Towards a general evaluation procedure for Geo Web Services

Stephan Schmid, Wolfgang Reinhardt

University of the Bundeswehr Munich, Germany

Abstract. Geo Web Services (GWS) gain more and more importance today. Since the INSPIRE directive was adopted in 2007 a significant increase of available GWS can be noticed. The INSPIRE directive requires a certain quality of service (QoS). Thus quality aspects for GWS become more important. The quality of GWS needs to be defined, specified and especially determined. It is also necessary to establish a common evaluation procedure including the corresponding test procedures. In this paper we present a general evaluation procedure for GWS and we demonstrate its applicability using the most famous cartographic service – the Web Map Service (WMS) – as an example.

Keywords: WMS, OGC Geo Web Services, Quality evaluation, Quality of Service, INSPIRE services

1. Quality of Service

In the last decades a lot of effort has been yield to establish national and international Spatial Data Infrastructures (SDI) according to the INSPIRE directive. The INSPIRE directive has been entering into force in May 2007, while its aim is to establish an infrastructure for spatial information in Europe. The directive is operated by the 27 member states of the European Union (INSPIRE 2007). For the implementation of the INSPIRE directive more and more Geo Web Services (GWS), based on specifications of the Open Geospatial Consortium (OGC) (OGC 2014), are used. With the growing number of used GWS, especially the Web Map Service (WMS), a growing awareness regarding quality aspects can be observed. According to ISO 9000 (2005) quality is understood *as "the degree of a set of inherent characteristics which fulfil requirements"* (ISO 2005). The Quality of Service (QoS) refers to match the needs of service requestors with those of the service providers based on a network resource.

As a reference for QoS the W3C proposes a set of quality elements (which is also called a “quality model”) for web services which have been established by a web services working group (W3C 2003). The document focuses on web services without any geospatial consideration.

Table 1 shows the proposed quality elements. For INSPIRE this proposal and others were discussed and a subset of these elements were adopted (elements in blue color in table 1) for the GWS and additionally the element regulatory was introduced (INSPIRE 2007).

Performance	Accuracy
Reliability	Integrity
Capacity	Scalability
Availability	Robustness
Security	Exception Handling
Interoperability	Network related QoS
Accessibility	Regulatory (INSPIRE only)

Table 1. Quality Elements according to W3C (black and blue color) and INSPIRE (blue color)

For three of these QoS elements measures were defined (INSPIRE 2013) and thresholds, which should be fulfilled by all INSPIRE service implementations of WMS, WFS and WCS. The following example shows the definition for an INSPIRE view service (WMS).

- Performance

  "*For a 470 KB image the response time for sending the initial response to a GetMap request to a view shall be a maximum of 5 seconds in a normal situation. A normal situation represents a period out of peak load, which is set at 90 % of the time*"

- Capacity

  “The minimum number of served simultaneous service requests to a view service according to the performance quality of service shall be 20 per second.”

- Availability

  “The probability of a Network Service to be available shall be 99% of the time.”

The INSPIRE directive provides only information about evaluation and assessment criteria but no information how to set up a valid test procedure to test the GWS. There is still a lack of a common basis for defining, specifying and especially determining the QoS of GWS. This paper presents a common evaluation procedure, including possibilities for defining quality elements with their corresponding metrics. Further the evaluation procedure is used to investigate the performance bottlenecks of a WMS.

The remainder of this paper is organized in the following way: In section 2 a general evaluation procedure for GWS is presented. This is a common evaluation procedure, which also includes a test management for GWS. In section 3 we apply this procedure in a case study and investigate the performance bottlenecks of a WMS. We analyzed an OGC compliant WMS to find out which components of a WMS consumes most of the time during the WMS-request. Section 4 concludes the paper and gives a short outlook on future research topics.

2. Evaluation procedure for Geo Web Services

An evaluation procedure for GWS is an important aspect and needs to be considered in order to determine the quality of a GWS. The following section points out different aspects for a quality evaluation of GWS.

Evaluation of software and corresponding testing is especially important during the process of software development but has only some overlaps with our case, the evaluation of the quality of services. The latter is characterized through the following:

A service (e.g. a WMS) is established by using one or more software components and delivers data (maps, data, metadata) in a specified quality. That means, the goal here is not the evaluation of the software but the evaluation of the quality of the service, which has been set up. Also it has to be noticed that not all of the quality elements defined e.g. by INSPIRE might be relevant in a specific case. That means a general evaluation procedure has to support the following parts:

- The Identification of the quality elements relevant for the specific case
- The choice of a suitable metric to measure the behavior of the services related to a specific quality element. Within this step it should also be possible to define thresholds related to some of the quality elements, as this has been done for INSPIRE (see section 1)
- The test of the service related to the quality elements identified under consideration of the selected metrics / measures. The goal of this test might be to investigate if given thresholds are fulfilled or to investigate

the general characteristics of the service, e.g. how the service behaves if the number of users increases or the amount of data grows

- Finally the results have to be analyzed and the set-up of the service might be changed. Of course these procedure can be iterative

Figure 1 illustrates this quality evaluation workflow. This evaluation procedure is discussed in more detail in the next section.

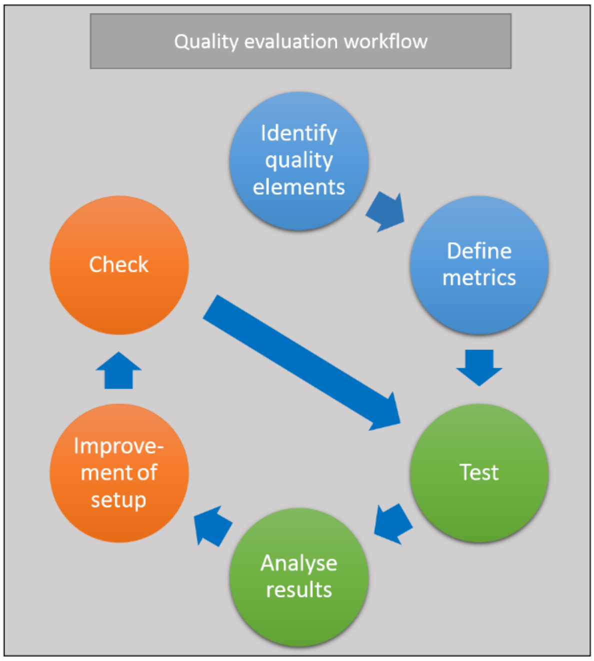

Figure 1. Quality evaluation workflow

1. Identify quality elements: Different quality elements need to be chosen according to the requirements. Requirements for a GWS are often stated in a Service Level Agreement (SLA).

2. Define metrics: Adequate metrics need to be defined for the quality elements in order to be able to measure the behavior of the service. The identification of quality elements and the definition of metrics can be done within two subsequent steps for example by using the Goal-Question-Method (GQM). The selection of the elements can be done by choosing from the quality models of W3C or INSPIRE (see above), but other models i.e. ISO 9126-1 (ISO 2001) which has been replaced by ISO 25010:2011 or the OASIS model (OASIS 2005) can be used. This step is highly depending from the specific goals and boundary conditions of the set-up of the service and has therefore to be performed for each set-up individually.

In applying the GQM method, the first step is to specify goals (together with the customer) and then a set of questions are formulated which help to fulfill the general goal. An example for the quality element performance is given in **Fehler! Verweisquelle konnte nicht gefunden werden.**. For a detailed introduction to the GQM method for GWS see [Schmid & Reinhardt 2013].

Additionally it is also possible to define thresholds related to some of the quality elements.

Goal	**Object of study**: **Purpose**: Focus: Points of view:	Performance To assess Throughput developers / users
Question	Is the database performance good enough? **Metric**: Number of request per second.	

Table 2. GQM-method

3. Test: This step implies all actions to measure the behavior of the set-up related to the chosen elements / metrics. This determines the current state of the GWS. Independent of the general test frame, it is necessary to establish a common test management concept for testing GWS. The test management concept can be stated within six phases for GWS. The phases can often run in parallel. **Fehler! Verweisquelle konnte nicht gefunden werden.** shows the procedure of the test management concept.

1. Test planning
2. Test design
3. Test case determination
4. Process planning
5. Test execution
6. Test evaluation

After finishing the tests, a report is generated. It documents the test activities and results and provides suggestions for future tests (Schmid 2014).

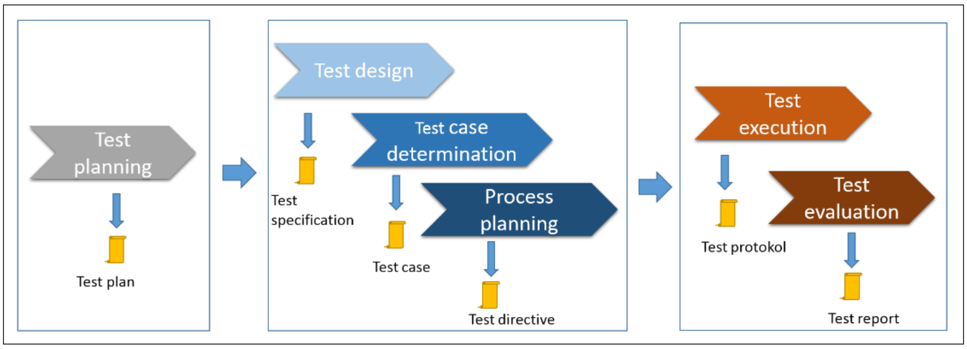

Figure 2. Test management concept

4. Analyze the results: If thresholds have been defined, it is important to analyze if the set-up of the service fulfills these requirements. If this is not the case, general behavior of the set-up related to the selected elements has to be analyzed and it has to be decided if this is sufficient for the application.

5. Improve the GWS setup: The evaluation of the results of the quality analysis might lead to an adjustment of the GWS setup/configuration. The enhancement of the GWS setup can be hardware related or software related.

- Hardware related: Increase the server hardware, especially RAM, Hard Disk and processor. A distributed server environment may also enhance the GWS
- Software related: Adjust the software setting. This can be done for example by increasing buffer and cache sizes or by limiting the amount of requests.

6. Check: This implies to repeat the GWS measurements. The quality evaluation workflow may be used and repeated during different phases of a GWS life cycle. Quality control is necessary during GWS creation, update and operation.

3. Case Study- WMS

The following case study was part of a project for a German utility company. The main evaluation subject was a WMS set-up based on the open-source software Geoserver. The service set-up should be inspected according to its usage within the company. The main issue for the company is the fast transfer of data to quite a number of users. For defining the required quality elements, the GQM-method was used in cooperation with the users. **Table 3** includes the results of the GQM-method for the quality element performance. Additionally the GQM-method was carried out for capacity and availability. The discussion about the threshold for the defined metrics led to the result to use the INSPIRE QoS thresholds for viewing services (see section 1).

Goal: Is the GWS set-up for the provision of a WMS within the company suitable	**Object of study:** Performance **Purpose:** To assess **Points of view**: users
Question:	Does the GWS setup fulfill the INSPIRE view service performance criteria?
Metric:	Response time in seconds
Threshold:	Response time for the WMS is max. 5 seconds for 470 kb image (see section 1)

Table 3. GQM-method for the case study

After this step the activities for the tests were planned. The performance of a WMS is influenced by three major components: Network, WMS-software and the database. As the applicability of the network already was proven, no tests related to that had to be performed. In consequence, tests related to the database and the WMS-software had to be planned, which aim at the analyses of the performance / capacity of the WMS set-up. It is important to see the influence of the database request in relation to the complete WMS-request ("Get map"). In order to determine the capacity of the WMS-set-up it was investigated how a number of connections (which represents multi user access) influence the performance. These tests were performed with parallel requests (to explore the behavior for very high traffic) as well as with requests within certain time intervals from 0.25 sec up to 1 sec. The latter cases (requests within certain intervals) are better representing the situation in practice, as users are seldom performing requests, which are exactly parallel in time. For the tests data provided by the utility company was used. This study extends other WMS performance and WMS-caching studies of our group (Loechel & Schmid 2013) (Schmid 2011).

3.1. Test procedure

In accordance with the goals of the tests, the complete WMS-request was tracked as well as the database query. In **Table** the original WMS-request is illustrated while **Table** represents the tracked WMS-request to the database. The latter is just the SQL-statement the WMS-software sends to the database to obtain the requested data. The SQL-statement is a complex request with an intersection of the data with a bounding box.

```
http://serverip:8080/geoserver/cite/wms?service=WMS&ver-
sion=1.1.0&request=GetMap&layers=cite:g_hauptltg_ab-
schnitt&styles=&bbox=1264644.27945,6101196.68809,1320914.6672
6,6153315.98172&width=1024&height=768&srs=EPSG:3857&for-
mat=image/png
```

Table 4. WMS-request

```
SELECT "system_id","d_leitung_in_betrieb_style_info",en-
code(ST_AsBinary(ST_Force_2D("d_leitung_in_be-
trieb")),'base64') as "d_leitung_in_betrieb" FROM
"swm_gas"."g_hauptltg_abschnitt" WHERE  "d_leitung_in_be-
trieb" && ST_GeomFromText('POLYGON ((...))', 3857) AND ST_In-
tersects("d_leitung_in_betrieb", ST_GeomFromText('POLYGON
((...))', 3857));
```

Table 5. SQL-request

Figure 3 gives an overview of the testing scenario, for a better understanding of the test procedure. Two test-rows were performed. One testing the WMS-Request, the other one testing only the SQL-Request sent by the WMS

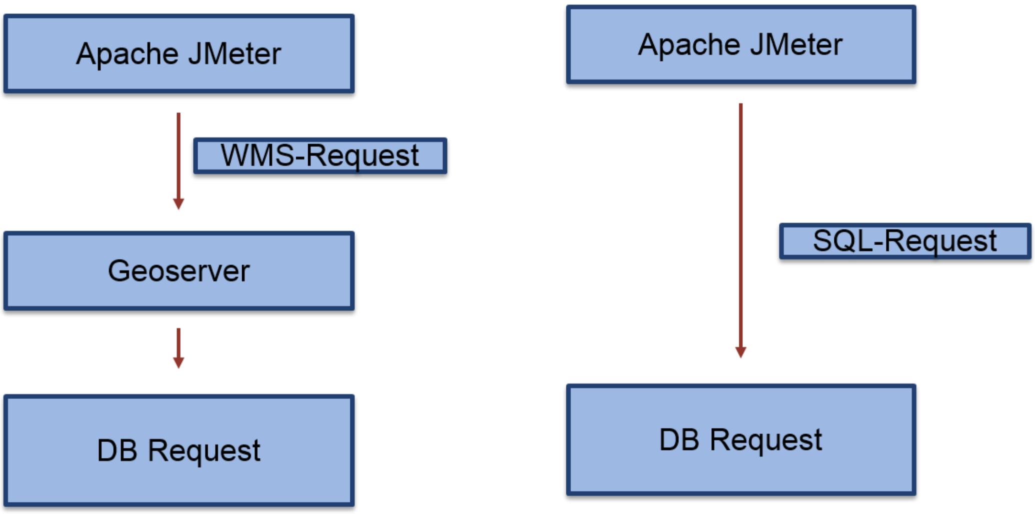

Figure 3. Overview test procedure

3.2. Test environment

For the tests the institutes testbed is used, which is established at the University of the Bundeswehr Munich. The testbed is a technical and scientific platform for experiments and investigations. For further information on the testbed, see (Schmid & Reinhardt 2011). For the study a virtual server was used with the following specifications:

- RAM: 8 GB
- CPU: 4 x 2,5 GHz
- HDD: 100 GB
- Windows Server 2008 R2

3.3. Test software

For measuring the performance Apache JMeter was used. Apache JMeter is a GUI based benchmarking tool. It is part of the Apache Jakarta Project, hence it is open source. Apache JMeter is a 100 % Java desktop application and is highly extensible through its provided API. It can be used to simulate a heavy load, using multiple threads, on the server or on the network (Halili, 2008). The most important adjustments according to the standard configuration are listed in **Table 6**.

Parameter	Configuration
Threads: Number of requests	**500**
Ramp-up Period: Time within all requests get started.	**Different**
Number of iterations: Iterations of used threads	**1**
Connections: Max. database connections	**different, increasing number**
Pool Time Out: Time to wait. After that amount of time an error will be reported.	**60000 ms (60 sec.)**

Table 6. Apache JMeter adjustments

For the set-up of the WMS the software “Geoserver” was used. Geoserver is an open-source java based GIS server software that allows users to view and edit geospatial data. Geoserver uses standards by the Open Geospatial Consortium. For the tests Geoserver 2.4 was used (Geoserver 2014).

As database backend software PostgreSQL 9.2 with Postgis 1.5 was used. In total, three different database configurations were investigated. Therefore several parameters listed in **Table 7** were adjusted.

Parameter	Minimal	Standard	High
Max. Connections: Configured on server-side. Conncetions are controlled on client side**.**	1000	1000	1000
Shared_buffers: Storage used for caching.	128 kb	32 MB	1024 MB
Tmp_buffers: Temp. Storage used for caching.	800 kb	8 MB	256 MB
Work_mem: Storage reserved in RAM.	64 kb	1 MB	4 MB
Maintenance_work_mem: Storage reserved in RAM.	1 MB	16 MB	64 MB
Max_files_per_process: Nummer of files hand able by the kernel at the same time.	25	1000	4000

Table 7. Database configuration

3.4. Results

However, with the standard DB-configuration it is also possible to get good results and even the minimal configuration leads to acceptable results. Even for 500 connections (which represents different requests) the response time does not exceed approximately 0.55 sec.

4 shows the results of the database tests for parallel requests. The DB-configuration with the highest parameter shows best results, as expected. However, with the standard DB-configuration it is also possible to get good results and even the minimal configuration leads to acceptable results. Even for 500 connections (which represents different requests) the response time does not exceed approximately 0.55 sec.

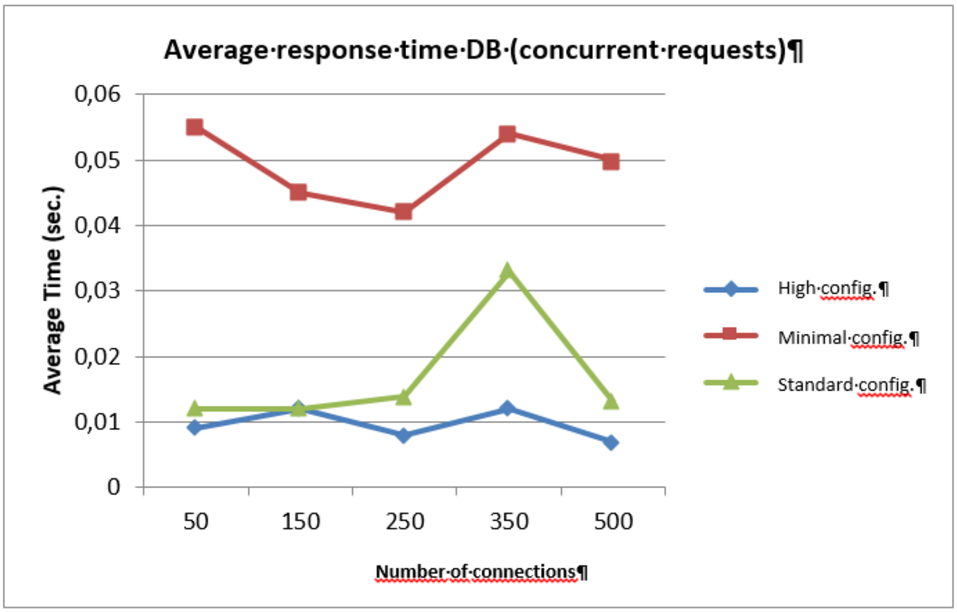

Figure 4. Average response time for the database requests. All request were started at the same time

5 shows the results according to time intervals of 0.25 up to 1 sec between the requests for the highest DB-configuration. The minimal and standard database configurations lead to similar results, and are therefore not presented here. In all cases the response times are below 0.15 sec which clearly indicates that the data base access is not a limiting factor at all.

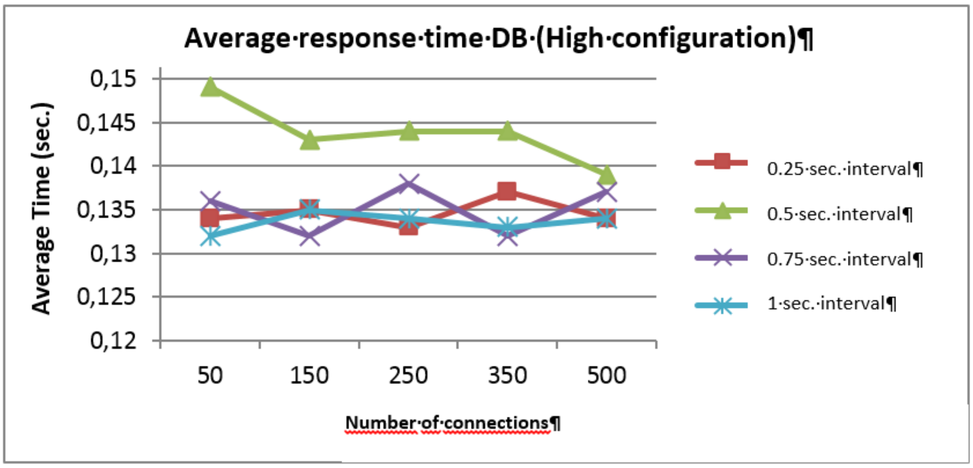

Figure 5. Database response time with high configuration according to time intervals of 0.25 up to 1 sec.

Fehler! Verweisquelle konnte nicht gefunden werden.**6** shows the results of the WMS performance tests (including DB access and rendering of the data). All requests were conducted at the same time. The chosen performance threshold (5 sec) is meet up to 25 concurrent requests. For a larger number of requests the response time exceeds that threshold.

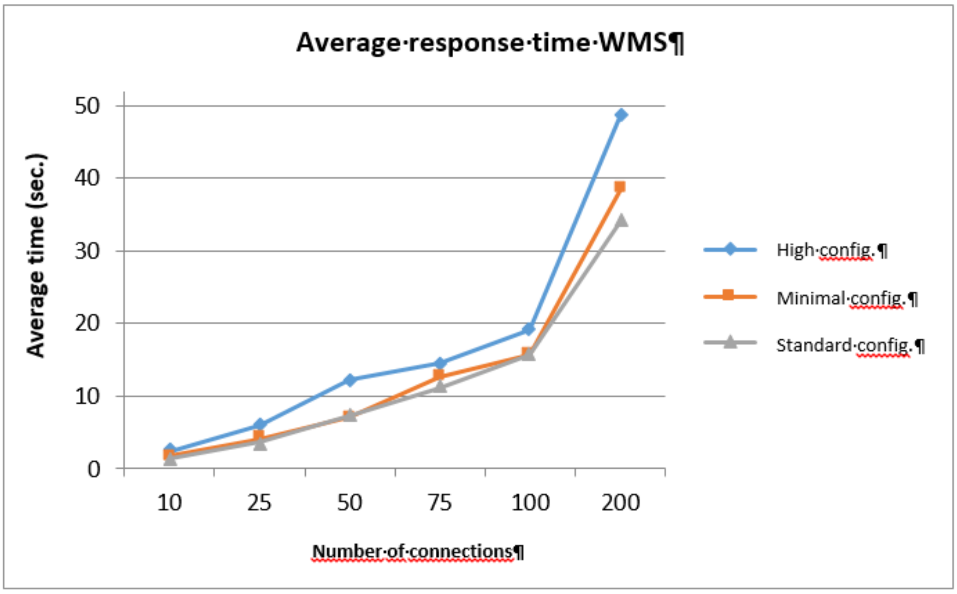

Figure 6. WMS response time for different database configurations

Figure 7 now shows the results of the WMS tests for a time interval of 1 sec. for the different DB-Configurations. The results show that for the high DB-configuration 50 requests can be handled within the chosen capacity threshold of 20 per second.

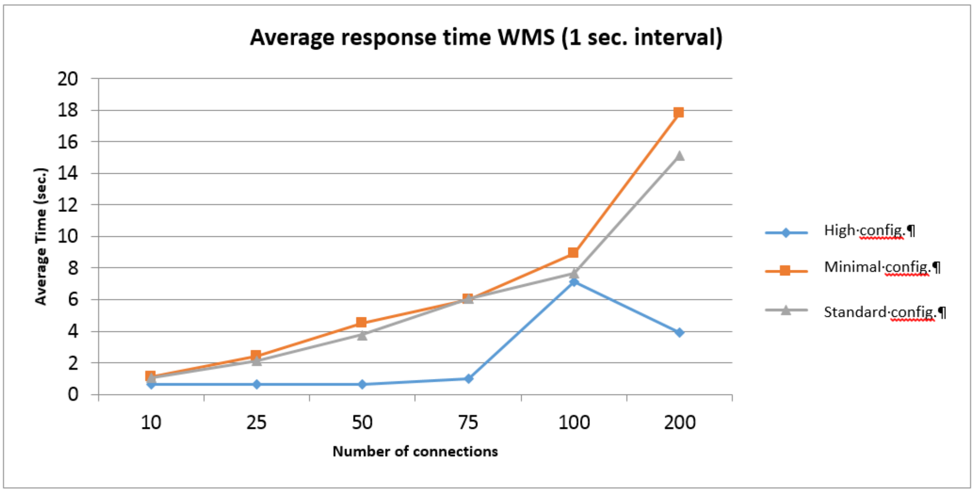

Figure 7. Average response time WMS for 1 sec interval with different DB-configurations

The results presented in the figures 4 to 7 show that the bottleneck for a WMS is the rendering time for the image. The database is able to process much more requests, as shown in figure 4 and 5.

In conclusion, we can state that the set-up of the WMS clearly fulfills the chosen thresholds.

4. Conclusion and Further Steps

The paper introduces a general evaluation procedure for GWS. Such a procedure enables service providers to provide quality proofed GWS. One component of the procedure, the GQM-method is suitable to select quality elements relevant for a specific case. Major parts of the procedure have been successfully applied in a case study.

In this case study a first bottleneck analyses on a WMS is presented. Several WMS-requests were investigated as well as to the corresponding database request. The results show the rendering time of the image is the most time consuming factor. Nevertheless, the WMS set-up clearly fulfills the thresholds which have been adopted from INSPIRE in this case.

Further research should focus on other possibilities to model the quality of services and to communicate this to stakeholders involved. One possible approach for this purpose seems to be the modelling of the QoS and its relations to involved components within a common modelling language using UML. This has the advantage that users and stakeholders have a:

- Detailed graphical specification and user friendly QoS description of the specific GWS set-up.
- Relationship presentation of quality elements and metrics.
- Documentation for QoS related software and hardware components.

For modelling the QoS several modelling languages are available, like CQML (Aagedal 2001), QML (Frolund 1998), WS-Policy (Schlimmer 2006) and other.

References

Aagedal, J (2001): Quality of Service Support in Development of Distributed Systems; Faculty of Mathematics and Natural Sciences, Universisty of Oslo; http://www-st.inf.tu-dresden.de/home/html/de/lehre/kp2003_docs/cqml_thesis.pdf; accessed 14. March 2014.

Frolund, S.; Koisten J. (1998): QML- A Language for Quality of Service Specification. HP Labs Technical Reports.

Halili, E. (2008): Apache JMeter: A practical beginner's guide to automated testing and performance measurement for your websites, Packt Publishing Limited, Birmingham. ISBN 9781847192950.

INSPIRE (2013); European Commission: http://inspire.jrc.ec.europa.eu/, accessed 09 October 2014.

INSPIRE (2013); European Commision: Technical Guidance for the implementation of INSPIRE View Services. http://inspire.ec.europa.eu/documents/Network_Services/TechnicalGuidance_ViewServices_v3.11.pdf, accessed 09 October 2014.

ISO-International Organization for Standardization (2002): ISO 9126- Software engineering – Product quality

ISO-International Organization for Standardization (2005): ISO 9000- Quality management systems — Fundamentals and vocabulary

Loechel, A.; Schmid, S. (2013): Comparison of Different Caching Techniques for High-Performance Web Map Services. In: International Journal of Spatial Data Infrastrucutures Research, Vol. 8.

OASIS (2005:) Quality Model for Web Services; https://www.oasis-open.org/committees /tc_home.php?wg_abbrev=wsqm#overview; accessed 09 October 2014.

OGC; Open Geospatial Consortium (2013): About OGC. http://www.opengeospatial.org/ogc. accessed 09 October 2014.

OGC; Open Geospatial Consortium (2012): TEAM Engine & CTL Quick Start. http://cite.opengeospatial.org/node/58; accessed 14 March 2012.

OpenGeo; GeoServer (2013): User Manual. http://geoserver.org/about/ accessed 09 October 2014.

PMQS (2014): Projektmanagement und Qualitätssicherung in IT-Projekten http://www.pmqs.de/index.php/testmanagement/prozesse/109-testmanagement-01-einleitung.html; accessed 09 October 2014.

Schlimmer, J. (ed.) (2004–2006): Web Services Policy 1.2 – Framework (WS-Policy). W3C Member Submission 25 April 2006, BEA Systems Inc., International-Business Machines Corporation, Microsoft Corporation, Inc., SAP AG, Sonic Software, VeriSign Inc. URL: http://www.w3.org/Submission/WS-Policy/

Schmid, S.; Reinhardt, W. (2011): Concept and Goals of a Geo Web Service Test Bed. In: Proceedings of International Conference on Military Technologies 2011, Brno.

Schmid, S.; Reinhardt, W. (2013): Quality of Geo Web Services. In: Proceedings of International Conference on Military Technologies 2013, Brno. ISBN:978-80-7231-918-3

Schmid, S. (2014): Konzeption einer Testumgebung zur Unterstützung beim Aufbau einer service-orientierten Architektur, internal report, unpublished.

W3C; World Wide Web Consortium (2003): QoS for Web Services: Requirements and Possible Approaches. http://www.w3c.or.kr/kr-office/TR/2003/ws-qos/, accessed 09 October 2014.