

Brianna Bransfield

CS 450

26 May 2019

### Assignment 1

Looking at Tanenbaum and Torvalds original debate, they make many points on the advantages to using the micro/monolithic kernel organizations. Tanenbaum who starts the debate has more of a focus on Microkernel systems, specifically MINIX, and as a professor prefers more of the modern approach that they have “a MODERN ‘free’ OS look”. He even brought up that “among the people who actually design operating systems, the debate is essentially over. Microkernels have won”. However not everyone agrees that the modern look is the preferred system.

Torvalds points out that LINUX may not be as nice as Microkernels and from a theory standpoint is looses, but since it is a monolithic system that is available now it wins. It is a more probable device that can have things ported much easier. All you really need is for a device to adhere to a portable API making linux fit under the exact definition of an operating system. Tanenbaum responds by bringing up his standpoint as a professor and talks about systems that students have access to and can use, explaining his hopes for the future of operating systems.

Overall there is a lot of discussion of versions or systems have been used in the past and what was expected of the future, being the past today. When multithreading in a monolithic system it adds complexity to the code and can be debatable if it is worth the trouble. Tanenbaum makes a claim that new hardware should use microkernel and old hardware should use monolithic systems. Mixing them up would cause more problems than solutions in his case.

Many others reply saying that microkernels are the new system and future, but monolithic systems overall are still much easier to use, even if they are 'outdated'. The 80x86 system is brought up often and how linux is most compatible with it. Linux also "greatly reduces the cost of using tools like gcc, bison, bash, which are useful in the development of such an OS".

Systems like minix may be too portable, doing just as much as if it wasn't portable at all. Most people in this debate are favoring Torvalds perspectives saying that the old system is something to stick with while most people find Tanenbaum's perspectives strictly to teaching, putting monolithic systems towards the win.

In the second round of debates both Tanenbaum and Torvalds seem to have their same ideas as before. The main difference is that after about 14 years they have seen what systems work and what don't. For Tanenbaum he does admit that microkernels were "long discarded as unacceptable because of their lower performance compared to monolithic kernels". This time he focuses more on the ends of how we can make operating systems reliable and secure. He feels microkernels are being reincarnated due to their potentially higher reliability which is now regarded as more important than performance. The discussions in the first debate focused primarily on performance, showing us this by giving examples that we know and explaining the positive and negative components in research that has been done. He is preserving many of his original claims but now have evidence to back him up and does a lot of projection into the future.

Torvalds defense isn't as direct as he was before, speaking more of a 'hybrid kernel'. His argument is centered around the fact that we shouldn't be focusing on implementation details. This is mainly due to space accessibility and address space. He explains how a traditional kernel works with both address space and access space bringing up both protection and semantic issues.

This is when he claims it is harder to write microkernels, bringing up more and more problems. He doesn't necessarily defend monolithic systems like before but rather rips down microkernels due to these faults. They both stick with their original argument but instead of proving they are right, they contradict the other system.

Most of the people in the rebuttal and original arguments have an education email linked to them, either being a student or professor. It becomes hard to see what is valid since different people work with different systems, but I would say that Tanenbaum is the most valid source. His paper *Can We Make Operating Systems Reliable and Secure?* did appear in the May 2006 IEEE Computer, meaning it must have some truth to it. All of the other comments were either through an open posting system or individual site where anything can be said. It was even brought up in the original debate how some people were saying crazy things that were invalid due to lack of background. Even myself as a student I wouldn't make a strong defense to prevent myself from inaccurately guiding someone down the wrong path as Torvalds seemed to do with some people. Also Tanenbaum was the only person to use multiple different research examples to defend his point and admit his original defeat, seeing microkernels as more of a come back. After 14 years you should have some solid evidence if you are going to defend something and he did that well.

With my experience working for the US government with older systems I have fallen more towards monolithic systems like linux, but this is only due to what works easier and can be changed. In a world where we are seeing technology everywhere running faster than ever before we may need to have more of a microkernel aspect to better handle message processing between the program and computer. Our society used to have few technical people, but this is changing

with our new devices. Even if people aren't computer scientists, they are learning how computer work to make their lives easier. With this simplicity in place in the future there is a microkernel increase possibility. A system like linux generally is taught to teach students the basics, but many people stem out from there. It is an essential so you can work on an older system if needed, but they aren't as reliable and secure as we may picture them. Of course all programs have problems with this but there are little ways to improve this, seen more in microkernels than in monolithic systems. Despite all of this, I think I side with monolithic systems since all of our new systems are built from old systems. I have grown to learn that most times the original items can be better to use than the newest one, having more time to be perfected. Using monolithic systems at my job is due to the fact that it can mix old and new, allowing wiggle room when changes are made. Yes it may be long and a pain at times, but I believe it is reliable if you use it right.