



**Defining Standards for Web Page Performance in Business Applications**


**Author: Garret Rempel**  
**MNP – Technology Consulting**

<http://www.mnp.ca/>  
[garret.rempel@mnp.ca](mailto:garret.rempel@mnp.ca)  
[@g\\_rempel](#)  
<http://mincingthoughts.blogspot.com/>

## **Introduction**

Good afternoon everyone! Thank you for having me, my name is Garret Rempel. I am a Senior Technology Consultant with MNP from Canada. I have been with my current employer for almost 10 years, and I have been working in the technology industry for almost 20. I have also been involved with performance analysis, investigation, and resolution for a large part of the last 8 years.

The paper I wrote is on Defining Standards for Web Page Performance in Business Applications. And really, what it is about is the poor quality of industry standards that exist today for web page performance, and a way by which we can go about defining performance requirements that are reasonable, useful, and are done in collaboration with the people who will be responsible for achieving them.



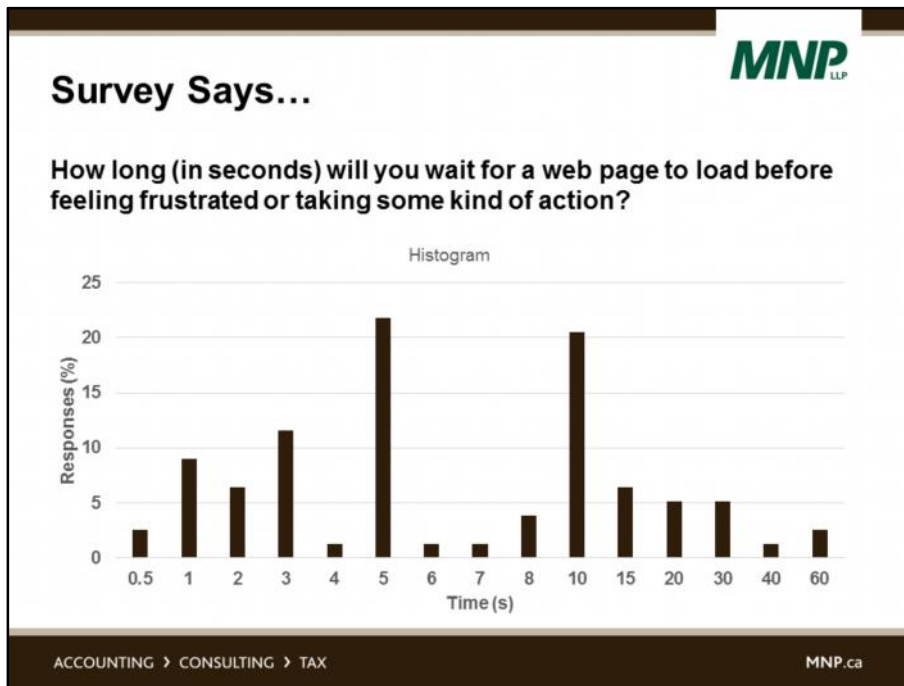
## My Job

- Educating Clients about Performance Engineering
- Building a Performance Strategy
- Testing Applications
- Measuring and Monitoring Performance
- Issue Resolution

ACCOUNTING > CONSULTING > TAX MNP.ca

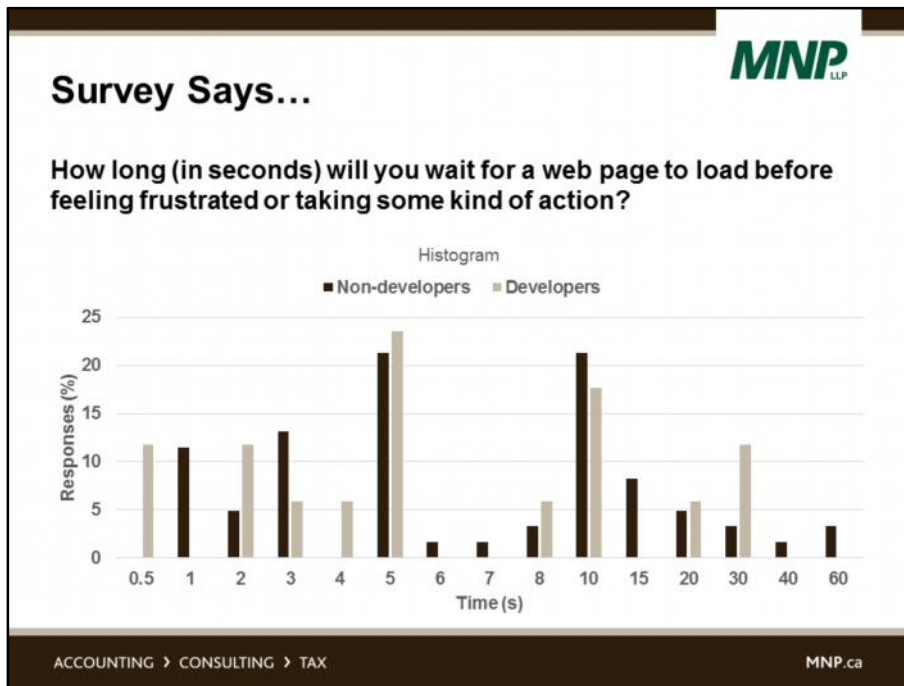
### My Job

On a new project, my job often **begins** with educating my clients. I work with their business and technology experts to aid them in building their strategy for setting performance targets, testing their applications, measuring and monitoring performance, and resolving any issues discovered.



**Problem**

One of the first problems I typically encounter on any new project though, is that, **people**, are *terrible* at setting performance targets.



And developers are not really any better. People are *awful* at assessing their own patience level and just as bad at translating that into actual requirements. Can you consider, the **one** person, that I have been given to talk to about setting performance requirements, feels that waiting 30 seconds for a web page to load is just fine?



**Challenge**

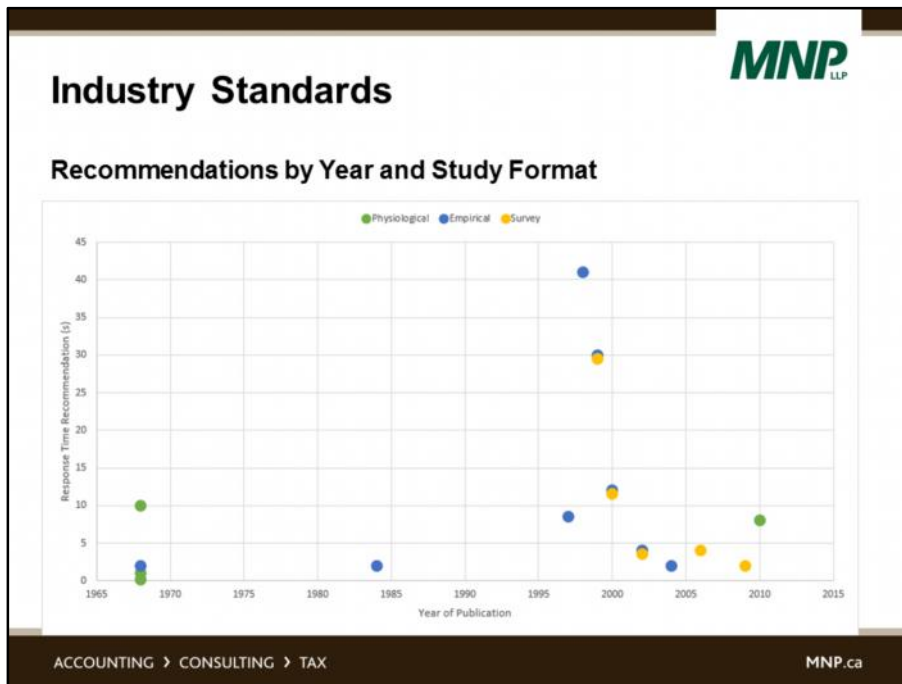
- Set accurate and precise performance requirements
- Participation from business and developers
- Buy-in from all parties involved

ACCOUNTING > CONSULTING > TAX MNP.ca

### **Challenge**


So the challenge is, how **do** we set accurate, and precise performance requirements in a manner that involves input from both the business and the developers in order to ensure they are invested in those requirements? In a way that we have buy-in from all parties involved?

If the business doesn't believe in the target, then they may undermine us with the end user. If the developers don't believe in the target, then they won't put a priority on fixing things when we fall short.



### Industry Standards

So! If we look towards industry literature for a gold standard to follow, what we find are results that are varied, inconsistent, based on questionable methodology. Frankly there is limited consensus and about as many ways of studying the subject as there are studies. Many recent ones actually rely on their subject's own self-assessment which, let's say, is of questionable value.



## Physiological Measurements

**Powers of 10 thresholds (Miller, 1968)**

- Instantaneous reaction (0.1s)
- Continuity of thought (1.0s)
- Focus on dialogue of interaction (10s)

Awareness of waiting begins at 2s  
Break in thread of communication at 4s

**Attention atrophy (Nielsen, 2010)**

Delay of 8s causes a 95% drop in user attention (Nielsen, 2010)

ACCOUNTING > CONSULTING > TAX MNP.ca

In fact, some of the most useful information dates back to Miller in 1968 who was studying Human-Computer Interfaces. Today, what we are seeing is a convergence and transparency between online and offline modes. People often don't know whether the information they are accessing is local or remote, and they are starting to expect the same level of responsiveness regardless. They expect that what they want is at their fingertips instantaneously, and they aren't going to give you the same leeway that they used to, just because you are providing an online service.

What we do know is that response times under 0.1s a person perceives as being instantaneous. Unfortunately if we are operating in a client-server model, consistent response times under 0.1s are probably not achievable – considering it takes light 0.13s to circle the earth. We start running into limitations imposed by physics. Response times under 1s though are fast enough for a person to maintain continuity of thought. That means that they do not perceive any time spent waiting, the response arrives as fast as their continuous thought process is able to interpret it. It is only in the range of 2-4s where a person begins to perceive the fact that they are actually waiting for something to happen. And once a user perceives that they are waiting, that is when interruptions, breaks in their train of thought, and

frustration can start to take hold.



## Empirical Studies – Impact of Feedback



### Nah, 2004

- Providing feedback doubles wait time tolerance
- Improves abandonment rates for slow responding pages
- Effective even after conditioning for instantaneous responses

Without Feedback	Mean	Median	Mode
First Response Failure	13s	9s	5-8s
Second Response Failure	4s	3.6s	2-4s
Third Response Failure	3.3s	2.5s	2-3s

With Feedback	Mean	Median	Mode
First Response Failure	37.6s	22.6s	15-16s, 20-22s, 45-46s
Second Response Failure	17s	8.4s	2-3s
Third Response Failure	6.7s	4.3s	2-3s


ACCOUNTING > CONSULTING > TAX

MNP.ca

Of more recent results the most referenced studies are able to provide some additional valuable insights, such as providing a responsive interface with feedback if the user needs to wait will actually result in users patiently waiting twice as long or more before becoming frustrated. However these studies also use one of the more useless gauges from a business perspective – median abandonment.

Tell me. If I am presenting to the technology director of a Fortune 500 company using one of these studies as the basis for my recommendations – and I state that if we work to achieve this level of performance I can say with confidence that only HALF of your customers will abandon your website out of frustration with its performance – really, what good is that?

If we are to set an industry standard, it must be one that produces a **high** level of *satisfaction* among users with a minimal, or better yet, zero rate of abandonment.

**Case Study – System Under Scrutiny** 

A passive, observational study of actual system performance and user behavior of a business application in production.

- Primary client information tracking and incident reporting system with an international company
- Industry-leading software platform supplied by a reputable international vendor
- 1200 users across 5 time zones in Canada and the United States by employees who are required to do so as part of their primary duties
- Peak usage is 800 simultaneous login, 50,000 page requests per hour over a 4 hour window.
- Average weekday receives 440,000 page requests with peak of 510,000 on the busiest day of the week, and 10,000,000 per month.

ACCOUNTING > CONSULTING > TAX MNP.ca

## Case Study

In order to better understand this abandonment threshold I had the good fortune to be able to work with a client to study the performance of a system I had previously helped them with. A system that had been in production mode for some time, and then take this knowledge and apply it to a new project. The system I studied has the following profile:

- Industry-leading client information tracking and incident reporting system supplied by a reputable international vendor
- 1200 users across 5 time zones in Canada and the United States who use the system as part of their primary duties
- Peak usage on this system is 800 simultaneous logins and 50k page requests per hour over a 4 hour window
- The average weekday receives 440k page requests with 10M per month.

The observation period of the study is a span of almost 2 years starting from initial go-live, a period of decaying performance, optimization, and then stable operation.

The advantage of studying this particular system is that this is a closed ecosystem. The users have no recourse to abandon the application, and they have no alternatives to work around the problems they encounter. They are motivated and encouraged to report problems and incidents of poor performance through a centralized service desk.

## Case Study – Results

System response times aggregated by month, presented as percentiles within 0.5s thresholds

	Percent of Requests Completed within Range					
	Jan 2012	Feb 2012	Mar 2012	Apr 2012	May 2012	Oct 2013
< 1.0 s	64.43	63.82	59.98	67.87	69.06	70.22
< 1.5 s	80.65	79.62	76.18	81.57	83.76	87.27
< 2.0 s	87.47	86.18	82.83	87.17	89.62	92.17
< 2.5 s	91.40	90.40	87.32	91.01	93.37	97.81
< 3.0 s	93.64	92.72	89.94	92.76	95.27	
< 3.5 s	94.94	94.12	91.65	93.82	96.29	
< 4.0 s	95.82	95.09	92.94	94.82	97.00	
< 4.5 s	96.51	95.88	94.05	96.01	97.66	
< 5.0 s	97.11	96.57	95.02	96.83	98.23	

	Jan 2012	Feb 2012	Mar 2012	Apr 2012	May 2012	Oct 2013
Complaints	17	20	22	21	13	0

### Case Study Results

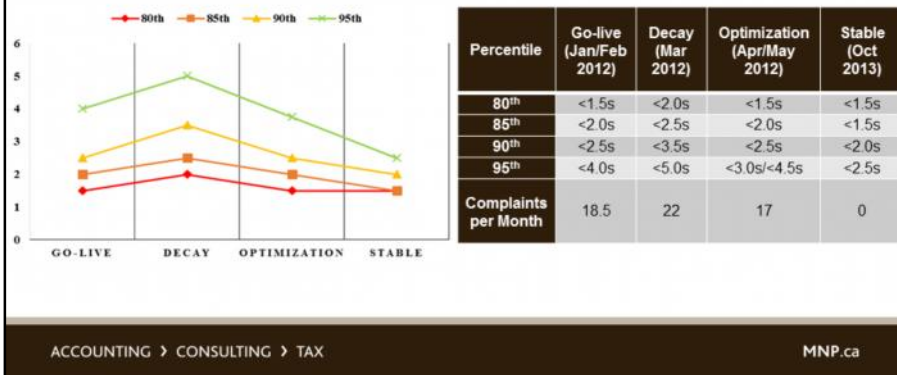
What we saw was a clear correlation between aggregate page response times, and the number of performance complaints that were issued to service desk. As you might expect, slower performance means more complaints.

What is very interesting to me though, is that the magnitude of the change in response times was really quite small, but effected a significant change on the number of complaints. And the performance threshold needed to reach a level of 0 complaints is rather tighter than many of the previously mentioned industry papers.

## Case Study – Major Percentiles

Based on this study we have sufficient information to set performance goals for future systems as follows:


- 95% of all page requests must be completed within 2.5s



The most noticeable difference between the phases of the application lifecycle is at the 95<sup>th</sup> percentile mark, which hovers around a threshold of 4-5s during the initial stages, improves to under 2.5s during its stable period.

What these numbers suggest is that for this class of application, for these users, in order to have an acceptable level of performance we need:

- 95% of page requests on aggregate must be completed within 2.5s



## Process – Gathering Requirements

**Goal:** Establish a performance requirements definition process that takes advantage of business user's input and experience and produces a result that closely matches case study observations.

To match the case study, we require:

- 95% of all web page requests achieve end-to-end response time of 2.5s or less
- A majority of individual page performance targets achieve 2.0s or less
- A limited number of pages may have larger performance targets, these must be identified as candidates for additional response feedback.

ACCOUNTING > CONSULTING > TAX MNP.ca

### Proposal

Our goal is to establish a process to define performance requirements that takes advantage of business user's input and experience and produces a result that closely matches our case study observations.

So in order to match our case study we must have:

- 95% of all web page requests achieve end-to-end response times of 2.5s or less
- A majority of individual pages achieve a target of 2s or less
- And, we need to allow for a limited number of pages to have targets greater than 2.5s where achieving that level of performance would be infeasible. These pages will be identified as candidates to provide additional response feedback in order to reduce the impact of waiting on the user's perception of system performance.

**MNP**  
LLP

## Process – Categorization

Define a set of page performance categories with examples and pre-set performance targets.

Individual pages are considered to have passed when:

- Under typical load – percentile response time measurement meets Target, overall maximum response time meets Maximum
- Under heavy (peak) load – percentile response time measurement meets Maximum

Category Name	Target Response Time	Maximum Response Time	Stability (Percentile)
Basic Operations	<2 s	<2 s	95th
Complex or Ambiguous Search or Save Operations	<5 s	<5 s	90th
Integration or Major Calculation Operations	<5 s	<15 s	85th
Heavyweight Operations	<10 s	<30 s	85th

ACCOUNTING > CONSULTING > TAX MNP.ca

### Process

To accomplish this, we designed a set of page performance categories with pre-set performance targets and in collaboration with system developers, we set specific examples of pages that would fit into each category.

A page is considered to have achieved its performance target when under typical load, the percentile response time measurement for the category meets the target, and under peak load the percentile response time meets the maximum.

The business users were then brought together and asked to categorize the complete list of pages and workflow functions into one of the given categories, or to note an exception and provide justification for why the exception was necessary.

**MNP**  
LLP

## Process – Page Aggregates

**Result:**

Category Name	# of Pages	% of Total Pages
Basic Operations	222	85.71
Complex or Ambiguous Search or Save Operations	29	11.20
Integration or Major Calculation Operations	1	0.39
Heavyweight Operations	7	2.70

Weighted averages for all pages:

- Target Response Time: 2.56s
- Maximum Response Time: 3.14s

ACCOUNTING > CONSULTING > TAX MNP.ca

### Results


Using this process against the application in the case study, we identified 259 distinct web pages or workflow functions that were categorized and evaluated.

85% of the pages were placed by the business into the Basic Operations category with a target of <2s. 11% were in the next category of Complex Operations, given a target of <5s. And only 8 pages, or 3%, were given larger targets. Every page was placed by the business into one of the predefined categories with not a single exception.

Calculating the weighted average for all the page performance requirements produced a resulting Target Response Time of 2.56s for 94% of all pages, and a Maximum Response Target of 3.14s.

This is pretty close to our actual observed level of 95% of pages at <2.5s that resulted in a 0 complaint level, not quite, but pretty close.





## Process – Request Aggregates

Performance requirement categorizations were then adjusted for the frequency of usage

**Result:**

Category Name	# of Page Requests During Test Cycle	% of Total Page Requests
Basic Operations	353,737	89.54
Complex or Ambiguous Search or Save Operations	33,550	8.49
Integration or Major Calculation Operations	2,942	0.74
Heavyweight Operations	4,819	1.21

Weighted averages for all page requests based on frequency:

- Target Response Time: 2.37s
- Maximum Response Time: 2.69s


ACCOUNTING > CONSULTING > TAX
MNP.ca

But it gets better.

When we adjust the categorized pages for their expected frequency of use what we see is that the Basic Operations category increases from 85% to almost 90% of all page requests.

Taking the weighted average then results in a Target Response time of 2.37s for 94.3% of all page requests, and a Maximum Response Target of 2.69s.

If our application can hit these targets, then we are right in line with what our case study has shown us, that if we can get 95% of all page requests under 2.5s, our application will perform at an acceptable level for everyone using it.



## Conclusions

Industry performance standards are widely variable and inconsistently structured and researched. However, a careful study of a web application that exists in a controlled environment shows that the actual wait time tolerance of the users in the study closely aligns with the most popular performance recommendations of <2s.

By using this case study to pre-define performance target categories with assistance from business analysts and system developers, business users with no particular training or experience with performance requirements were able to independently define performance requirements that closely aligned with the observed optimal performance state of an existing production application.

ACCOUNTING › CONSULTING › TAX MNP.ca

### Conclusions

What this shows us is that having a general understanding of our target audience's wait time tolerance, with some assistance from the development team, we **can** enable business users with no particular training or experience in the field of performance, to independently define requirements that closely align with their user's needs.



**Defining Standards for Web Page Performance in Business Applications**

**Author: Garret Rempel**  
**MNP – Technology Consulting**

<http://www.mnp.ca/>  
[garret.rempel@mnp.ca](mailto:garret.rempel@mnp.ca)  
[@g\\_rempel](#)  
<http://mincingthoughts.blogspot.com/>

Slides:

1. Background
2. Problem Statement
3. Convergence in Expectations
4. Types of Studies
  1. Physiological Measurements
  2. Empirical Studies
  3. Empirical Studies – Feedback vs None
  4. Surveys
  5. Findings
5. Case Study
  1. System Under Scrutiny
  2. Methodology
  3. Findings
    1. Communication Latency
    2. Render Time
    3. Response Percentiles
    4. Complaints
    5. Major Percentiles
6. Gathering Requirements

1. Categorization
  2. Process
  3. Page Aggregates
  4. Request Aggregates
1. Conclusions