

## DEPARTMENT: MICRO LAW

# A Review of *Wisconsin Alumni Research Foundation v. Apple*—Part VIII

Joshua J. Yi , *The Law Office of Joshua J. Yi, PLLC, Austin, TX, 78750, USA*

Part I of this series introduced the *Wisconsin Alumni Research Foundation v. Apple* cases and described the asserted patent (U.S. Patent No. 5,781,752). That article also summarized some recent large verdicts for patents asserted by academic institutions and provided several reasons why this series may be of interest to the readership of *IEEE Micro*, most notably because the inventors are well-known and several well-known computer architects worked as experts on this case. Part II described the complaints, namely, it described the plaintiff, Wisconsin Alumni Research Foundation (WARF), the inventors, and WARF's allegations as to how the Apple's products infringed WARF's patent. Part III described Apple's answer to the allegations in WARF's complaint, Apple's counterclaims, and WARF's response to those counterclaims. Part IV examined Apple's allegation of inequitable conduct by the inventors, a technical analysis of that allegation, and Judge Conley's legal analysis of the sufficiency of Apple's allegations. Parts V, VI, and VII reviewed three of WARF's motions to compel Apple to produce 1) executable versions of the simulators for its A6, A7, and A8 processors (hereinafter "accused products"), 2) documentation regarding future (unreleased) processors, and 3) a version of each accused processor that had the accused feature disabled, and the hearings that the court had on those motions.

## CLAIM CONSTRUCTION

On 27 March 2015, WARF and Apple each filed their own motions for summary judgment. A motion for summary judgment (MSJ) is a motion where the filing party requests that the court rule in their favor for one or more issues. If the court does so, then that party wins on that issue and it is no longer part of the trial. Some issues are case-dispositive, e.g., infringement or

invalidity, such that if the court grants the party's MSJ, then the case is over and the trial will be canceled. MSJs typically occur immediately before trial, i.e., very late in the case.

As part of their MSJs, both WARF and Apple included the list of claim terms for the court to construe, which is called "claim construction." The meaning of a claim term is a legal question, i.e., one for the court—and not the jury—to decide. For most courts, claim construction occurs relatively early in the case. But Judge Conley, the judge in the *WARF v. Apple* cases, preferred to handle them later, as part of claim construction (see page 2 of *Wis. Alumni Rsch. Found. v. Apple, Inc.*<sup>1</sup>).

There are advantages and disadvantages for both the court and the parties to handle claim construction hearings later in the case, in this case, right before trial. One advantage is that it may save the court and parties some time and effort. The parties may write claim construction briefs that may be hundreds of pages (depending on the number of patents in the case or the number of claims at-issue), which increases the time spent and cost for the parties. After reading those briefs and having a hearing on the claim terms, the court writes a claim construction order that can be similarly lengthy. This takes up a lot of the court's attention and bandwidth. By delaying claim construction toward the end of the case, courts can avoid spending all that time if the parties settle before then. In addition, because claim construction law is its own separate area of law and requires a lot of technical analysis, some courts do not like doing claim construction, so by delaying claim construction toward the end of the case, courts can delay working on this until absolutely necessary.

On the other hand, delaying claim construction until the end of the case means that the parties have more "live" issues during the case. As such, the parties may file more motions related to those issues, which increases the time spent and cost for the parties, and increases the court's workload. In addition, given that claim construction is often case-dispositive such that the case ends after claim construction (because the

---

0272-1732 © 2026 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies.  
Digital Object Identifier 10.1109/MM.2026.3659394  
Date of current version 25 February 2026.

patent is invalidated or the plaintiff agrees that the claim cannot be infringed under the court's claim construction for a particular term), waiting to construe the claim terms means that the court will need to spend time on otherwise unnecessary motions.

The purpose of the claim construction is to interpret the meaning of claim terms so that the parties know the scope of the asserted claims. Based on that scope, the parties can prepare their arguments so the jury (or the court in some cases) can determine if the accused products infringe the asserted claims and/or are invalid. As such, parties fashion their claim construction arguments accordingly. More specifically, the plaintiff typically prefers that claim terms be construed more broadly so that the scope of each asserted claim is broader such that the accused products are more likely to infringe. However, if the claims are broader, then they are also more likely to be invalid as a broader idea may have already been previously invented. On the other hand, the defendant usually argues that a claim term, and concomitantly the claim scope, has a narrow meaning such that the accused product does not fall within the scope of the claim and thus does not infringe.

### POTENTIAL NEED FOR COURT-APPOINTED TECHNICAL ADVISORS

To properly construe a term, a court needs to both understand the technology that the patent covers and claim construction law. Most judges and clerks, however, do not have a science, technology, engineering, and mathematics (STEM) background. For example, one study found that only 7.6% of federal judges majored in science or technology fields.<sup>2</sup> As such, those judges may not adequately understand the technology in even technically simple patents. Some judges realize the importance of having "in-house" expertise with science and technology, so they may hire a clerk with a technical background. But, even that is not a silver bullet as those clerks may only have a bachelor's degree in a STEM field, but not necessarily in the same technical field as the patents. For example, that clerk may have a mechanical engineering degree or chemistry degree when the patents are electrical engineering patents. In addition, the average clerk with a technical background generally does not have a graduate degree in a STEM field and/or any significant technical work experience; rather, that clerk more likely immediately went to law school after getting a bachelor's degree. Furthermore, many judges and clerks do not have much experience with claim construction, so they are also less familiar with claim construction law.

As such, on appeal, the Federal Circuit, which is the appellate court that handles the appeals for all patent cases, frequently finds that lower courts incorrectly construe claim terms. One study found that the Federal Circuit found that 33% to 44% of claim terms were incorrectly decided by the district courts.<sup>3</sup> The reasons for that high reversal rate is not because the judges and clerks are not smart or do not work hard. Rather, it is generally due to their unfamiliarity with technology and their inexperience with claim construction.

To address these two problems, some judges appoint a technical advisor, which is usually a lawyer with a technical background. Those lawyers often have previously clerked for that judge and been appointed by that judge multiple times. As such, that lawyer usually works well with that judge and knows how the judge approaches things. Furthermore, because these lawyers are frequently appointed to help the court with claim construction (or other issues), they gain a lot of experience such that they could be claim construction experts. That said, the lawyer may not have a technical background in the same area as the patents, so the lawyer's technical knowledge for those patents may be only slightly better than a nontechnical judge or clerk.

When courts appoint a technical advisor (or other similar positions), one study found that the parties appeal fewer issues and the Federal Circuit is less likely to reverse the district court on the issues that were actually appealed.<sup>3</sup> This is not surprising as having a technical background—even if it is in a different field as the patents—is still better than not having a technical background, and having significant claim construction experience means that the technical advisor can help the judge come to the right technical and legal decision.

### PERSON OF ORDINARY SKILL IN THE ART

One key concept in patent law is the person of ordinary skill in the art ("POSITA"). Courts evaluate the meaning of a claim term based on what a POSITA would understand the term to mean. The POSITA is not a person with expert level skill, but rather a person of ordinary skill. The typical qualifications for a POSITA is a person with a B.S. in electrical engineering and two to three years of work experience in a specific area of electrical engineering or an M.S. in electrical engineering and one to two years of work experience; more work experience may be a substitute for a lack of degrees and vice-versa.

In this case, Apple proposed that a POSITA "would have had a Ph.D. in electrical engineering, computer

engineering, or computer science with a focus on computer architecture or microprocessor design; or an M.S. or B.S. degree in electrical engineering, computer engineering, or computer science with significant work experience relating to computer architecture or microprocessor design” (see page 13 of *Wis. Alumni Rsch. Found. v. Apple, Inc.*<sup>4</sup>). WARF proposed that a POSITA would have similar or slightly lesser credentials, namely, “at least a bachelor’s degree in electrical engineering or computer science, and at least three to five years of experience in computer design and computer architecture. Alternately, a person of ordinary skill in the art would have a master’s degree in electrical engineering or computer science, and at least two to three years of experience in computer design and computer architecture. The experience could be derived from either industry or working in a dedicated computer architecture research program” (see page 13 of *Wis. Alumni Rsch. Found. v. Apple, Inc.*<sup>4</sup>). Both parties may have proposed slightly higher-than-typical level of skill for the POSITA given that this patent was invented by leading computer architects and its corresponding paper was published in a prestigious conference. Similarly, Apple’s proposal that a Ph.D. is the level of a skill for a POSITA—which, again, is the ordinary level of skill and not an expert—may also be due to the same reasons.

Another reason why Apple may have proposed that a Ph.D. was the required level of skill for the POSITA is to help with invalidity. For example, a POSITA that has a higher level of skill may be more likely to understand that the combination of two or more prior art references disclose the invention in the asserted patent.

Neither of the judges who presided over the WARF cases had a technical background, and it is also highly unlikely that any of their clerks did either. Furthermore, neither judge appointed a technical advisor to help the court with claim construction. Therefore, despite the fact that the parties agreed that the POSITA had a higher level of skill than is normal and despite the fact that neither judge had a technical background, both judges made decisions on what the meaning of various claim terms meant to a POSITA.

### Claim Construction in the WARF Cases

Prior to suing Apple, WARF sued Intel. WARF used the same law firm, Irell & Manella, in both the *Intel* and *Apple* cases, and Apple and Intel both used the same law firm, Wilmer Hale. Because the same patent was at issue in both the *Intel* and *Apple* cases, and because the parties used the same law firms, in addition to looking at the parties’ claim construction arguments in the *Apple* case, it is also relevant to look at the disputed claim

terms in the *Intel* case, and WARF’s and Intel’s claim construction arguments.

Table 1 lists the claim terms that were at-issue in the *Intel* case while Table 2 lists the terms that were at-issue in the *Apple* case, but not in the *Intel* cases. In addition to the six terms in Table 2, one term (“prediction”) in Table 1 was at-issue in the *Apple* case. In the second and third columns of Tables 1 and 2 are the parties’ competing constructions for each term.

Of the nine terms in Table 1, the court decided that, based on “the parties’ arguments at the hearing, their prehearing briefs, the patent claims, patent specification and prosecution history,” that it only needed to rule on five of the nine terms in order to “resolve the parties’ disputes” (see pages 1–2 of *Wis. Alumni Rsch. Found. v. Intel Corp.*<sup>5</sup>). In particular, the court only construed Terms #1 (“in fact executed”), #2 (“data speculation circuit”), #3 (“mis-speculation”), #5 (“predictor”), and #7 (“prediction”). In the subsequent *Apple* case, the parties adopted the *Intel* court’s constructions for all but “prediction.” For that term, the parties proposed slightly different constructions from what the court decided in the *Intel* case.

Due to space limitations, a full description of the parties’ arguments for each term in Tables 1 and 2, the courts’ analyses and reasoning, and my analysis of the parties’ arguments and court’s reasoning will appear in the next articles in this series. That said, there are a few high-level observations that are common across terms.

First, in my opinion, the parties’ proposed construction unnecessarily construe the term. The “heavy presumption”<sup>6</sup> is that a claim term should be construed as having its ‘plain-and-ordinary meaning,’ which is the meaning that a POSITA would understand the term to have in light of the claim, specification, prosecution history, and the meaning that the term has in the technical field. For terms such as “pipeline,” “superscalar,” “branch predictor,” “reorder buffer,” “cache,” or “latency,” there is no need to provide a specific construction as the plain-and-ordinary meaning is well-understood. Rather, the parties can propose and the court can conclude that no construction is needed, i.e., the term should be construed according to its plain-and-ordinary meaning.

For terms where there is a well-understood meaning, it is better for a court—especially one where the judge and clerk do not have technical backgrounds—to adopt a “plain-and-ordinary meaning” construction rather than attempt to describe what the plain-and-ordinary meaning is. In this situation, any construction that the court will come up with may not be exactly what a POSITA would understand this term to mean.

**TABLE 1.** List of disputed claim terms in the *Intel* Case, and the parties' competing constructions.

Term	WARF's Proposed Construction	Intel's Proposed Construction
#1: in fact executed	A LOAD instruction is "in fact executed" before the STORE instruction when the LOAD instruction has actually accessed or was certain to access data that has not yet been updated by the STORE instruction.	A load instruction is "in fact executed" when the load instruction actually has loaded data from a memory location.
#2: data speculation circuit	a circuit that detects data dependence between LOAD and STORE instructions and tracks execution of such instructions in order to detect any mis-speculations arising from the data speculative execution of LOAD instructions	a circuit that detects data dependence between load/store pairs and detects a mis-speculation by a load instruction that is in fact executed
#3: mis-speculation	where a LOAD instruction, that is dependent for its data on a STORE instruction appearing earlier in program order, is in fact executed before the STORE instruction	where an instruction has been in fact executed prematurely and erroneously
#4: where a data consuming instruction ... is in fact executed before the data producing instruction	where a LOAD instruction, that is dependent for its data on a STORE instruction appearing earlier in program order, has actually accessed or was certain to access data that has not yet been updated by the STORE instruction	where a load instruction that depends on a store instruction has actually loaded data from a memory location before the store instruction has put data into that same memory location
#5: predictor	a circuit that receives a mis-speculation indication from the data speculation circuit to produce a prediction	a circuit that receives a mis-speculation indication from the data speculation circuit to produce a prediction based on historical mis-speculation indications
#6: mis-speculation indication	an indication that the data speculative execution for a LOAD instruction was incorrect.	an indication that an instruction has been executed prematurely and erroneously
#7: prediction	a dynamic multibit value which indicates the likelihood that the data speculative execution of a LOAD instruction will result in a mis-speculation	a value indicating the likelihood that data speculative execution of a load/store pair will result in a mis-speculation
#8: prediction associated with the particular data consuming instruction	a dynamic multibit value associated with the particular LOAD instruction which indicates the likelihood that the data speculative execution of that LOAD instruction will result in a mis-speculation, and which is based on historical mis-speculation indications	a value associated with the particular load instruction that indicates the likelihood that data speculative execution of a load/store pair will result in a mis-speculation, and which is based on mis-speculation indications
#9: a prediction threshold detector	a circuit that prevents data speculation for a particular LOAD instruction when the prediction associated with said LOAD instruction is in a predetermined range	a circuit that prevents data speculation of a load/store pair if the prediction value for the pair is within a predetermined series of multiple values

Even if the parties' proposed construction accurately reflect the plain-and-ordinary meaning, it is often unnecessary to provide a construction as claim construction is "not an exercise in redundancy."<sup>7</sup>

Second, the parties' proposed constructions improperly limits the claimed invention to predicting potential memory dependences when the claim language does not limit the claim in that manner. More specifically, Claim 1 recites:

- 1) In a processor capable of executing program instructions in an execution order differing from their program order, the processor further having a data speculation circuit for detecting data de-

pendence between instructions and detecting a mis-speculation where a data consuming instruction dependent for its data on a data producing instruction of earlier program order, is in fact executed before the data producing instruction, a data speculation decision circuit comprising:

- a) a predictor receiving a mis-speculation indication from the data speculation circuit to produce a prediction associated with the particular data consuming instruction and based on the mis-speculation indication
- b) a prediction threshold detector preventing data speculation for instructions having a prediction within a predetermined range.

**TABLE 2.** List of additional disputed claim terms in the *Apple* Case, and the parties' competing constructions.

Term	WARF's Proposed Construction	Intel's Proposed Construction
#1: the prediction table	Not indefinite, no construction required	Indefinite pursuant to § 112, ¶ 2
#2: the certain data consuming instructions	Not indefinite, no construction required	Indefinite pursuant to § 112, ¶ 2
#3: the certain data producing instructions	Not indefinite, no construction required	Indefinite pursuant to § 112, ¶ 2
the synchronization table	Not indefinite, no construction required	Indefinite pursuant to § 112, ¶ 2
the flag value	Not indefinite, no construction required	Indefinite pursuant to § 112, ¶ 2

Claim 1 only recites that the speculation circuit detects a data dependence between instructions, i.e., it does not require detecting a dependence between *memory* instructions, and detecting a mis-speculation between a data producing instruction and a data consuming instruction, i.e., not necessarily between a store and load, respectively. As such, the dependence in Claim 1 is not limited to memory instructions, but rather could encompass all types of instructions including arithmetic (integer and floating-point) and logical instructions, and even potentially branch and other control instructions. But the parties' proposed constructions improperly limit the claim terms to memory instructions only. For example, for "data speculation circuit," WARF's proposed construction ("a circuit that detects data dependence between LOAD and STORE instructions and tracks execution of such instructions in order to detect any mis-speculations arising from the data speculative execution of LOAD instructions") improperly limits the scope of this claim term to circuits that only track memory dependences, while Intel's proposed construction ("a circuit that detects data dependence between load/store pairs and detects a mis-speculation by a load instruction that is in fact executed") also improperly does likewise. While the description in the specification may be limited to memory instructions, it is improper to limit a claim term to the examples in the specification.

*Third*, the terms in Table 1 appear in independent Claims 1 and 9. As its name implies, an independent claim is a claim that does not depend on another claim. By contrast, a dependent claim is a claim that depends one or more other claims and adds one or more limitations to the independent claim(s). Intel proposed that all terms in Table 1 should be construed (see pages 23–45 of *Wis. Alumni Rsch. Found. v. Intel Corp.*<sup>8</sup>). The likely reason it did so was, because under its proposed constructions, the accused products would not infringe the independent claims. And because dependent

claims depend on the independent claims, if the accused products do not infringe the independent claim, they also do not infringe the dependent claims.

*Fourth*, for the terms in Table 2, Apple's proposed constructions are that these terms are indefinite. A term is indefinite if a POSITA would not understand the scope of the claim term with "reasonable certainty."<sup>9</sup> If a court concludes that a term is indefinite, then all claims that that term appears in are invalid and thus cannot be infringed.

The next article in this series will describe the parties' arguments for each term in Tables 1 and 2, the courts' analyses and reasoning, and my analysis.

## REFERENCES

1. *Wis. Alumni Res. Found. v. Apple Inc.*, No. 3:14-cv-00062, ECF No. 24, Apr. 21, 2014 (W.D. Wis.).
2. C. Laser, "Scientific educations among U.S. judges," 2024. [Online]. Available: [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=4764194](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4764194)
3. J. P. Kesen and G. G. Ball, A Study of the Role and Impact of Special Masters in Patent Cases, *Federal Judicial Center*, 2009.
4. *Wis. Alumni Res. Found. v. Apple Inc.*, No. 3:14-cv-00062, ECF No. 127, Mar. 27, 2015 (W.D. Wis.).
5. *Wis. Alumni Res. Found. v. Intel Corp.*, No. 3:08-cv-00078, ECF No. 65, Sep. 18, 2008 (W.D. Wis.).
6. *Omega Eng'g, Inc. v. Raytek Corp.*, 334 F.3d 1314, 1323, 2003 (Fed. Cir.).
7. *U.S. Surgical Corp. v. Ethicon, Inc.*, 103 F.3d 1554, 1568, 1997 (Fed. Cir.).
8. *Wis. Alumni Res. Found. v. Intel Corp.*, No. 3:08-cv-00078, ECF No. 33, Sep. 18, 2008 (W.D. Wis.).
9. *Nautilus Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 910, 2014.

**JOSHUA J. YI** is a solo practitioner at The Law Office of Joshua J. Yi, PLLC, Austin, TX, 78750, USA. Contact him at [josh@joshuayipatentlaw.com](mailto:josh@joshuayipatentlaw.com).