Project Name:	Allegro - Optimize TCP Services over Cellular Networks	**Start Date:**	2022-02-01
Project Number:	2202	**Completion Date:**	2022-11-30

Project Details:

Scientific or Technological Objectives:

Measurement	Current Performance	Objective	Has results?
Scalability (%)	(not set)	(not set)	No
Throughput (MB)	(not set)	(not set)	No
Reduce TCP timeouts (number)	(not set)	(not set)	No
Reduced data vs TCP (%)	60	55	Yes
supported devices (number)	500	500	Yes
Minimum timeout (minutes)	5	1	Yes

The following example is based on the Tax Court of Canada judgement ALLEGRO WIRELESS CANADA INC., and HER MAJESTY THE QUEEN, March 31, 2021.

TA1/TO1 of 2010 Project 2

142] As discussed previously, the CRA split the 2010 Project 2 into three components. Mr. Rupel explained to the Court a number of general terms/concepts that applied to the entire 2010 Project 2.

143] The technological objective of 2010 Project 2 was to:

. . . develop methods and techniques to improve the scalability and throughput of TCP (Transmission Control Protocol) services transmitted over IP (Internet Protocol) on cellular networks. In particular, the objective was to develop methods to enable more efficient streaming of digital audio, connection-handling mechanisms to translate UDP to TCP and reduce the overhead related to TCP timeouts.3

144] Mr. Rupel explained to the Court the meaning of UDP and TCP. He also explained what is meant by a load balancer, session control and caching.

145] He noted that TCP is a protocol for internet communication. TCP is built on top of the internet protocol and provides a reliable way for the vast majority of things on the internet to communicate.

146] UDP is another protocol that is also built on top of the internet protocol, similar to TCP. UDP is a very lightweight protocol when compared to TCP, but TCP does a number of things that are not done by UDP.

147] Mr. Rupel explained that when a large amount of data is being sent over the internet, it gets broken down into pieces (packets) and each packet is sent separately through the internet protocol.

148] He noted that the advantages of TCP include the fact that it guarantees that the packet of information sent over the internet actually arrives at its destination. If the packet does not show up at the destination, TCP sends a notice to the sender of the information identifying which packet did not arrive. It also has a feature that ensures that packets of data, once received, are placed in the correct order.

149] UDP does not have these features but since it does not have as much “overhead” it can be faster than TCP. The Appellant created a UDP protocol that allowed its clients to reduce their data usage on the wireless cellular networks, which significantly reduced the clients’ costs. Mr. Rupel emphasized that at the time the cost of bandwidth on cellular networks was very expensive.

150] When the Appellant created the UDP protocol, it worked very well, however at some point problems developed. It determined that the problems were being caused by the interaction between its UDP protocol and new firewalls that were being installed by its clients.

151] Another problem related to load balancers. Clients that had a large amount of traffic on their networks and multiple servers used these load balancers. The purpose of the load balancers was to balance the usage of each of the servers.

152] The Appellant encountered problems with the interaction of load balancers and session control. Session control refers to managing sessions between a server and a specific user (referred to as a client). Instead of the client having to send all of the same information repeatedly to the server, the server stores some of the information until the session is completed. This is referred to as caching. A problem arose when load balancers caused portions of the information transferred to be stored on different servers.

153] Because of these issues, the Appellant was required to abandon its UDP protocol. It then worked to develop a TCP

protocol that would work better than its UDP protocol and still reduce the client's data usage.

Field of Science/Technology:

Telecommunications (2.02.07)

Project Details:

Intended Results:	Improve existing processes
Work locations:	Research Facility
Key Employees:	Wesley Rupel (Allegro PI) (Physics - PhD (1985) / President & CTO), Gerald Penn (Allegro Expert Witness) (Computer Science - PhD (1991) / Expert witness)
Evidence types:	Test protocols, test data, analysis of test results, conclusions; Design, system architecture and source code; Design of experiments; Records of trial runs

Scientific or Technological Advancement:

Uncertainty #1: Byte array pool

154] Mr. Rupel discussed the portion of 2010 Project 2 identified by the CRA as TA1/TO1.

155] The Appellant's CRA filing described the technological advancement that the Appellant was trying to achieve with respect to the TA1/TO1 portion of the 2010 Project 2 as follows,

"the implementation of a non-disposable byte array pool

into which digital audio was compressed for transmission

completely eliminating audio breakup caused by buffer under runs

(the under runs were in turn caused by insufficient packet throughput)."

156] When switching from a UDP protocol to a TCP protocol the Appellant encountered a problem with audio files.

They were not being sent fast enough and only a portion of the audio file could be played when first accessed by the recipient of the audio files.

The most significant underlying key variables are:

causes low package througput (unresolved), unsafe attributes checking vs speed, parameters relevant to digital audio transmission, eliminating buffer under runs

Technology or Knowledge Base Level:

Benchmarking methods & sources for citings:

Benchmark Method/Source	Measurement	Explanatory notes
Internet searches	20 Articles	Client searches did not yield any existing library of required info.

Activity #1-1: Activity 1 (Fiscal Year 2022)

Methods of experimentation:

Method	Experimentation Performed
Analysis / simulation:	3 alternatives
Trials:	14000 runs / samples

156] When switching from a UDP protocol to a TCP protocol the Appellant encountered a problem with audio files. They were

not being sent fast enough and only a portion of the audio file could be played when first accessed by the recipient of the audio files.

157] The Appellant began experimenting with different ways to compress the audio files. At first, the methods it tried were not successful.

158] It then began experimenting with what is known as unsafe attributes. Mr. Rupel explained that the software language that the Appellant was using had a managed environment. Basically, it ensured that the various source code being used operated properly. However, the code used to do this slowed things down.

159] The Appellant tried writing so-called unsafe attributes by adding code that was not going to be managed. Mr. Rupel described the effect of unsafe attributes as follows:
. . . You don't have a safety net underneath you anymore, you're walking across the tightrope hoping that you don't have any bugs at that point because if you do it's not going to catch them, it's not going to prevent you from hurting yourself.5

160] He noted, however, that it resulted in less overhead, which meant that the Appellant could hopefully push data through quickly enough to solve the problem.

Results:

- Reduced data vs TCP: 52 % (160% of goal)
- supported devices: 500 number (100% of goal)
- Minimum timeout : 2 minutes (75% of goal)

161] The use of unsafe attributes did not work. The ultimate solution involved going back into the so-called managed world and using a hybrid solution where the Appellant was "doing things that [were] a little bit unsafe but not particularly unsafe."

162] Mr. Rupel provided a detailed technical description of the solution. It involved reusing certain of the objects that had been transferred. This saved enough overhead that the Appellant was able to solve the problem with the audio files. The solution worked for whatever audio was sent and on the approximately 500 different devices it encountered.

Conclusion:

165] In his expert report, he (Dr. Penn) stated the following:

. . . Programming with audio is a very niche expertise that most software engineers lack. This observation, combined with the increasing demand for smartphones over the last seven years, has led to a commodification of audio processing hardware and audio processing APIs within the mobile device industry that has greatly consolidated during the interval.

In 2010, however, there was still a considerable variance among handheld mobile devices in the range of supported audio formats, audio codecs, available audio transfer rates and supported functionality for audio in vendor-supplied APIs.

In the present component, these audio-specific parameters were underlying technological uncertainties in an ecology of foreign devices that Allegro's platform developers would have had to adapt their product to. ...characterizing the distribution of parameters relevant to digital audio transmission in 2010,

Significant variables addressed: eliminating buffer under runs, parameters relevant to digital audio transmission, unsafe attributes checking vs speed

Documentation:

- Offline Documents: docs

Uncertainty #2: Synchronous event wrapper

TA3/TO3 of 2010 Project 2
166] The Appellant's CRA filing described the technological advancement that it was trying to achieve with respect to the TA3/TO3 portion of 2010 Project 2 as follows: "The development of an [sic] synchronous event wrapper capable of timing out a process quickly, eliminating an average wait of 5-8 minutes for a TCP timeout from a mobile device".10

167] Mr. Rupel explained what a synchronous event was by distinguishing between a synchronous event and an asynchronous event.

A synchronous event occurs when, in the course of communication, the system sends a request for information and then waits until it receives the answer.

An asynchronous occurs when the systems sends a request for information and then does other things while another part of

the system waits for the answer.

168] Mr. Rupel noted that with a synchronous event the whole system is waiting for the response and with an asynchronous event it is not waiting for the response. The synchronous method is used when the system cannot move forward in the logic of the program until the system receives an answer.

169] The problem the Appellant faced was that Microsoft had built a five-to-eight- minute timeout into its software that controlled the low-level features of the hand- held devices. The Appellant had no control over this timeout. Problems occurred in the TCP communication when a request was going out for information and no information was coming back. The Microsoft software would then take at least five to eight minutes to reset. This was a problem for the Appellant, which was trying to make devices that worked in real time, i.e., were always connected to the network.

170] Mr. Rupel described the problem as a software problem that occurred because Microsoft developed the software using protocols from a wired network and the devices were now being used on a wireless network.

He noted that the designers of the software never envisaged a situation where the device would be connected but could not send data, but this is a common occurrence for a device on a wireless network. An example of this type of situation is when a device is taken into a parking garage with poor reception.

171] While it was a design feature of the Microsoft software, the Appellant had to fix the problem without access to the code used by Microsoft, while operating in a very complex system.

The most significant underlying key variables are:

sync vs async events, redesign legacy hardwire code for wireless, firewall and deep packet inspection, packet loopback process, parallel session process

Technology or Knowledge Base Level:

Activity #2-1: Supporting Activity (Fiscal Year 2022)

Methods of experimentation:

172] Mr. Rupel described three methods that the Appellant tested in an attempt to solve the problem.

173] The first method involved using a firewall and deep packet inspection to terminate long-running connections that were waiting for a response. The Appellant was trying to deal with the situation where the software would tell it that it was connected, but there was actually a problem and the device was not communicating.

174] He explained that deep packet inspection meant that the Appellant was “peeking” into places that it would not normally be expected to go, namely the network buffers where the information was coming in through the TCP network.

175] Since the firewalls monitored the system traffic and knew exactly what was passing through the network, the firewall could be used to find information on what was going through the network.

176] Testing using the deep packet inspection and firewalls did not lead to a solution to its problem.

177] The second method it tried involved experimenting with a loopback process which involved sending a packet out through the networking layers with instructions that the place it should go is back to the point where it originated.

178] It hoped to avoid the five-to-eight-minute timeout problem by killing the network session, which, theoretically, would cause everything to immediately reset.

The problem it encountered was that it was only able to kill one side of the session (such as the device side) but was not able to kill the session on the other side (the server side). This left the system in what Mr. Rupel referred to as an inconsistent state, which caused a problem.

Results:

179] The problem was resolved by developing a two-pronged mechanism to eliminate the issue. Mr. Rupel described the process as follows:

. . . by creating another process we created a parallel situation where we could start a new session for -- the original session, which is where everything is still happening, all the important stuff is still going on there, but we create this other process that then creates a new session with the server, and then we have to basically keep track of what's happening over there but we can use that channel then to do our communication until that five-to-eight-minute timeout finally times out.

And so we have sort of a temporary communication channel that we set up during the period of time that the five-to-eight-minute window is blocking us.

Conclusion:

Doctor Penn's Opinion - SUPPORTING ACTIVITY VS. SEPARATE PROJECT

181] Doctor Penn does not believe that TA3/TO3 of 2010 Project 2 by itself constituted experimental development or research. However, he believes that TA3/TO3 supported the other parts of 2010 Project 2, which he believes constituted experimental development or research.

In effect the part of 2010 Project 2 that the CRA identified as TO3/TA3 supported the other portions of 2010 Project 2 in such a way that it contributed to the overall aims of an experimental development project.

He questioned whether it made sense for the CRA (or anyone) to split the Appellant's 2010 Project 2 into three components.

182] He stated the following in his expert report:
. . . This project's [2010 Project 2] description proposes one overarching technological advancement: a TCP-based application protocol that surpasses UDP in throughput and scalability. Whether or not this could be achieved was a technological uncertainty.

To achieve that advancement, there are certain design features of TCP that are inconsistent with its use in this application protocol. One of those, TO3, is the long timeout delays that are typically built into TCP stacks.

It is a defect of the subproject terminology, "TA3/TO3", that it implies such a limited scope of work as to preclude the identification of a TA or TU for just this one component. This component shares in the technological advancements and uncertainties of the project to which it contributes. . .

. . . I am unable to reasonably ascertain that a systematic investigation was conducted as part of this component's work on the basis of the documents and interviews available to me. I am, on the other hand, reasonably certain that Project 2 as a whole did consist of research and experimental development alongside some inevitable routine development that took place in support of the project's overall research programme. I also find it reasonably probable that the work described in TA3/TO3 and the associated technical content by itself was sufficiently novel to serve as the basis for a standardized extension to the TCP protocol for low-latency communication on unreliable networks. What is unclear to me is whether the realization of TA3/TO3's research potential in fact took place.

Significant variables addressed: firewall and deep packet inspection, packet loopback process, parallel session process, redesign legacy hardwire code for wireless, sync vs async events