A Descriptive Model for Predicting Popular Areas in a Web Map

RICARDO GARCÍA, JUAN PABLO DE CASTRO, MARÍA JESÚS VERDÚ
ELENA VERDÚ, LUISA MARÍA REGUERAS and PABLO LÓPEZ
University of Valladolid
Higher Technical School of Telecommunications Engineering
Department of Signal Theory, Communications and Telematics Engineering
Campus Miguel Delibes, Paseo Belén 15, 47011 Valladolid
SPAIN
{ricgar, juacas, marver, elever, luireg}@tel.uva.es, plopec@ribera.tel.uva.es

Abstract: The increasing popularity of web map services has motivated the development of more scalable services in the Spatial Data Infrastructures (SDI). Tiled map services have emerged as an scalable alternative to traditional map services. Instead of rendering image maps on the fly, a collection of pre-generated image tiles can be retrieved very fast from a server-side cache. However, storage requirements and start-up time for generating all tiles are often prohibitive for many potentially providers when the cartography covers large areas for multiple rendering scales, which forces to use partial caches containing a subset of the total tiles. This work proposes a descriptive model based on the mining of real-world logs from several nationwide public web map services in Spain. The proposed model is able to determine in advance which areas are likely to be requested in the future based exclusively on past accesses. Tiles that are anticipated to be requested soon can be pre-generated and cached for faster retrieval. As the number of tiles grows exponentially with the rendering resolution level, it is rarely feasible to work with statistics of individual tiles. To overcome this issue, a simplified model is proposed which combines statistics from multiple tiles to reduce the dimension of the tiling space. Simulations demonstrate that significant savings of storage requirements can be achieved by using a partial cache with the proposed model, while maintaining a high cache hit ratio.

Key–Words: Web mapping, Map tile, WMTS, SDI, WMS, Descriptive model, Logs, Proxy cache

1 Introduction

The diverse Web Map Service (WMS) specifications [1] of the Open Geospatial Consortium (OGC) focuses on flexibility, enabling clients to obtain exactly the final image they want. However, spatial parameters in map requests are not restricted, which forces map images to be generated on the fly through an expensive process that implies access to the data store, style application, layer composition and compression of the final image.

This process has been proved to be ineffective to satisfy the requirements of some massive applications, as explained in [2] after NASA’s experience with the OnEarth map server. For this reason, most popular commercial services, like Google Maps or Microsoft Bing Maps, have adopted their own specifications where the geographic space is tiled according to a predefined grid, and content is usually pre-generated [3].

This situation has also motivated the development of tile-based recommendations like the Web Map Service - Cached (WMS-C) specification from OSGeo [4] or the new Web Map Tile Service (WMTS) of the OGC[5].

When an OGC service is used in a demanding environment, with stationary and bit configurable parameters, the proxy web cache pattern can be used to improve the perceived quality of service. A proxy is a device placed seamlessly anywhere between the client and the final service, intercepting user’s requests [6].

Using this approach, providers have to cope with serious decisions about the design and maintenance of their tile caches. The immediate option is to pre-generate all tiles from all available scales. However, only big corporations have currently enough storage resources to store all the tiles. This is not a problem for them, but smaller companies must decide carefully which part of the content should be pre-generated.

Anyway, there are cartographic layers that should...
be updated frequently and every of them start from an empty cache. In general, to pre-generate all the objects is not a good approach when serving frequently updated maps, such as weather or traffic information maps.

The rest of the paper is organized as follows. Section 2 introduces a formal characterization of the tiling space. Section 3 defines a descriptive model to predict which tiles should be cached from logs of past server accesses. This model is analyzed and the experiment results are presented later. Finally, Section 4 includes the main conclusions of this work.

2 Tiling space

In order to offer a tiled web map service, the web map server renders the map across a fixed set of scales through progressive generalization. Rendered map images are then divided into tiles, describing a tile pyramid.

At any moment a cache status can be defined where its managed objects can be available with a certain probability.

Spatial cache objects can be identified by their coordinates. Each tile is defined as \( T(i, j, n) \), having \( i, j, n \in \mathbb{N} \) (where \( n \) is the resolution level in the scale pyramid and \( i, j \) are the spatial indexes for this level). Therefore, \( P_h \{ T(i, j, n) \} \) is defined as the probability of getting a cache hit for the requested tile \( T(i, j, n) \), at time \( t \). Denominate \( \tau_h \) to the cost in seconds to deliver an object directly from the cache and \( \tau_m \) to the cost in seconds to build a tile through the original services.

Being \( f_{req}(x, y, n) \) the spatial probability density that characterizes the spatial distribution of the centroids of the requested map tiles for the scale \( n \) at time \( t \), in general, for any request distribution, tiles are requested with probability (1).

\[
P_{req} \{ T(i, j, n) \} = \int_{x=0}^{x_{n,i+1}} \int_{y=0}^{y_{n,j+1}} f_{req}(x, y, n) \, dx \, dy
\]  

Although this result can be useful to analyze non tiled requests to a proxy-cache, it can be simplified assuming that requests are constrained to a reference grid cell (Figure 2). In this case, the probability of receiving a request of the tile \( T(i, j, n) \) with size \( \Delta x \Delta y \) is (2):

\[
P_{req} \{ T(i, j, n), t \} = f_{req}(x, y, n, t) \, \Delta x \Delta y
\]  

The latency to serve a request for a given tile with coordinates \((i, j, n)\) at time \( t \) can be determined by (3):

\[
\tau(i, j, n, t) = P_h \{ T(i, j, n) \}(t) \tau_h + (1 - P_h \{ T(i, j, n) \}(t) \tau_m)
\]  

Combining (3) and (1) a probabilistic expression can be obtained to estimate the average latency of the service:

\[
\tau(t) = \sum_{(i,j,n)} (\tau_m - P_h \{ T(i, j, n) \}(t) (\tau_m - \tau_h))
\]

Some components can be identified in (4), which should be parametrized at least locally.

Some important characteristics of this model can be:

- There is no independence between \( P_h \{ T(i, j, n) \}(t) \) and \( P_{req} \{ T(i, j, n), t \} \) because the cache status is intimately connected to the service requests history.
- There is no temporal invariance during start-up of transient states.
- The probability density function is not uniform and it probably represents a direct relationship with the spatial structure of the underlying information.
Another factor that must be addressed is the problem of managing exponential data structures. In a cache with pyramidal scales, the number of objects increases exponentially according to the \( n \) value. Therefore, the use of analytic or predictive algorithms can be impractical, even with the support of heuristic algorithms, if they aim to be used with all the pyramid levels.

For this reason, it is very useful to obtain a statistical relationship model between different scale levels. The statistics of a level can be extrapolated to other near levels. Assuming that the geographic location is a relevant information for all the scale levels, and that requests have significant spatial correlation, heuristic algorithms like Locality Principle [7] can be used. These algorithms should manage the whole cache through statistical probes within levels containing a manageable number of objects (tiles).

![Figure 2: Probability estimation of requests to tile \( T(i,j,n) \) calculated from lower levels](image)

3 Descriptive model

Descriptive models determine the most requested map areas from map servers logs. For example, the web application Microsoft HotMap\(^1\) represents in a heatmap the requests to Bing Maps service [8, 9, 10]. However, it is not possible to access data itself, which limits the possible analysis to a visual exploration and makes the use of automatic algorithms to extract patterns of interest difficult.

3.1 Analyzed map services

This study has been carried out with data retrieved from requests made to the WMS-C services Cartociudad\(^2\), IDEE-BASE and PNOA, provided by the National Geographic Institute (IGN)\(^3\) of Spain.

Tiled versions of these services use caches implemented by Metacarta Tilecache [11]. This cache system follows the OSGeo WMS-C specification.

3.2 Retrieving request data

Tile map requests are extracted from the Apache’s standard access log configured using the Common Log Format [12]. Information retrieved from these records is as follows:

- Request date, with precision in seconds.
- IP address or hostname of the remote client that made the request to the server.
- Server status code returned to the client. This information is very valuable, because it reveals whether the request was successfully returned or not.
- Size of the returned object. This value does not include the response headers, and it is expressed in bytes.

The following WMS-C request parameters are extracted: service, version, request, layers, width, height, format, styles, exceptions and coordinate reference system.

Figure 3 plots the number of requests made versus the resolution level for the analyzed services. IDEE-BASE and Cartociudad present an anomalous peak in level 4, which can be justified because it is the more suitable level for displaying the whole Spain area in a single map screen.

![Figure 3: Normalized request distribution along the different scales](image)
tion tiles can be included in the cache later, or even let them be included in the cache as they are requested.

Main part of the analyzed requests to these services are referred to the Spain. So, the studied area has been reduced to the bounding box [-11.9971, 32.8711, 5.5371, 46.0107], which represents Spain in a grid of 400 tiles width and 300 tiles height in level 12.

3.3 Simplified model
Given the exponential nature of the scale pyramid, and the impossibility of working with statistics from individual tiles, a simplified model has been proposed. This model tries to approximate the probability of receiving a tile request in a particular level from the statistics retrieved from another level covering the same area. Specifically the model has been simplified to the 400x300 tiles area defined above. The pyramidal structure of scales is transformed in some way in a prism-like structure with the same number of items in all the scales (see Figure 5).

3.4 Experiment and results
In order to experiment with the proposed model, request logs were divided in two time ranges. The first one was used as source to make predictions and the second one for proving the prediction created previously. The experiment was conducted with the simplified model to the grid cell defined by the level of resolution 12.

Figure 6 shows the heatmap for the request logs following the proposed model. These figures demonstrate that some entities such as coast lines, cities and major roads are highly requested. These elements could be used as entities for a predictive model to identify priority objects, as explained in [13].

Table 1, 2 and 3 represent the hit percentage for each service. These tables show the percentage of hits obtained for the level identified by the column index from the statistics collected in the level identified by the row index. Last column shows the resources consumption, as a percentage of cached tiles. Shadowed cell in Table 1 indicates that using retrieved statistics of level 13 as the prediction source, a hit rate of 91.95% is obtained for predictions made in the level 12.
18, being necessary the storage of a 8.83% of the tiles in cache.

<table>
<thead>
<tr>
<th>scale</th>
<th>prop</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>87.7</td>
<td>90.4</td>
<td>91.5</td>
<td>90.2</td>
<td>94.4</td>
<td>95.4</td>
<td>96.0</td>
<td>92.3</td>
<td>18.08</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>69.6</td>
<td>78.8</td>
<td>82.3</td>
<td>83.0</td>
<td>88.6</td>
<td>80.8</td>
<td>91.9</td>
<td>85.0</td>
<td>8.83</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>56.6</td>
<td>69.8</td>
<td>76.4</td>
<td>78.8</td>
<td>84.6</td>
<td>78.8</td>
<td>89.0</td>
<td>80.2</td>
<td>5.40</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>41.4</td>
<td>54.9</td>
<td>65.8</td>
<td>74.5</td>
<td>78.4</td>
<td>74.5</td>
<td>85.5</td>
<td>73.3</td>
<td>3.00</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>33.2</td>
<td>45.6</td>
<td>57.6</td>
<td>66.3</td>
<td>73.8</td>
<td>71.2</td>
<td>84.1</td>
<td>69.0</td>
<td>2.04</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>29.3</td>
<td>42.4</td>
<td>53.3</td>
<td>63.5</td>
<td>54.8</td>
<td>78.2</td>
<td>81.5</td>
<td>66.4</td>
<td>1.61</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>26.3</td>
<td>37.7</td>
<td>50.4</td>
<td>61.5</td>
<td>52.8</td>
<td>67.8</td>
<td>81.7</td>
<td>63.4</td>
<td>1.35</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>15.2</td>
<td>23.4</td>
<td>33.6</td>
<td>44.7</td>
<td>37.9</td>
<td>48.6</td>
<td>72.7</td>
<td>44.5</td>
<td>0.76</td>
<td></td>
</tr>
</tbody>
</table>

prop 67.7 | 76.3 | 80.6 | 84.4 | 87.4 | 90.8 | 92.7 | 83.8 | 6.68 |

Table 1: Percentage (%) of cache hits through the simplified model obtained from Cartociudad logs, using the mean of the normalized frequencies as the probability threshold.

<table>
<thead>
<tr>
<th>scale</th>
<th>prop</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>88.0</td>
<td>85.7</td>
<td>93.7</td>
<td>94.5</td>
<td>96.0</td>
<td>93.7</td>
<td>93.5</td>
<td>79.3</td>
<td>21.02</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>74.8</td>
<td>85.9</td>
<td>90.7</td>
<td>92.1</td>
<td>94.0</td>
<td>89.7</td>
<td>82.4</td>
<td>54.8</td>
<td>15.28</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>48.1</td>
<td>57.7</td>
<td>84.3</td>
<td>86.6</td>
<td>89.1</td>
<td>78.5</td>
<td>59.8</td>
<td>38.7</td>
<td>7.63</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>34.6</td>
<td>45.6</td>
<td>76.0</td>
<td>82.7</td>
<td>81.7</td>
<td>66.9</td>
<td>50.6</td>
<td>31.5</td>
<td>4.58</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>41.6</td>
<td>53.1</td>
<td>80.9</td>
<td>84.0</td>
<td>89.8</td>
<td>74.7</td>
<td>56.2</td>
<td>37.5</td>
<td>6.10</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>30.2</td>
<td>37.6</td>
<td>57.0</td>
<td>60.1</td>
<td>62.2</td>
<td>69.5</td>
<td>55.5</td>
<td>23.4</td>
<td>3.26</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>23.5</td>
<td>25.5</td>
<td>41.7</td>
<td>46.1</td>
<td>45.8</td>
<td>41.4</td>
<td>61.9</td>
<td>33.3</td>
<td>2.33</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>8.8</td>
<td>8.6</td>
<td>12.4</td>
<td>13.1</td>
<td>14.3</td>
<td>12.3</td>
<td>13.0</td>
<td>44.1</td>
<td>1.23</td>
<td></td>
</tr>
</tbody>
</table>

prop 67.1 | 78.0 | 87.7 | 90.1 | 92.3 | 86.8 | 83.9 | 71.6 | 12.50 |

Table 2: Percentage (%) of cache hits through the simplified model obtained from IDEE-BASE logs, using the mean of the normalized frequencies as the probability threshold.

Nevertheless, it must be noted that the main benefit of using a partial cache is not the reduction in the number of cached tiles. The main benefits are the savings in storage space and generation time. As explained in [13], the amount of saved tiles is bigger than the storage saving. It reveals that the most interesting tiles come at a bigger cost. Mainly, popular areas are more complex, and it is necessary more disk space to store them.

Low hit rates for the PNOA service are caused by the strange request distribution for this service. The simplified model is not able to make precise predictions for this kind of services.

Figure 7 represents the cache hit ratios obtained by the simplified model for the IDEE-BASE service. From a certain percentage of cached objects, identified by the continuous vertical line, the simplified model is not able to make predictions. Tiles situated at the right of this line correspond to objects that have never been requested and therefore have not been col-

Figure 7: Percentage of hits vs cached objects for IDEE-BASE service through the simplified model with grid cell for level 12

The simplified model obtains better results for predicting user behavior from near resolution levels. Descending in the scale pyramid, the requested objects percentage decreases, so the model prediction range decreases too.

4 Conclusion

Web map tiled services have reached high popularity in recent times, improving response times and scalability versus traditional mapping services, by serving pre-generated images from cache. In environments with reduced storage capabilities or where the cartography is updated frequently, it is not suitable to pre-generate the whole cartographic content. In this cases, it is necessary to work with incomplete caches. The solution proposed in this paper is based on the definition of priority areas for pre-fetching and replacement mechanisms, maximizing the user QoS while keeping
resource consumption under a given level. It tries to keep in cache the tiles which are likely to be requested in the future. To determine those priority areas, a descriptive model has been proposed. This model has been tested with different Spanish national map services logs. High hit rates obtained prove that it is possible to predict the future accesses to a Web map service based solely on the information collected from past. This model can be applied when user behavior is relatively stationary. The multi-scale analysis made supports the use of statistics collected in a certain level to predict the behavior at other near levels.

In the future, higher cache hit ratios could be achieved by combining the weighted information collected from different scale levels, instead the single level analysis in the current model.

From the usage logs of analyzed map services some conclusions have been obtained. Areas like coast lines, cities or major roads are more requested than others. These elements could be used as entities for a predictive model to identify priority objects that should be taken into account during the cache maintenance tasks.

Acknowledgements: This work has been partially supported by the Spanish Ministry of Science and Innovation through the project “España Virtual” (ref. CENIT 2008-1030), National Centre for Geographic Information (CNIG) and the National Geographic Institute of Spain (IGN).

References: