if traffic information is available for a specific country, there may be roads or places for which it cannot be obtained. The quality and range of traffic data for the covered areas and street types (highways, main roads, local streets, etc.) vary between countries and depend on the available data.

Detailed information about traffic tile request parameters can be found under Maptile on page 44.

**A Traffic Tile Request**

The following example request obtains a basic traffic tile for the map view scheme `normal.day`. Note that the request is shown with a variable element `{type}` – for possible values, please see Traffic Tile Types on page 22.

```
http://1.traffic.maps.cit.api.here.com/maptile/2.1/{type}/newest/
normal.day/12/2200/1343/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```

**Traffic Tile Types**

In the example above, `{type}` is a variable element. Its value can be one of the following tile types:

- `traffictile/`
- `flowtile/`

The tile type specified in the request determines the visual style of the image delivered in the response. The images below demonstrate the full range of possible styles. For easy comparison, the leftmost map tile is one without traffic information, while the remaining images follow the order of tile types in the list above.

**Figure 4: Traffic information in map tiles**
Examples

This section provides examples of requests against the Map Tile API resources along with the corresponding results.

Note that all requests require the authentication credentials provided via the parameters app_id and app_code (see Acquiring Credentials on page 12). The example requests in the following articles show all the required parameters, but discuss only those specific to each request.

Normal Day Tile

This article shows a request for a normal day map tile.

User story

The user wants to obtain an image showing a map of the area north of Cairo in Egypt. The tile is to represent the normal day view of the map.

Request Summary

The following list summarizes the elements required to create a request matching the user story and shows, in square brackets, how the those elements are used in the example(s) below. Note that the request example also uses the authentication parameters.

| Resource:  | maptile [maptile], see also (resource) type |
| URL variables: | {map id} [newest], see also Requesting the Right Map Version on page 17 |
|            | {scheme} [normal.day], see also scheme |
|            | {zoom}/(column)/(row) [11/1202/843], see also Mercator Projection on page 15 |
|            | {size} [256], see also size |
|            | {format} [png8], see also format |
Request

The code block below demonstrates a complete request for a map tile.

```
  normal.day/11/1202/843/256/png8
  ?app_id=DemoAppId01082013GAL
  &app_code=AJKnXv84fjrb0KIHawS0Tg
```

Response

The response to the request delivers the following image:

Figure 5: Normal Day Tile

![Normal Day Tile](image)

Normal Day Grey Tile

This article shows a request for a color-reduced tile in daylight mode (used for background maps).

User story

The user wants to obtain an image showing a map of the area north of Cairo in Egypt. The tile is to represent the normal day grey view of the map.
Request Summary

The following list summarizes the elements required to create a request matching the user story and shows, in square brackets, how the those elements are used in the example(s) below. Note that the request example also uses the authentication parameters.

| Resource: | maptile [maptile], see also (resource) type |
| URL variables: | (map id) [newest], see also Requesting the Right Map Version on page 17 |
| | (scheme) [normal.day.grey], see also scheme |
| | (zoom)/(column)/(row) [11/1202/843], see also Mercator Projection on page 15 |
| | (size) = 256, see also size |
| | (format) = png8, see also format |

Request

The code block below demonstrates a complete request for a map tile.

```
  normal.day.grey/11/1202/843/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```
Response

The response to the request delivers the following image:

Figure 6: Normal Day Grey Tile

![Normal Day Grey Tile](image)

Terrain Map Tile

This article shows a request for a terrain map tile.

User story

The user wants to obtain an image showing a terrain map tile of the area north of Cairo in Egypt. The tile is to represent the normal day terrain view of the map.

Request Summary

The following list summarizes the elements required to create a request matching the user story and shows, in square brackets, how those elements are used in the example(s) below. Note that the request example also uses the authentication parameters.

| Resource: | maptile [maptile], see also (resource) type |
| URL variables: | {map id} [newest], see also Requesting the Right Map Version on page 17 | {scheme} [terrain.day], see also scheme | {zoom}/{column}/{row} [11/1202/843], see also Mercator Projection on page 15 |
Request

The code block below demonstrates a complete request for a terrain map tile.

```
  terrain.day/11/1202/843/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJJnXv84fjrb0KIHawS0Tg
```

Response

The response to the request delivers the following image:

![Figure 7: Terrain Map Tile](image)

Satellite Map Tile

This article shows a request for a satellite map tile in daylight mode.

**User story**

The user wants to obtain an image showing a satellite map tile of the area north of Cairo in Egypt. The tile is to represent the normal day satellite view of the map.
Request Summary

The following list summarizes the elements required to create a request matching the user story and shows, in square brackets, how those elements are used in the example(s) below. Note that the request example also uses the authentication parameters.

<table>
<thead>
<tr>
<th>Resource</th>
<th>maptile [maptile], see also (resource) type</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL variables:</td>
<td></td>
</tr>
<tr>
<td>{map id}</td>
<td>[newest], see also Requesting the Right Map Version on page 17</td>
</tr>
<tr>
<td>{scheme}</td>
<td>[satellite.day], see also scheme</td>
</tr>
<tr>
<td>{zoom}/{column}/{row}</td>
<td>[11/1202/843], see also Mercator Projection on page 15</td>
</tr>
<tr>
<td>{size}</td>
<td>[256], see also size</td>
</tr>
<tr>
<td>{format}</td>
<td>[png8], see also format</td>
</tr>
</tbody>
</table>

Request

The code block below demonstrates a complete request for a satellite map tile.

```
  satellite.day/11/1202/843/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```
Response

The response to the request delivers the following image:

Figure 8: Satellite Tile

Hybrid Map Tile

This article shows a request for a hybrid map tile in daylight mode.

User story

The user wants to obtain an image showing a hybrid map tile of the area north of Cairo in Egypt. The tile is to show roads and labels over a satellite image, representing the normal day view of the map.

Request Summary

The following list summarizes the elements required to create a request matching the user story and shows, in square brackets, how the those elements are used in the example(s) below. Note that the request example also uses the authentication parameters.

<table>
<thead>
<tr>
<th>Resource:</th>
<th>maptile [maptile], see also (resource) type</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL variables:</td>
<td>{map id} [newest], see also Requesting the Right Map Version on page 17</td>
</tr>
<tr>
<td></td>
<td>{scheme} [hybrid.day], see also scheme</td>
</tr>
<tr>
<td></td>
<td>{zoom}/{column}/{row} [11/1202/843], see also Mercator Projection on page 15</td>
</tr>
</tbody>
</table>
{size} [256], see also size

{format} [png8], see also format

Request

The code block below demonstrates a complete request for a hybrid map tile.

```
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```

Response

The response to the request delivers the following image:

Figure 9: Hybrid Tile

Normal Day Transit Map Tile

This article shows a request for a transit map tile.

User story

The user wants to obtain an image showing a transit map tile (public transport) for an area east of Vincennes (suburb of Paris). The tile is to show a normal day view of the map indicating public transport lines.
Request Summary

The following list summarizes the elements required to create a request matching the user story and shows, in square brackets, how the those elements are used in the example(s) below. Note that the request example also uses the authentication parameters.

<table>
<thead>
<tr>
<th>Resource:</th>
<th>mpxtile [maptile], see also (resource) type</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL variables:</td>
<td>{map id} [newest], see also Requesting the Right Map Version on page 17</td>
</tr>
<tr>
<td></td>
<td>{scheme} [normal.day.transit], see also scheme</td>
</tr>
<tr>
<td></td>
<td>{zoom}/{column}/{row} [13/4152/2818], see also Mercator Projection on page 15</td>
</tr>
<tr>
<td></td>
<td>{size} [256], see also size</td>
</tr>
<tr>
<td></td>
<td>{format} [png8], see also format</td>
</tr>
</tbody>
</table>

Request

The code block below demonstrates a complete request for a transit map tile.

```plaintext
normal.day.transit/13/4152/2818/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```
Response

The response to the request delivers the following image:

Figure 10: Normal Day Transit Map Tile

Fleet Tile

This article shows a request for a fleet map tile.

User story

The user wants to obtain an image showing a fleet map tile for an area west of the central station in Berlin, Germany. The tile is to show a normal day view of the map indicating public fleet information.

Request Summary

The following list summarizes the elements required to create a request matching the user story and shows, in square brackets, how the those elements are used in the example(s) below. Note that the request example also uses the authentication parameters.

<table>
<thead>
<tr>
<th>Resource:</th>
<th>maptile [maptile], see also (resource) type</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL variables:</td>
<td>{map id} [newest], see also Requesting the Right Map Version on page 17</td>
</tr>
<tr>
<td></td>
<td>{scheme} [normal.day], see also scheme</td>
</tr>
<tr>
<td></td>
<td>{zoom}/{column}/{row} [16/35201/21491], see also Mercator Projection on page 15</td>
</tr>
</tbody>
</table>
Parameters: style = ?style=fleet, see also style

Request

The code block below demonstrates a complete request for a fleet map tile.

```plaintext
normal.day/16/35201/21491/256/png8
?style=fleet
&app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```

Response

The response to the request delivers the following image:

**Figure 11: Fleet Tile**
Truck Tile

This article shows requests for truck map tiles.

User story

The user wants to obtain two images showing a truck map tile for an area west of the central station in Berlin, Germany. The first tile is to show a normal day view of the map indicating public truck information, while the second tile is to show only those roads where trucks are allowed as well as truck-related information (in other words, this tile is not to show truck information overlaid on a normal map).

Request Summary

The following list summarizes the elements required to create a request matching the user story and shows, in square brackets, how the those elements are used in the example(s) below. Note that the request example also uses the authentication parameters.

| Resource: | maptile = trucktile or truckonlytile, see also (resource) type |
| URL variables: | {map id} = newest, see also Requesting the Right Map Version on page 17 |
| | {scheme} = normal.day, see also scheme |
| | {zoom}/{column}/{row} = 16/35201/21491, see also Mercator Projection on page 15 |
| | {size} = 256, see also size |
| | {format} = png8, see also format |
| Parameters: | style = ?style=fleet, see also style |

Request for a Truck Map Tile

The code block below demonstrates a complete request for a truck map tile.

```
http://2.base.maps.cit.api.here.com/maptile/2.1/trucktile/newest/
  normal.day/16/35201/21491/256/png8
  ?style=fleet
  &app_id=DemoAppId01082013GAL
  &app_code=AJKnXv84fjrb0KIHawS0Tg
```
Response to Truck Tile Request

The response to the request delivers the following image:

**Figure 12: Truck Tile**

Request for a Truck-information-only Tile

The code block below demonstrates a request for a truck-information-only tile.

```plaintext
http://2.base.maps.cit.api.here.com/maptile/2.1/truckonlytile/newest/
normal.day/16/35201/21491/256/png8
?style=fleet
&app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```
Response to Truck-information-only Tile Request

The response contains the image below. Note that the tile contains a transparent background which is shown here in black.

Figure 13: Truck-only Tile

Traffic Tile

This article shows requests for map tiles that contain traffic information.

User story

The user wants to obtain a map tile displaying the current traffic conditions in a part of London and another tile that displays the traffic flow in the same area of London.

Request Summary

The following list summarizes the elements required to create a request matching the user story and shows, in square brackets, how the those elements are used in the example(s) below. Note that the request example also uses the authentication parameters.
Resource: maptile [traffictile] or flowtile, see also (resource) type

URL variables: {map id} [newest], see also Requesting the Right Map Version on page 17

{scheme} [normal.day], see also scheme

{zoom}/{column}/{row} [15/16358/10898], see also Mercator Projection on page 15

{size} [256], see also size

{format} [png8], see also format

Request for a Traffic Map Tile

The code block below demonstrates a complete request for a traffic map tile.

```text
http://1.traffic.maps.cit.api.here.com/maptile/2.1/traffictile/newest/
normal.day/15/16358/10898/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```
Response to Truck Traffic Map Tile Request

The response contains the following image.

Figure 14: Traffic tile for a part of London

Request for a Traffic Flow Tile

The code block below demonstrates a request for a traffic tile, using the `flowtile` resource.

http://1.traffic.maps.cit.api.here.com/maptile/2.1/flowtile/newest/normal.day/15/16358/10898/256/png8?app_id=DemoAppId01082013GAL&app_code=AJKnXv84fjrb0KIHawS0Tg
Response to Traffic Flow Tile Request

The response contains the image below. Note that the tile contains a transparent background which is shown here in black.

Figure 15: Transparent traffic tile for a part of London

Panorama Tile

This article shows requests for map tiles that contain panorama information.

User story

The user wants to see the current panorama coverage of London. The tile is to represent the parts of London currently covered with panoramas, as an overlay on top of the normal map tile.

Request Summary

The following list summarizes the elements required to create a request matching the user story and shows, in square brackets, how the those elements are used in the example(s) below. Note that the request example also uses the authentication parameters.
<table>
<thead>
<tr>
<th>Resource:</th>
<th><code>rctile</code> or <code>rconlytile</code>, see also <em>(resource) type</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>URL variables:</td>
<td><code>{map id}</code> [newest], see also Requesting the Right Map Version on page 17</td>
</tr>
<tr>
<td></td>
<td><code>{scheme}</code> [normal.day], see also <em>scheme</em></td>
</tr>
<tr>
<td></td>
<td><code>{zoom}/(column)/(row)[13/4094/2724]</code>, see also Mercator Projection on page 15</td>
</tr>
<tr>
<td></td>
<td><code>{size}</code> [256], see also <em>size</em></td>
</tr>
<tr>
<td></td>
<td><code>{format}</code> [png8], see also <em>format</em></td>
</tr>
</tbody>
</table>

### Request for a Panorama Map Tile

The code block below demonstrates a complete request for a panorama map tile.

```bash
  normal.day/13/4094/2724/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```
Response to Panorama Map Tile Request

The response contains the following image.

Figure 16: Panorama tile for a part of London

Request for a Panorama Only Tile

The code block below demonstrates a request for a panorama tile, using the ronlytile resource.

?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrboKIHawS0Tg
Response to Panorama Only Tile Request

The response contains the image below. Note that the tile contains a transparent background which is shown here in black.

Figure 17: Transparent panorama only tile for a part of London

Service Support

If you need assistance with this or other HERE products, please contact your HERE representative or Technical Customer Support via email at tcsplatform@here.com.
Chapter 4

API Reference

Topics:
- Resources
- HTTP Status Codes

Descriptions of the resources, parameters, return types and error codes of the HEREMap Tile API.

Map Tile API provides the following resources:

- Map tile request: returns a map tile image
- Traffic view/tile request: returns a map tile with a traffic overlay
- Metadata: returns information about map regions, map tile resolutions, format, and schemes
- Copyright: returns copyright information for specific map types, map areas and map zoom levels
- Version: returns the software version
Resources

This section provides a comprehensive reference to the Map Tile API resources and the parameters supported by those resources.

Maptile

This resource is responsible for servicing requests for a variety of map tile types that differ in terms of the information they include and consequently also in terms of the visual style. A response delivers a single map tile image or a set of four, nine or sixteen map tiles encoded as text.

Resource Types

To make a map tile request, use one of the following resource types:

- mptile/
- traffictile/
- flowtile/
- basetile/
- labeltile/
- rctile/
- rconlytile/
- rcdistonlytile/
- trucktile/
- truckonlytile/

A traffic tile request returns a single traffic tile image, or a set of four, nine or sixteen traffic tiles encoded as text. To make a traffic tile request use the traffictile resource type or one of the following alternative resource types:

- traffictile/
- flowtile/
Request Format

Requests against the maptile resource must follow the pattern summarized by the following formula:

```
http://{1-4}.base.maps.api.here.com/maptile/2.1/{type}/{map id}/{scheme}/
{zoom}/{column}/{row}/{size}/{format}?
app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
&{param}={value}
```

In this formula, the first part is the base URL, followed by the URL path, API version and then the parameters as detailed in the table below.

**Note:** Use the the Aerial Tiles base URL for all satellite, terrain and hybrid schemes and the Map Tiles base URL for any other requests.

The following table documents the basic parameters that can be used in a map tile request.

**Table 3: Map tile request parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>app_code</td>
<td>String (required)</td>
<td>A URL-safe encoded string obtained during the application registration.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>See Acquiring Credentials on page 12 for information on getting credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> Usage of token is accepted although is deprecated</td>
</tr>
<tr>
<td>app_id</td>
<td>String (required)</td>
<td>A URL-safe encoded string obtained during the application registration.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>See Acquiring Credentials on page 12 for information on getting credentials.</td>
</tr>
<tr>
<td>callback_func</td>
<td>String (optional)</td>
<td>A function name that is wrapped around the JSON output. This technique allows for a more secure way of parsing the output with JavaScript;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(e.g., callback_func=test)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Can return: test( '{ ... }' );</td>
</tr>
<tr>
<td>col</td>
<td>Integer (required)</td>
<td>col – can be any number between 0 and number of columns – 1, both inclusive.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The number of tiles per column is a function of the zoom: number of columns = 2^{zoom}</td>
</tr>
<tr>
<td>congestion</td>
<td>Boolean (optional)</td>
<td>Flag that will enable congestion and environmental zones display, if available for the requested tile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>It is available for the following list of tiles</td>
</tr>
<tr>
<td>Parameter</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• maptile</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• traffictile</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• rctile</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• trucktile</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• truckonlytile</td>
</tr>
</tbody>
</table>

**format**  
String (required)  
Returned image format. The following image formats are supported:

- **png** – PNG format, 24 bit, RGB  
- **png8** – PNG format, 8 bit, indexed color  
- **jpg** – JPG format at 90% quality  

Please note that JPG is recommended to be used for satellite and hybrid schemes only.

**lg**  
String (optional)  
The MARC three-letter language code for requesting a map tile rendered in a certain language. If the given language is not available, the default language **eng** is used.

The following languages are supported for the international version:

- **ara** – Arabic  
- **chi** – Chinese (simplified)  
- **cht** – Chinese (traditional)  
- **dut** – Dutch  
- **eng** – English  
- **fre** – French  
- **ger** – German  
- **gle** – Gaelic  
- **gre** – Greek  
- **heb** – Hebrew  
- **hin** – Hindi  
- **ind** – Indonesian  
- **ita** – Italian  
- **per** – Persian  
- **pol** – Polish  
- **por** – Portuguese  
- **rus** – Russian  
- **sin** – Sinhalese  
- **spa** – Spanish  
- **tha** – Thai  
- **tur** – Turkish  
- **ukr** – Ukrainian  
- **urd** – Urdu  
- **vie** – Vietnamese  

The following languages are supported for the Chinese version:

- **chi** – Chinese (simplified)
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cht – Chinese (traditional)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>eng – English</td>
<td></td>
</tr>
<tr>
<td>map id</td>
<td>String (required)</td>
<td>Specifies the map version. One of:</td>
</tr>
<tr>
<td></td>
<td>version id &amp; – the map version of the requested map tile resource; request the latest version by using newest as map id.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hash &amp; – A unique key, unambiguously representing a certain map version.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>See Requesting the Right Map Version on page 17 for more information on map versions.</td>
<td></td>
</tr>
<tr>
<td>metadata</td>
<td>String (optional)</td>
<td>Type of metadata to generate, valid values are:</td>
</tr>
<tr>
<td></td>
<td>metatile – both the requested image and the associated metadata are returned; output=base64 is automatically on</td>
<td></td>
</tr>
<tr>
<td>output</td>
<td>String (optional)</td>
<td>Can only have the value base64 which indicates that the map tile is returned as base64 encoded text instead of as a stand alone image.</td>
</tr>
<tr>
<td>pois</td>
<td>Boolean (optional)</td>
<td>If present, pois are shown from zoom level greater or equal to 15, if not present, regular tiles are shown.</td>
</tr>
<tr>
<td>ppi</td>
<td>Integer (optional)</td>
<td>Pixels per inch. Resolution that can be requested, valid values are between 72 and 600, both included. Standard values are:</td>
</tr>
<tr>
<td></td>
<td>72 – normal, used by default if no value provided</td>
<td></td>
</tr>
<tr>
<td></td>
<td>250 – mobile</td>
<td></td>
</tr>
<tr>
<td></td>
<td>320 – hi-res</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Note: This parameter is not accepted for carnava scheme, for mobile schemes only mobile ppi value can be used, for hi-res schemes only hi-res ppi value can be used. For normal.day.custom only 72 is accepted.</td>
<td></td>
</tr>
<tr>
<td>range</td>
<td>String (optional)</td>
<td>Only relevant if output=base64 is also specified as a parameter. range must be one of 2x2, 3x3 or 4x4, which indicates the size of the array of tiles returned. The tile specified by col and row is the top right of the array, and must divide by the array size.</td>
</tr>
<tr>
<td>row</td>
<td>Integer (required)</td>
<td>row – can be any number between 0 and number of rows - 1, both inclusive. The number of tiles per row is a function of the zoom: number of rows = 2^zoom.</td>
</tr>
<tr>
<td>scheme</td>
<td>String (required)</td>
<td>Specifies the view scheme. A complete list of the supported schemes may be obtained by using the Info on page 50. The following schemes are available:</td>
</tr>
<tr>
<td>Parameter</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>normal.day</td>
<td>string</td>
<td>for a normal map view in daylight mode</td>
</tr>
<tr>
<td>normal.day.custom</td>
<td>string</td>
<td>for a normal map view in daylight mode</td>
</tr>
<tr>
<td>normal.day.grey</td>
<td>string</td>
<td>for a color-reduced map view in daylight mode (especially used for background maps)</td>
</tr>
<tr>
<td>normal.day.grey.mobile</td>
<td>string</td>
<td>mobile version of the scheme</td>
</tr>
<tr>
<td>normal.day.mobile</td>
<td>string</td>
<td>mobile version of the scheme</td>
</tr>
<tr>
<td>normal.day.transit</td>
<td>string</td>
<td>for a color-reduced map view with public transport scheme in daylight mode</td>
</tr>
<tr>
<td>normal.day.transit.mobile</td>
<td>string</td>
<td>mobile version of the scheme</td>
</tr>
<tr>
<td>normal.night</td>
<td>string</td>
<td>for a normal map view in night mode</td>
</tr>
<tr>
<td>normal.night.mobile</td>
<td>string</td>
<td>mobile version of the scheme</td>
</tr>
<tr>
<td>normal.night.grey</td>
<td>string</td>
<td>for a color-reduced map view in night mode (especially used for background maps)</td>
</tr>
<tr>
<td>normal.night.grey.mobile</td>
<td>string</td>
<td>mobile version of the scheme</td>
</tr>
<tr>
<td>carnav.day.grey</td>
<td>string</td>
<td>for a normal map view in daylight mode for car navigation</td>
</tr>
<tr>
<td>hybrid.day</td>
<td>string</td>
<td>for a satellite map view with streets in daylight mode</td>
</tr>
<tr>
<td>hybrid.day.mobile</td>
<td>string</td>
<td>mobile version of the scheme</td>
</tr>
<tr>
<td>pedestrian.day</td>
<td>string</td>
<td>pedestrian map view in daylight mode for mobile usage</td>
</tr>
<tr>
<td>pedestrian.night</td>
<td>string</td>
<td>pedestrian map view in night mode for mobile usage</td>
</tr>
<tr>
<td>satellite.day</td>
<td>string</td>
<td>for a satellite map view in daylight mode</td>
</tr>
<tr>
<td>terrain.day</td>
<td>string</td>
<td>for a terrain map view in daylight mode</td>
</tr>
<tr>
<td>terrain.day.mobile</td>
<td>string</td>
<td>mobile version of the scheme</td>
</tr>
</tbody>
</table>

**Note:** For all satellite, hybrid and terrain schemes, the *Aerial Tiles* base URL must be used instead of the normal one. Invalid combinations are rejected.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>size</td>
<td>Integer (required)</td>
<td>Returned image size. The following sizes ([width, height]) are supported:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>256 = [256, 256]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>512 = [512, 512]</td>
</tr>
</tbody>
</table>

The following sizes ([width, height]) are deprecated, although usage is still accepted:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>128 = [128, 128]</td>
</tr>
</tbody>
</table>

**Note:** Raster coverage tiles (rctile, rconlytile and rcdistonlytile) only support 256 sizes.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>style</td>
<td>String (optional)</td>
<td>If present, selects the style to use to render the tile. The possible values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fleet</td>
</tr>
<tr>
<td>style</td>
<td></td>
<td><strong>Note:</strong> Only a subset of the schemes are available for the styles, it are check in runtime. If the scheme is not available for the given style a 400 Error are generated.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> Fleet only supports normal.day, hybrid.day and terrain.day.</td>
</tr>
<tr>
<td>type</td>
<td>String (required)</td>
<td>Must have one of the following values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>maptile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>traffictile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>flowtile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>basetile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>labeltile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rctile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rconlytile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rcdistonlytile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>trucktile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>truckonlytile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>traffictile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>flowtile/</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> A traffic tile request (traffic or flow) should always use the Traffic Tiles base URL, and the URL variable newest to indicate map version. Although a hash value is accepted, it should not be used as the life span of a traffic tile is too short. Invalid base URL and/or paramater combinations are rejected.</td>
</tr>
<tr>
<td>zoom</td>
<td>Integer (required)</td>
<td>Zoom level of the map image. Minimum and maximum zoom levels are specified in the Info on page 50.</td>
</tr>
</tbody>
</table>

### Example Aerial Tile

This request example retrieves a satellite view map tile from the most recent (newest) map.

http://2.aerial.maps.cit.api.here.com/maptile/2.1/maptile/newest/satellite.day/11/1202/843/256/png8?app_id=DemoAppId01082013GAL&app_code=AJKnXv84fjrb0KIHawS0Tg
Example Traffic Tile

This request example retrieves a traffic tile from the most recent (newest) map using default traffic profile.

http://4.traffic.maps.cit.api.here.com/maptile/2.1/traffictile/newest/normal.day/18/140837/85990/256/png8
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg

Info

info is a resource that retrieves information about the map rendering service available on the server (metadata about the service), including the available map versions, map tile formats, resolutions, map view schemes, supported map languages, etc.

The following example shows a metadata request addressing this resource.

http://1.base.maps.cit.api.here.com/maptile/2.1/info
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
&{param}={value}

Metadata requests retrieve information specific to the base URLs they target. This information cannot be used in subsequent requests that use other base URLs. For example, to obtain a traffic map version, use metadata from the response to a request addressed to the Traffic Tiles base URL, but for a version of aerial map, use metadata from the response to a request addressed to Aerial Tiles base URL.

A metadata request includes a number of mandatory and optional parameters. The table below provides the details.

Table 4: Metadata request parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>app_code</td>
<td>String (required)</td>
<td>A URL-safe encoded string obtained during the application registration. See Acquiring Credentials on page 12 for information on getting credentials.</td>
</tr>
<tr>
<td>app_id</td>
<td>String (required)</td>
<td>A URL-safe encoded string obtained during the application registration. See Acquiring Credentials on page 12 for information on getting credentials.</td>
</tr>
</tbody>
</table>

Note: Usage of token is accepted although it is deprecated.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>callback_func</td>
<td>String (optional)</td>
<td>A function name that is wrapped around the JSON output. This technique allows for a more secure way of parsing the output with JavaScript;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(e.g., <code>callback_func=test</code>)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Can return: <code>test( '{ ... }' );</code></td>
</tr>
<tr>
<td>callback_id</td>
<td>String (optional)</td>
<td>If specified (associated with callback_func), the function wrapper contains this id as first parameter. This allows the user to check if the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>response is actually the response to the request, and not some maliciously injected code; for example <code>callback_id=1425</code> returns: <code>test( '1425', '{ ...}' );</code></td>
</tr>
<tr>
<td>hash_for</td>
<td>String (optional)</td>
<td>Used to obtain the information about a specific map version. The parameter is only valid if output is set to header. The parameter is designed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>to offer automated caches a simple way to relate map version strings to their current hash. The valid values of the parameter are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;newest&quot; – indicates the most recent map version</td>
</tr>
<tr>
<td></td>
<td></td>
<td>a string containing the map version ID – indicates that the hash for this specific version is to be retrieved</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the parameter is used, the response header contains a key, <code>maphash</code>, with a hash value identifying the specified map version.</td>
</tr>
<tr>
<td>output</td>
<td>String (optional)</td>
<td>Output specifier. The API supports the following output specifier values (and output formats):</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>xml</code> – delivers the API metadata in XML format according to the XSD (this is the default).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>json</code> – delivers the API metadata in JSON format; for JSONP requests see parameter callback_func/callback_id</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>xsd</code> – delivers the specific XML schema (XSD) of the API metadata to process and to verify the Service Meta Data Response.</td>
</tr>
</tbody>
</table>

**Metadata Request Response**

The following example shows a request for metadata to be delivered in XML format. An alternative output format is JSON. Both output formats have the same structure and type names.

```
http://{1-4}.base.maps.api.here.com/maptile/2.1/info
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```
The XML response to the above request contains the following data:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <maps>
    <map region="all" newest="true" id="8.30.53.106" hash="c376276c01" isbn="" />
  </maps>
  <resolutions>
    <resolution id="256" height="256" width="256" />
    <resolution id="128" height="128" width="128" />
  </resolutions>
  <formats>
    <format encoding="png" bbp="24" id="png" />
    <format encoding="png8" bbp="8" id="png8" />
    <format encoding="jpg" bbp="24" id="jpg" />
  </formats>
  <schemes>
    <scheme id="carnav.day.grey" />
    <scheme id="normal.day.grey" />
    <scheme id="normal.day.transit" />
    <scheme id="normal.day" />
    <scheme id="terrain.day" />
    <scheme id="satellite.day" />
    <scheme id="hybrid.day" />
    <scheme id="hybrid.day.mobile" />
    <scheme id="normal.day.grey.mobile" />
    <scheme id="normal.day.mobile" />
    <scheme id="normal.day.transit.mobile" />
    <scheme id="terrain.day.mobile" />
    <scheme id="normal.day.custom" />
    <scheme id="pedestrian.day" />
    <scheme id="pedestrian.night" />
  </schemes>
  <tiletypes>
    <tiletype id="maptile" />
    <tiletype id="basetile" />
    <tiletype id="xbasetile" />
    <tiletype id="labeltile" />
  </tiletypes>
  <languages>
    <language id="ARA" />
    <language id="CHI" />
    <language id="CHT" />
    <language id="ENG" />
    <language id="FRE" />
    <language id="GER" />
    <language id="ITA" />
    <language id="POL" />
    <language id="RUS" />
    <language id="SPA" />
  </languages>
  <zoomLevels min="0" max="20" />
</response>
```
The table below describes the XML elements in this output.

Table 5: Elements in the output

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>map</strong></td>
<td>Available maps. A map has the following attributes:</td>
</tr>
<tr>
<td></td>
<td>id: the map version, newest always refers to the latest map version. Several latest map versions for different regions are possible.</td>
</tr>
<tr>
<td></td>
<td>region: the region of the world where the map is valid, currently only all (world) is supported.</td>
</tr>
<tr>
<td></td>
<td>hash: the unique key use to reference the specific map version in the map id parameter of a map tile request Maptile on page 44.</td>
</tr>
<tr>
<td></td>
<td>isbn: unique identifier used for validated chinese maps, it will either be empty or not present for international maps.</td>
</tr>
<tr>
<td><strong>resolution</strong></td>
<td>Supported map tile image resolutions. A resolution has the following attributes:</td>
</tr>
<tr>
<td></td>
<td>width: the width of the supported image resolution.</td>
</tr>
<tr>
<td></td>
<td>height: the height of the supported image resolution.</td>
</tr>
<tr>
<td></td>
<td>id: used to reference the resolution in the size parameter of a map tile request Maptile on page 44.</td>
</tr>
<tr>
<td><strong>format</strong></td>
<td>Supported image formats. A format has the following attributes:</td>
</tr>
<tr>
<td></td>
<td>encoding: the image encoding format, one of: png, png8 or jpg.</td>
</tr>
<tr>
<td></td>
<td>bbp: bytes per pixel, 8 bbp refers to a palette based image format, 24 bbp is a true RGB image format.</td>
</tr>
<tr>
<td></td>
<td>id: used to reference the format in the format parameter of a map tile request Maptile on page 44.</td>
</tr>
<tr>
<td><strong>scheme</strong></td>
<td>Supported map tile schemes. A scheme has the following attribute:</td>
</tr>
<tr>
<td></td>
<td>id: used to reference the scheme in the scheme parameter of a map tile request Maptile on page 44.</td>
</tr>
<tr>
<td><strong>tiletype</strong></td>
<td>Supported map tile types. A tiletype has the following attribute:</td>
</tr>
<tr>
<td></td>
<td>id: used to reference the scheme in the type parameter of a map tile request Maptile on page 44.</td>
</tr>
<tr>
<td><strong>language</strong></td>
<td>Supported languages for rendered map tiles. A language has the following attribute:</td>
</tr>
<tr>
<td></td>
<td>id: a MARC three-letter code used to reference the language in the lg parameter of a map tile request Maptile on page 44.</td>
</tr>
<tr>
<td><strong>zoomLevels</strong></td>
<td>Minimum and maximum values for the zoom parameter in a map tile request Maptile on page 44.</td>
</tr>
<tr>
<td></td>
<td>min: minimum possible map tile zoom.</td>
</tr>
<tr>
<td></td>
<td>max: maximum possible map tile zoom.</td>
</tr>
</tbody>
</table>
Copyright

The resource named `copyright` handles requests for copyright information about different map tiles.

A typical request against this resource must comply with the following formula.

```plaintext
http://{1-4}.base.maps.api.here.com/maptile/2.1/copyright/{map id}
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```

**Note:** Metadata requests retrieve information specific to the base URLs they target. This information cannot be used in subsequent requests that use other base URLs. For example, to obtain a traffic map version, use metadata from the response to a request addressed to the `Traffic Tiles` base URL, but for a version of aerial map, use metadata from the response to a request addressed to `Aerial Tiles` base URL.

In both output formats (XML and JSON), the copyright information associated with the different view schemes, zoom level ranges, and geographical areas is shown. Variant copyrights are defined by the bounding boxes and the relevant zoom level range. The default copyright information of the view scheme has no restriction elements and is therefore valid for all remaining areas and zoom levels.

The table below describes the parameters that can be used to formulate a copyright information request.

**Table 6: Copyright information request parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>app_code</td>
<td>String (required)</td>
<td>A URL-safe encoded string obtained during the application registration. See Acquiring Credentials on page 12 for information on getting credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> Usage of token is accepted although is deprecated</td>
</tr>
<tr>
<td>app_id</td>
<td>String (required)</td>
<td>A URL-safe encoded string obtained during the application registration. See Acquiring Credentials on page 12 for information on getting credentials.</td>
</tr>
<tr>
<td>callback_func</td>
<td>String (optional)</td>
<td>A function name that is wrapped around the JSON output. This technique allows for a more secure way of parsing the output with JavaScript; (e.g., callback_func=test) Can return: test( '{ ... }' );</td>
</tr>
</tbody>
</table>
### API Reference

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>callback_id</td>
<td>String (optional)</td>
<td>If specified (associated with callback_func), the function wrapper contains this id as first parameter. This allows the user to check if the response is actually the response to the request, and not some maliciously injected code; for example:&lt;br&gt;<code>javascript&lt;br&gt;callback_id=1425&lt;br&gt;returns:&lt;br&gt;test('1425', '{ ... }');&lt;br&gt;</code></td>
</tr>
<tr>
<td>map id</td>
<td>String (required)</td>
<td>Specifies the map version. One of:&lt;br&gt;• <strong>version id &amp;</strong> – the map version of the requested map tile resource; request the latest version by using <code>newest</code> as <code>map id</code>.&lt;br&gt;• <strong>hash &amp;</strong> – A unique key, unambiguously representing a certain map version.&lt;br&gt;See <a href="#">Requesting the Right Map Version</a> on page 17 for more information on map versions.</td>
</tr>
<tr>
<td>output</td>
<td>String (optional)</td>
<td>Output specifier. The API supports the following output specifier values (and output formats):&lt;br&gt;• <strong>xml</strong> – delivers the API metadata in XML format according to the XSD (this is the default).&lt;br&gt;• <strong>json</strong> – delivers the API metadata in JSON format; for JSONP requests see parameter callback_func/callback_id&lt;br&gt;• <strong>xsd</strong> – delivers the specific XML schema (XSD) of the API metadata to process and to verify the Service Meta Data Response.</td>
</tr>
</tbody>
</table>

### Example Copyright Request in XML

The following example requests copyright information in XML:

```xml
http://1.base.maps.cit.api.here.com/maptile/2.1/copyright/newest<br>?app_id=DemoAppId01082013GAL<br>&app_code=AJKnXv84fjrb0KIHawS0Tg<br>&output=xml
```

The following code block shows the XML response.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<copyrights xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="nokia:maptiler:copyright:1.0" xsi:schemaLocation="nokia:maptiler:copyright:1.0 CopyrightInfo.xsd">
  <providers>
    <provider name="NavInfo">
      <text>2010 NavInfo</text>
      <full>copyright 2010 NavInfo Co Ltd. All Rights Reserved</full>
      <link>http://www.nav2.com.cn/</link>
    </provider>
    <provider name="THTC">
```

---

**Map Tile API Developer’s Guide**

---

**Example Copyright Request in XML**

The following example requests copyright information in XML:

```xml
http://1.base.maps.cit.api.here.com/maptile/2.1/copyright/newest<br>?app_id=DemoAppId01082013GAL<br>&app_code=AJKnXv84fjrb0KIHawS0Tg<br>&output=xml
```

The following code block shows the XML response.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<copyrights xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="nokia:maptiler:copyright:1.0" xsi:schemaLocation="nokia:maptiler:copyright:1.0 CopyrightInfo.xsd">
  <providers>
    <provider name="NavInfo">
      <text>2010 NavInfo</text>
      <full>copyright 2010 NavInfo Co Ltd. All Rights Reserved</full>
      <link>http://www.nav2.com.cn/</link>
    </provider>
    <provider name="THTC">
```
<area>
<areas>
<schemes>
<scheme name="carnav.day.grey">
<copyright zoomMin="5" zoomMax="20" area="China">NavInfo</copyright>
<copyright zoomMin="6" zoomMax="20" area="Iran">THTC</copyright>
</scheme>
<scheme name="normal.day.grey">
<copyright zoomMin="5" zoomMax="20" area="China">NavInfo</copyright>
<copyright zoomMin="6" zoomMax="20" area="Iran">THTC</copyright>
</scheme>
<scheme name="normal.day.transit">
<copyright zoomMin="5" zoomMax="20" area="China">NavInfo</copyright>
<copyright zoomMin="6" zoomMax="20" area="Iran">THTC</copyright>
</scheme>
<scheme name="normal.day">
<copyright zoomMin="5" zoomMax="20" area="China">NavInfo</copyright>
<copyright zoomMin="6" zoomMax="20" area="Iran">THTC</copyright>
</scheme>
<scheme name="terrain.day">
<copyright zoomMin="5" zoomMax="20" area="China">NavInfo CIAT</copyright>
<copyright zoomMin="6" zoomMax="20" area="Iran">THTC CIAT</copyright>
<copyright>CIAT</copyright>
</scheme>
<scheme name="satellite.day">
<copyright>DigitalGlobe</copyright>
</scheme>
<scheme name="hybrid.day">
<copyright zoomMin="5" zoomMax="20" area="China">NavInfo DigitalGlobe</copyright>
<copyright zoomMin="6" zoomMax="20" area="Iran">THTC DigitalGlobe</copyright>
</scheme>
<scheme name="hybrid.day.mobile">
<copyright zoomMin="5" zoomMax="20" area="China">NavInfo DigitalGlobe</copyright>
<copyright zoomMin="6" zoomMax="20" area="Iran">THTC DigitalGlobe</copyright>
</scheme>
</schemes>
</area>
This XML response contains the following elements:

Table 7: Response elements

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>provider</td>
<td>The map provider name with the following description elements:</td>
</tr>
<tr>
<td></td>
<td>text: copyright text to display after the copyright symbol on the map.</td>
</tr>
<tr>
<td></td>
<td>full: verbose copyright text of the label to display by mouse over label or info menu entry.</td>
</tr>
<tr>
<td></td>
<td>link: link to the map provider website.</td>
</tr>
<tr>
<td></td>
<td>icon: logo of the map provider that can be optionally added to the map. Specified by the URL (src), dimensions (width, height) and the mimetype.</td>
</tr>
<tr>
<td>area</td>
<td>The name, size and shape of a geographical area with a particular copyright. The area is specified by geoRect. The bounding boxes (bottom - latitude, left - longitude, top - latitude, right - longitude).</td>
</tr>
<tr>
<td>scheme</td>
<td>The view schemes and corresponding copyrights are defined here.</td>
</tr>
<tr>
<td></td>
<td>copyright: Map provider or list of providers for a particular view scheme. Copyrights are defined by the given geographical area and the relevant zoom level range (zoomMin, zoomMax). The default copyright is the one that has no geographical restriction and is therefore valid for all other areas.</td>
</tr>
<tr>
<td></td>
<td>All relevant copyrights of the map view must be displayed. Different areas may contribute by overriding or adding to the copyright text depending on whether the view is contained completely within one of these areas or overlaps partially one or more of them. If there are parts of several areas in the view, check the composite list of providers to avoiding duplicate entries with the different copyright data.</td>
</tr>
<tr>
<td></td>
<td>For example, using a hybrid.day view scheme, the default copyright will be shown and then all the relevant ones that are inside of the area of the bounding box currently being displayed.</td>
</tr>
</tbody>
</table>
Copyright Request Response in JSON

In the example below, the request for copyright information does not indicate the output format.

http://1.base.maps.cit.api.here.com/maptile/2.1/copyright/newest
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg

The default output format is JSON and this is the format delivered by the response to the above request:

```json
{
  "satellite": [
    {
      "minLevel": 0,
      "maxLevel": 20,
      "label": "2010 DigitalGlobe",
      "alt": "copyright2010DigitalGlobe Inc."
    }
  ],
  "normal": [
    {
      "minLevel": 0,
      "maxLevel": 20,
      "label": "2010NAVTEQ",
      "alt": "copyright2010NAVTEQAllRightsReserved"
    },
    {
      "minLevel": 5,
      "maxLevel": 20,
      "label": "2010NavInfo",
      "alt": "copyright2010NavInfoCoLtd.AllRightsReserved",
      "boxes": [
        [29.1171, 41.5657, 124.1865],
        [41.5630, 47.1087, 90.6305],
        ...
      ]
    },
    {
      "minLevel": 6,
      "maxLevel": 20,
      "label": "2010THTC",
      "alt": "copyright2010IranMapsprovidedbyTHTC",
      "boxes": [
        [29.8960, 48.0337, 36.6154, 60.4761],
        [35.7922, 45.2450, 38.9187],
        ...
      ]
    }
  ]
}
```
53.8126
},
...
]
]

"hybrid": [
[
"minLevel": 0,
"maxLevel": 20,
"label": "2010NAVTEQ&DigitalGlobe",
"alt": "copyright2010NAVTEQ&DigitalGlobeAllRightsReserved"
],
[
"minLevel": 5,
"maxLevel": 20,
"label": "2010NavInfo&DigitalGlobe",
"alt": "copyright2010NavInfoCoLdt.&DigitalGlobeAllRightsReserved",
"boxes": [
[29.1171, 84.2005, 41.5657, 124.1865]
]
],
[
"minLevel": 6,
"maxLevel": 20,
"label": "2010THTC&DigitalGlobe",
"alt": "copyright2010IranMapsprovidedbyTHTC&DigitalGlobe",
"boxes": [
[29.8960, 48.0337, 36.6154, 60.4761]
]
],
]

"terrain": [
[
"minLevel": 0,
"maxLevel": 20,
"label": "2010NAVTEQ&SRTMV4, 2008, CIAT",
"alt": "copyright2010NAVTEQAllRightsReserved&SRTMV4, copyright2008, CIAT"
],
[
"minLevel": 5,
"maxLevel": 20,
"label": "2010NavInfo&SRTMV4, 2008, CIAT",
"alt": "copyright2010NavInfoAllRightsReserved&SRTMV4, copyright2008, CIAT",
"boxes": [
[29.1171, 84.2005, 41.5657, 124.1865]
[41.563, 85.7494, 47.1087, 90.6305]
]
The copyright information is defined for each view scheme and contains the following elements:

**Table 8: Copyright information elements (response)**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>minLevel</td>
<td>Minimum zoom level for the specified copyright label.</td>
</tr>
<tr>
<td>maxLevel</td>
<td>Maximum zoom level for the specified copyright label.</td>
</tr>
<tr>
<td>label</td>
<td>Copyright text to display after the copyright symbol on the map.</td>
</tr>
<tr>
<td>alt</td>
<td>Verbose copyright text of the label to display by mouse over label or info menu entry.</td>
</tr>
<tr>
<td>boxes</td>
<td>The bounding boxes define areas where specific copyrights are valid. A bounding box is defined by <code>bottom (latitude), left (longitude)</code> and <code>top (latitude), right (longitude)</code>. The default copyright has no <code>boxes</code> element and covers all other areas. The bounding boxes may override or add to the copyright label depending on whether the map view is contained completely within one of the bounding boxes or overlaps partially one or more of them.</td>
</tr>
</tbody>
</table>

**Version**

This version resource handles requests for software version details.

The following URL shows the format of a version request and the table below explains each parameter.

http://{1-4}.base.maps.api.here.com/maptile/2.1/version?app_id=DemoAppId01082013GAL
Map Tile API supports the standard **HTTP status codes** detailed in the table below.

**Note:** Metadata requests retrieve information specific to the *base URLs* they target. This information cannot be used in subsequent requests that use other base URLs. For example, to obtain a traffic map version, use metadata from the response to a request addressed to the *Traffic Tiles* base URL, but for a version of aerial map, use metadata from the response to a request addressed to *Aerial Tiles* base URL.

### Table 9: Version request parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>app_code</td>
<td>String (required)</td>
<td>A URL-safe encoded string obtained during the application registration. See <em>Acquiring Credentials</em> on page 12 for information on getting credentials. <strong>Note:</strong> Usage of <em>token</em> is accepted although is deprecated</td>
</tr>
<tr>
<td>app_id</td>
<td>String (required)</td>
<td>A URL-safe encoded string obtained during the application registration. See <em>Acquiring Credentials</em> on page 12 for information on getting credentials</td>
</tr>
</tbody>
</table>

**Example**

The following example shows a request addressed to the *version* resource.

```
http://(1-4).base.maps.api.here.com/maptile/2.1/version
?app_id=DemoAppId01082013GAL
&app_code=AJKnXv84fjrb0KIHawS0Tg
```

The response contains information as shown in this example.

```
MRS: 2.1.32.1
MOS: 2.0.1.3448383637
```

Note that the first element is the version of the Map Tile API, while the second is the version of the underlying map rendering system.
<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200 OK</td>
<td>Indicates success, but may also be returned when an invalid resource name and/or an invalid parameter combination has been used in the request.</td>
</tr>
<tr>
<td>400 Bad request</td>
<td>Indicates an invalid parameter value in the request, for example zoom out of range.</td>
</tr>
<tr>
<td>401 Unauthorized</td>
<td>Indicates authentication failure.</td>
</tr>
<tr>
<td>403 Forbidden</td>
<td>Indicates incorrect app_code or app_id in the request. See Acquiring Credentials on page 12 for more information.</td>
</tr>
<tr>
<td>404 Not found</td>
<td>Indicates that the resource was not found.</td>
</tr>
<tr>
<td>500 Internal error</td>
<td>Indicates a server configuration issue.</td>
</tr>
<tr>
<td>503 Service Unavailable</td>
<td>Indicates that the service is temporarily unavailable due to system overload or maintenance.</td>
</tr>
</tbody>
</table>