

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ARM LTD. and MEDIATEK INC.,
Petitioner,

v.

DAEDALUS PRIME LLC,
Patent Owner.

IPR2025-00207
Patent 8,984,228 B2

Record of Oral Hearing
Held: June 25, 2026

Before GREGG I. ANDERSON, ARTHUR M. PESLAK, and
MARY C. HOFFMAN, *Administrative Patent Judges*

APPEARANCES:

ON BEHALF OF THE PETITIONER:

KEVIN P. ANDERSON, ESQ.
PATRICK D. MCPHERSON, ESQ.
Duane Morris LLP.
901 New York Avenue NW
Suite 700 East
Washington, DC 20001-4795
(202) 776-5213
kpanderson@duanemorris.com

ON BEHALF OF THE PATENT OWNER:

TAREK N. FAHMI, ESQ.
Ascenda Law Group, PC.
2150 N First Street
Suite 420
San Jose, California 95131
(408) 799-0612
tarek.fahmi@ascendalaw.com

The above-entitled matter came on for hearing on Thursday, June 25, 2026, commencing at 1:02 p.m. EDT, at the U.S. Patent and Trademark Office, 600 Dulany Street, Alexandria, Virginia.

P R O C E E D I N G S

- - - - -

1:02 p.m.

JUDGE PESLAK: I'll be keeping track of the time for the hearing on the clock behind me. For the record, there's a public line open and members of the public may be listening in. I'll note that neither party informed us that confidential information will be discussed today, so the hearing will be open to the public for the entire time.

As set forth in the hearing order, each side will receive 45 minutes of argument time total, including any rebuttal. With that, we're ready to begin. Counsel for Petitioner, come up to the podium. And how much time would you like to reserve for rebuttal?

MR. ANDERSON: Yes, Your Honor. I'd like to reserve 15 minutes for rebuttal, if I could.

JUDGE PESLAK: Okay. You have 30 minutes, and you can start when ready.

MR. ANDERSON: Okay, may it please the Court. My name is Kevin Anderson. I'm here on behalf of the Arm, Petitioner. What we have here is we have a patent that covers a particular way of sharing and managing the core, managing the cache for a shared cache. And what I'd like to do, if I can at the outset, is I want to explain -- go through the claims, explain what the claims say, and then go through the specification portions that support the claims, and then I'll turn to the prior art and the specific arguments that the Patent Owner has made.

Okay. So on the screen, I have our demonstrative slide 9, which is recitation of claim 1 of the '228 patent. And to just step through what each

1 of these components are, first component's a multi-core processor with a
2 plurality of cores. Second component is a shared cache memory. I don't
3 think there's any dispute that the sharing is that the plurality of cores share
4 the cache memory, meaning that each of those cores can access and use the
5 shared cache memory.

6 Third component is an integrated input-output module, which
7 basically is a component that allows for IO devices to access the memory of
8 the core, including the shared cache. The fourth component, and the one
9 which is primarily at dispute here, is a cache agent to perform cache
10 coherency operations for the plurality of core and the IO module. I'll stop
11 there, explain what that is. It's the cache agent is managing the shared
12 cache. That's the one cache that's referenced before to ensure cache
13 coherency, meaning that the cores, when they go to use a particular cached
14 value, would all receive the same value.

15 The second clause in red is that the cache agent, and the word "is" is
16 missing there, but the cache agent is a single caching agent for the multi-
17 core processor, and including a plurality of distributed portions, each
18 associated with a corresponding one of the plurality of cores. What this is
19 describing is a logical construct, the cache agent, whereby the collection of
20 the portions, the distributed portions, are managing the cache coherence for
21 the shared cache memory. That's the critical piece, that they're managing it
22 together for the shared cache memory.

23 Now if we look at the specification of the '228 patent, and I'm now
24 switching to the '228 patent, and I'm going to just go down to column 2 at
25 line 66 over to column 3 at line 3. This is one of the two places in the
26 specification where it discusses the distributed cache. And it notes that,

1 starting on line 67 of column 2, understand that a caching agent can be
2 distributed such that each of different portions of the caching agent can be
3 associated with a corresponding core and LLC, which means last level
4 cache, bank or slice. Now bank or slice is a portion of the last level cache.
5 Beyond that, there's no guidelines or restraints on what distributed means.

6 The other passage that discusses the distributed cache is in column 6,
7 starting at line 20, where it refers to figure 6, which I will show in a minute.
8 In column 6, it also notes that starting on line 22, as shown in the
9 embodiment of figure 6, processor 700 includes a distributed configuration
10 having partitions or slices, each including a core and a partition of caching
11 agent 715 and LLC 720.

12 Note that while distributed caching agents are shown, understand that
13 these distributed portions form a single caching agent, and which is
14 configured to handle cache coherency operations both for the cores as well
15 as the IO modules. Again, that is only -- there's only those two passages
16 discussing the term distributed with respect to caching agents.

17 Now, before I turn to figure 6, I just want to know what the patent
18 says about the particular type of caching agent, a particular type of
19 coherency operations that may be used within the '228 patent. It does not
20 require or specify any particular coherency operations. What it says, starting
21 on column 2 at line 60, although not shown for ease of illustration,
22 understand that caching agent may further include various coherency logic
23 to perform cache coherency operations on behalf of the cores and IO
24 module, and to enable coherent access with respect to both the core
25 transactions and IO device transactions as described herein.

1 What that's saying is the coherency, the type of coherency operations
2 that are encompassed within the '228 patent, as well as the challenged
3 claims, can use any form of coherency logic. That language does not
4 exclude a particular form of coherency logic and it does not require a
5 particular form of coherency logic. And nor does the claim. The claim
6 doesn't have any language that requires or excludes them. Now --

7 JUDGE PESLAK: But counsel, you started this by saying your
8 caching agent's a logical construct.

9 MR. ANDERSON: Yes.

10 JUDGE PESLAK: So from that, I take that it's not like a particular
11 device that's a controller, right?

12 MR. ANDERSON: Correct.

13 JUDGE PESLAK: So let me ask you a question about what the
14 Patent Owner says about this. So Patent Owner says the single caching
15 agent should be understood as a logically unified entity. That is a cohesive
16 caching agent where all distributed portions are connected. Can you tell us
17 why that's wrong?

18 MR. ANDERSON: I don't disagree that it's a single distributed entity,
19 and I have it in front of me right now. I have demonstrative slide 6, which is
20 figure 6 of the '228 patent as we annotated. This comes from our petition as
21 we've annotated here. And one can see that each of the different cores,
22 which in this instance is 8027 have a core, a distributed caching agent
23 portion, and then also a portion of the last level cache that's associated with.

24 So collectively, those components are managing the last level cache.
25 What is being distinguished in the patent is the situation where each of these
26 caching agents would attempt to manage the entire cache on their own,

1 meaning they would all have effectively copies of each other and they would
2 all be controlling the entire cache. What the claims recite, what's shown in
3 figure 6 and what's described in columns 2 to 3 and in column 6, starting at
4 line 20, is a configuration where the caching agents each work together to
5 manage the overall cache.

6 JUDGE PESLAK: So let me just ask another question, because the
7 second part of this limitation is that the plurality of distributed portions each
8 associated with the corresponding one of the plurality of cores.

9 MR. ANDERSON: Correct.

10 JUDGE PESLAK: What's the Petitioner's position as to what
11 associated means there? And if I'm looking at figure 6, how do I know what
12 portion is associated with what part of the core?

13 MR. ANDERSON: So our position on this is that figure 6 shows the
14 associated with, and that that term associated with, with respect to what's in
15 the claim, is mentioned only twice in the specification. It's mentioned at
16 column 2, line 53 to 55, and it's mentioned at column 3, lines 1 to 3. And in
17 both instances, it just uses the language. It doesn't put any particular
18 restriction on associated with. And so, in our view, associated with means
19 any association. It doesn't have to do a particular -- it doesn't have -- there's
20 no restrictions on associated with.

21 JUDGE PESLAK: Okay. So when you get to Agarwal, tell me how
22 it's associated in Agarwal, because that's a little unclear to me.

23 MR. ANDERSON: Okay. And actually, I can answer that just right
24 immediately. Agarwal uses exactly the same language, exactly associated
25 with. It says that, and it has it in the claims and it has it in the specification.
26 And so, if one would turn to Agarwal, I can just turn to the claim. I'll turn

1 down to claim 9 of Agarwal. Agarwal says, at least some of the directory
2 controllers are associated with a processor core. And the directory
3 controllers are the caching agent in Agarwal. So Agarwal has exactly the
4 same language in the claims.

5 And then, if I would go up to column 3, unsurprisingly -- I'm sorry,
6 column 3. At least in column 3, at lines 42 to 44, it says the same as what it
7 claims. At least some of the directory controllers are associated with a
8 processor core, which matches the exact language.

9 So in order for Patent Owner's position to prevail on associated with,
10 the Court would have to find that the '228 patent in the two passages where
11 it mentions associated with and just repeats the language somehow has an
12 entirely different meaning than is not encompassed by Agarwal when it uses
13 exactly the same language. And I don't think that that's remotely the case.

14 And just to give an example, if I go back to my slides, you'll see that -
15 - I'm on demonstrative slide 6 showing figure 6 from the '228 patent. This is
16 an example, in fact, it's the only example in the '228 patent of a caching
17 agent being associated with a core. And what it's showing here is that
18 whether through physical or connection, there is a one-to-one core and
19 caching agent relationship here for at least some of the cores. It's not
20 required for all by the claims.

21 Now, if I go down to slide -- okay, I'm sorry. Down to demonstrative
22 slide 7, which is showing the annotated versions of Agarwal that we had in
23 our brief, in our petition, on the left-hand side, you'll see that in green, there
24 is a multi-core processor where each of those cores is identified in Agarwal
25 calls the cores tiles. They're the same thing as cores. And so you see where

1 the '228 had an example of eight cores, you see Agarwal's figure 2 shows an
2 example of nine cores in a processor.

3 On the right-hand side, you'll see a blowout of each of those cores.
4 And it shows in a manner virtually identical to figure 6 of the '228 patent
5 that each core has a directory controller, which is the caching agent. And
6 each core has the L3 cache, which is exactly identical to the LLC cache in
7 figure 6 of the '228 patent. So the configuration for associated with not only
8 uses the exact same language, but is depicted in a functionally identical way
9 in Agarwal as it is in the '228 patent.

10 Now, I should note, and I'll just turn to Agarwal in particular.
11 Agarwal has 30 columns of the details of how one would actually implement
12 this. The '228 patent has seven columns with really only five lines
13 describing how the distributed portion works. But those five lines in the
14 '228 patent do not exclude anything. They basically say, we're not going to
15 teach you how we do cache coherency because we're relying on what's
16 already known in the art. And Agarwal is one of those things that's already
17 known in the art.

18 With respect to the distributed portions, I think that one of the primary
19 arguments by Patent Owner is that there must be some centralized controller
20 managing those distributed portions. They say that throughout their brief. If
21 one looks at the '228 patent, there is no teaching of a centralized controller.
22 And if one looks at the only embodiment in figure 6, there is no centralized
23 controller. That is simply not there. That is something that they've
24 manufactured up with attorney argument.

25 I will note while I'm showing, and I'm showing right now for the
26 record, back to figure 6 of the '228 patent. It's very telling that in their

1 response brief and their sur-reply and their expert opinion, they don't cite to
2 or discuss figure 6 a single time, which is shocking because that's the only
3 embodiment showing the distributed configuration. And I think they avoid
4 that because figure 6 depicts a distributed configuration that is so close to
5 what is in the prior art.

6 Even after in our reply brief we noted that they had not discussed
7 figure 6 anywhere, in their sur-reply brief, they avoided figure 6 as well.
8 But figure 6 is the sole embodiment. And what they're basically trying to do
9 is manufacture a variety of new limitations, including this centralized
10 controller limitation, and the other limitations related to distributed that don't
11 exist in the '228 patent. They don't exist in the claims there, and so there's
12 no basis to manufacture those and read those into the claims.

13 I would also note that their other, I think, primary argument is that it's
14 basically a complaint that Agarwal actually has the details that they don't
15 have. They talk about Agarwal's individual components using message
16 passages or requests to communicate the coherence information among each
17 other. First of all, the claims don't exclude that. And second, that's the sort
18 of detail that probably should have been in the '228 patent, but it's not in
19 there. And even the specification of the '228 patent doesn't exclude that.

20 What Agarwal's system does, and just switching to slide 7, is that
21 these directory controllers, each of them manages the cache for that
22 particular tile, that particular core. And the way that works is the directory
23 controller is managing the L1 and the L2 caches, level 1 and level 2, which
24 are private. And then the L3, which is the shared cache, which is
25 corresponding to the shared cache of memory, is distributed with each of the

1 different cores having different portions of that, just like in the '228 patent,
2 where it says that the cores have their own bank or slice.

3 And what happens then in Agarwal, when one of the cores decides it's
4 accessing a memory, it will look in its own private L1, it doesn't find it. It
5 looks in its own private L2, doesn't find it. So, it knows it has to go to L3.
6 Agarwal describes this system where each of the directory controllers has a
7 common algorithm for identifying where to find that particular memory
8 address in L3, which of the cores has it.

9 Now that algorithm is described, it could be a hash table, it could be
10 an address mapping table, could be any number of different algorithms. But
11 Agarwal's tile knows that I need to go for this address, I'll go find it out
12 there. And that's how they're collectively managing the shared cache. That's
13 the important piece, is that each tile is responsible for a portion of shared
14 cache.

15 And the collection of tiles, distributed portions, which Agarwal
16 describes itself as distributed throughout, the collection of those is managing
17 the shared cache, which of course is what the claim requires. So, it's the
18 management of the overall cache. And so we have individual portions,
19 which have to work together and are distributed among the different tiles or
20 the different cores to manage the cache. I'll stop there. And if there's any
21 questions, I'll continue on.

22 JUDGE PESLAK: Do you have any questions, Judge Anderson?

23 JUDGE ANDERSON: Too many screens open. No, I do not have
24 any.

25 JUDGE PESLAK: Okay.

1 MR. ANDERSON: Okay. I'll move on to some of the specific or
2 other arguments we made. I've already addressed the argument about
3 associated with and where Agarwal has not only the identical language in its
4 spec and in its claims, but also describes the identical configuration of the
5 caching agent being located with the core there.

6 They also, turning to the dependent claims, they make an argument
7 that there has to be a single table of request. The claim doesn't say that, and
8 the specification doesn't describe that. And in fact --

9 JUDGE PESLAK: It does -- that's claim 2 now, right?

10 MR. ANDERSON: Claim 2, yes. What claim -- what is described for
11 a table request, a TOR or a table request is that, and this is in column 2, line
12 47, TOR may be a buffer or other temporary storage for holding incoming
13 requests. Now, if I go and look at figure 2, I will see in figure 2 in the
14 middle is TOR 220. What figure 2 is is it is describing the caching agent as
15 a whole. And so what the claim requires though is the caching agent have
16 distributed portions on each core. So the claim at least encompasses, if not
17 requires, that the portions -- each of the portions shown in figure 2 be
18 distributed among each of the cores. So there would be a TOR in each of the
19 cores.

20 What is not described anywhere in the specification and is not
21 required by the claim is that somehow there would have to be a single TOR
22 for all of the cores. And in fact, if one looks at figure 6, which is showing
23 the overall processor, there is no single TOR.

24 The other thing I will note is that in the sur-reply, the Patent Owner
25 took the position that the TOR must perform arbitration. They said that
26 repeatedly in their sur-reply. That is demonstrably false. The Arbiter, 215

1 there, which is a separate component, performs the arbitration. As I showed,
2 and I'll turn back to the specification, I'm just discussing these again back in
3 column 2. In column 2, starting at lines 44, the arbiter can arbitrate among
4 incoming requests and provide them to the TOR.

5 The TOR is just a buffer storage. Very, very clear that their sur-reply
6 argument about the TOR needing to arbitrate is just contrary to what's in the
7 specification there. And they don't describe any sort of configuration. I'll
8 note that -- I'll stop. Do you want me to just -- I'll stop here.

9 JUDGE PESLAK: Judge Anderson is having problems with his
10 computer today. We may have to get him back on the line.

11 MR. ANDERSON: Okay.

12 JUDGE PESLAK: So let me just stop the time.

13 JUDGE ANDERSON: Hey, I think I'm here. Can you all hear me?

14 JUDGE PESLAK: We can hear you, yes.

15 JUDGE ANDERSON: Okay. I will just -- let me just say for the
16 record that I don't have -- apparently, I have bigger problems than I thought I
17 did, and I can't even get my local drive going. So be aware that I will not be
18 able to see the demonstratives. However, I am extremely familiar with the
19 record, and once we have the transcript, I will make sure to accommodate
20 any issues that that might raise. So, with that, I hope everybody can
21 proceed. And I will listen intently, and let's go ahead.

22 JUDGE PESLAK: Okay. Thank you, Judge Anderson. So let's just
23 all make sure when we're referring to a paper, demonstrative, whatever, that
24 we particularly -- the particular section in the record, and then when we get
25 the transcript, we can put it all together. But anyway, you have about four
26 minutes left in your opening time. So continue, please.

1 MR. ANDERSON: Thank you, Your Honor. I will turn to their
2 argument about the claim that requires storage into the cache without a
3 memory transaction, which I believe is, for the record, is claim 12, for
4 example. Agarwal describes two particular configurations. In one
5 configuration, it says expressly, and shows expressly, that the write to the
6 cache is then forwarded on to the memory.

7 In the other configuration, it shows a write to the cache which does
8 not go on to memory. And those are shown in figures 9b, which shows the
9 write to the cache goes out to memory, and figure 9c, which shows that there
10 is a write to cache with no corresponding access to memory.

11 Patent Owner's position is that because Agarwal didn't detail all of the
12 things that it didn't do, for example, it didn't expressly say we don't write to
13 main memory, that such a negative limitation is not disclosed. Our position
14 is the juxtaposition of the two embodiments, one that expressly shows a
15 write to main memory and one that does not disclose the lack of the write to
16 main memory for that particular claim.

17 I would say also, if one looks at figure 9a in Agarwal, there is a
18 question raised by Patent Owner that there is no -- I'm not exactly certain,
19 but it has to do with the forwarding of the transaction in claim 11, I believe.
20 I'll note that 913 in figure 9a is described as an IO interface, with specific
21 examples of network ports given out there. 912 is the IO module that
22 receives that cache access.

23 And then as we've described, and as is throughout Agarwal, 912 then
24 uses its mapping table to determine where to look for, where I should look
25 for that, which of the tiles has that particular access, and then forwards that
26 request to the appropriate tile for the cache coherence operations that are

1 described throughout our brief, and I don't think are really seriously
2 challenged.

3 Now, I'll just close with this last minute by noting that our second
4 reference, which I haven't gone into detail, Sinharoy, is identical to the
5 reference which the Board found invalidated the trial patent.

6 JUDGE PESLAK: Actually, I wanted to ask a couple of questions --

7 MR. ANDERSON: Yeah, certainly.

8 JUDGE PESLAK: -- before your time.

9 MR. ANDERSON: Sure.

10 JUDGE PESLAK: So I know you mentioned in your reply brief that
11 collateral estoppel may apply, but you didn't really explain much.

12 MR. ANDERSON: Sure. Yes, Your Honor.

13 JUDGE PESLAK: I look at this, there's sort of three issues involved
14 here. One is we construed single caching agent there. And it's a related
15 patent, and it's a very similar claim, a substantially similar claim limitation.
16 The second thing is we found that Sinharoy disclosed a single caching agent
17 under that construction.

18 And the third thing is I'm not sure I agree with you, is that claim 1 in
19 that patent is substantially identical, because it has some different -- and the
20 patent involved in the 1344 IPR had some additional limitations as opposed
21 to claim 1 in '228 patent. But it seems like an argument can be made for a
22 collateral estoppel on all those issues.

23 MR. ANDERSON: Yes, Your Honor.

24 JUDGE PESLAK: What's your position on that? I'm going to ask
25 Patent Owner's counsel the same thing.

1 MR. ANDERSON: Sure. So, what I would say is as we showed in
2 our expert report, and we referenced in footnote one of our petition on the
3 first page, is that the claims of the '228 patent, every element of that is in the
4 claim 1 of the '895 patent. The '895 patent also has some additional
5 elements which are found in some of the dependent claims of the '228
6 patent. So by invalidating claim 1 of the child '895, necessarily, all of the
7 elements were found in the '228 claim 1, because they're all there. With
8 respect to estoppel, it's a, I would say --

9 JUDGE PESLAK: I mean, it's a final judgment now, because the
10 Patent Owner didn't appeal it. That's the only reason I'm asking.

11 MR. ANDERSON: Yeah, the Patent Owner didn't appeal it, and so I
12 know there's different forms of estoppel. There's statutory estoppel and what
13 maintain means, and whether that includes the PTAB or only prosecution. I
14 would say we don't need this panel to make new law in terms of writing
15 some opinion that estoppel applies.

16 JUDGE PESLAK: The Board has applied collateral estoppel before.

17 MR. ANDERSON: Okay. In that instance, I don't see how the
18 standard principles of collateral estoppel don't apply. It's exactly the same
19 issues on exactly the same prior art. They didn't address it in their response
20 brief. They did address it in their sur-reply brief and basically said, well,
21 there's different issues and different records, but they don't really develop
22 that at all.

23 And even if there is different issues and different records, the whole
24 point of collateral estoppel is that one doesn't get a second bite of the apple.
25 Otherwise, everyone would have the incentive to effectively have identical

1 claims and different patents and just keep coming back to the Board and
2 trying to relitigate it as they're doing here.

3 JUDGE PESLAK: Okay.

4 MR. ANDERSON: Did that answer your question?

5 JUDGE PESLAK: Yes.

6 MR. ANDERSON: Okay. Your Honor, any more questions?

7 JUDGE HOFFMAN: No questions. Thank you.

8 MR. ANDERSON: Okay. Thank you.

9 JUDGE PESLAK: I think you went a couple minutes over. But if
10 you need a few minutes, we can do that.

11 MR. ANDERSON: That's fine. I should be fine. Thank you.

12 JUDGE PESLAK: Patent Owner Mr. Fahmi, you're up. How much
13 time would you like to reserve for rebuttal?

14 MR. FAHMI: Thanks, Your Honor. I'd like to reserve ten minutes,
15 please.

16 JUDGE PESLAK: Okay. So you have 35 minutes. You can start
17 when ready. And I didn't mean to sandbag either party on the collateral
18 estoppel issue. So, if you want -- I don't want to hijack your argument, Mr.
19 Fahmi. So if you want, you can address it as you see fit.

20 MR. FAHMI: Thanks, Your Honor. May it please the Board. Let me
21 jump right into it for you, since that was most recently on your mind. You
22 should reject Petitioner's claim for collateral estoppel. Basically, what
23 they're asking this board to do is substitute the requirement that they prove
24 their case.

25 The starting point for all this should be the decisions first by the
26 Supreme Court in *SAS v. Iancu*. And secondly, by the Federal Circuit in

1 *Magnum Oil Tools*. Together, those cases clarify that the Board needs to
2 decide the case in front of it on the current proceeding, and that the
3 Petitioner bears the burden of proof. The Petitioner can't use later filings
4 such as a reply to substitute for anything in their petition. And it's
5 incumbent upon them to set forth the case in the petition.

6 The Federal Circuit has spoken about possible preclusive effects of
7 decisions in related patents or similar claims. We cited in our sur-reply, the
8 *Novartis* case, and that stands for the proposition that the claim language
9 must be analyzed in the context of the specific claim under review and the
10 evidence presented in the particular proceeding.

11 Another case that is instructive is *Kroy IP Holdings*. That's 127 F.4th,
12 1376, last year from the Federal Circuit. And that involved claims in the
13 same patent that had not been subject to adjudication. And it was improper
14 for the Court to then apply collateral estoppel to those claims, even though
15 they're in the same patent.

16 Now, here, we're dealing with a continuation, similar specification,
17 similar claims. But if collateral estoppel doesn't apply to the claims having
18 the same terms in the same patent, it certainly can't apply across patents,
19 even if they're in the same family.

20 JUDGE PESLAK: Can I ask you one question? Something you said
21 in the beginning. Are you saying that Petitioners should have addressed
22 collateral estoppel in the petition?

23 MR. FAHMI: Well, for one, they needed to have addressed it in order
24 to make it part of their case. But even if they had --

25 JUDGE PESLAK: I was just asking procedurally if that's what you're
26 saying.

1 MR. FAHMI: Absolutely. Yes, absolutely. Now, Petitioner in its
2 opening presentation, confined itself to a discussion of Agarwal. They didn't
3 discuss the Sinharoy reference, at least on the merits. So I'll do likewise and
4 just address the Agarwal reference.

5 And, Your Honors, we don't have to look far to understand the
6 distinctions in Agarwal. We all seem to agree about what claim 1 requires in
7 terms of the single caching agent being a logically unified entity. I think I
8 heard Mr. Anderson say that he agreed with that. We disagree about what it
9 means for the distributed portions and the associated with language.

10 The Petitioner's position, if I understand it correctly, is that mere
11 physical proximity is sufficient. Our position has been that mere physical
12 proximity or co-location is not what is being discussed in the claim, and that
13 the associated with limitation in the claim means that a particular distributed
14 portion performs cache coherence for its associated portion of the shared
15 cache. How do we get there? Well, one, we have the testimony of Dr.
16 Bagherzadeh, but you can look at the specification of the '228 patent itself.
17 I'd refer you to the background portion.

18 JUDGE PESLAK: Let me ask you this question, because, you know,
19 Petitioner is referring to figure 6 in the '228 patent. So is it Patent Owner's
20 position that, if I look at figure 6 that, for example, we have core 710-0, we
21 have caching agent 715-C next to that. Is that associated? And how do I
22 know those two are associated, if that's Patent Owner's position?

23 MR. FAHMI: Well, the text says they're associated. And what the
24 text means is that that portion of the single cache controller performs cache
25 coherency for that portion of the LLC. And we know that that's how
26 associated with is used. Because in the background section, the '228 patent

1 explains, and I'm referring specifically to column 1, line 31, that one of the
2 problems --

3 JUDGE PESLAK: You may want to hold on one second. I think
4 Judge Anderson might be able to get back on. Sorry to interrupt.

5 MR. FAHMI: That's all right.

6 JUDGE ANDERSON: Let me just say I'm up. I'm up at least
7 momentarily. So if I get lost again, I will just stick to the phone and not
8 interrupt any further. Please --

9 MR. FAHMI: Thank you, Your Honor. Are you hearing me okay
10 then?

11 JUDGE ANDERSON: I can.

12 MR. FAHMI: Thank you. As I was mentioning, we know how
13 associated with is used in the patent because in column 1, line 31, one of the
14 problems that's being addressed by this solution is that there was no single
15 cache controller that was also handling the IO, and that a separate cache
16 controller was associated with the IO and performing cache coherency for
17 the IO. That's how associated with is used in this patent. It's the entity that's
18 performing cache consistency, whether it be for the IO device or for a
19 particular core.

20 And that's why Dr. Bagherzadeh indicated that's how the person of
21 ordinary skill in the art would understand it. This is in paragraph 47 of his
22 declaration. So what does --

23 JUDGE PESLAK: Actually, I want to ask you a question about Dr.
24 Bagherzadeh. In terms of your claim construction, in the briefing, you use
25 this term and it seems to be a term that Dr. Bagherzadeh -- is that it?

26 MR. FAHMI: I believe that's right.

1 JUDGE PESLAK: Okay. He testified in paragraph 33, he says
2 logically unified entity. I don't see that term anywhere in the '228 patent.
3 And is it fair to say that his interpretation of these various provisions and the
4 specification that he says, that's terms not in the '228 patent?

5 MR. FAHMI: Yes, I think that's fair, Your Honor. That is his
6 opinion as an expert in the field. He's obviously been through the patent
7 many times and decided that contextually, this is how it would be
8 understood as the single logically unified entity. We know it's not
9 physically unified, otherwise it would make sense to talk about distributed
10 portions. But single has to mean something. So the only thing left is logical
11 unification.

12 And Petitioner wants to read single out of the claims because they
13 can't point to any logical unification, right? Agarwal certainly doesn't
14 provide it. Agarwal, as you heard Mr. Anderson explain, uses separate
15 directory controllers, each with an assigned memory region. Those separate
16 controllers handle cache coherency for their assigned memory regions
17 independently. They don't consult with other directory controllers. Why
18 would they? The other directory controllers are responsible for different
19 memory regions.

20 Dr. Bagherzadeh explains this concept in the home tile concept, the
21 way it's referred to in Agarwal, in paragraphs 41 and 42 of his declaration.
22 And it's a distinguishing characteristic from Agarwal. In Agarwal, we saw
23 that if a tile, let's say it's tile A, is seeking data at a particular cache line and
24 it doesn't find it in its own local cache, it's forced to go to the directory
25 controller at a different tile that has responsibility for that address region.

1 And it's that different directory controller that will end up supplying the
2 necessary data that's being requested. Now --

3 JUDGE PESLAK: Let me ask you this. I like to look at the figures in
4 the patent, okay? And figure 6. What collectively in figure 6 is the single
5 cache image?

6 MR. FAHMI: You're talking about the '228 patent?

7 JUDGE PESLAK: Yeah, '228 patent. I'm sorry.

8 MR. FAHMI: Yeah, it is the sum total of all the individual distributed
9 portions that are shown in figure 6.

10 JUDGE PESLAK: So all the 715s?

11 MR. FAHMI: Yes.

12 JUDGE PESLAK: Okay. And the association then is with the core
13 that's next to it?

14 MR. FAHMI: That's correct. It's handling the caching operations for
15 that core.

16 JUDGE PESLAK: And then there's a separate last-level cache for
17 each core?

18 MR. FAHMI: I believe the way that's described in the patent, Your
19 Honor, is that the last-level cache is a shared last-level cache. They show,
20 the figure 6 shows a partitioning that might exist, but it's a shared cache.

21 JUDGE PESLAK: Okay. So let's jump to Agarwal then. Petitioner's
22 counsel put up figure, I guess it's 6a or 6b, of Agarwal. And can you tell me
23 what's wrong with the mapping that they did for Agarwal to claim 1?

24 MR. FAHMI: Well, figure 6a and 6b are showing what I was just
25 describing. One of the tiles, I think it's the tile on the right of the figure, is
26 looking for some data. It consults its local cache. The data is not there. The

1 local cache is forced to go to the home tile, that's the one on the left, in order
2 to ask for the data.

3 It's the tile on the left, the home tile, that is performing the acts of
4 retrieving the data from memory, storing it in its left tile cache, managing
5 the coherency, all because it's the directory controller that has been assigned
6 this address region. And it's then providing a copy back to the original
7 requesting tile so that it can be used.

8 Now, the problem with that is the Petitioner is conflating a system of
9 distributed cooperation with one of logical unity. What you're seeing in
10 Agarwal is what you see in all distributed directory protocols. You're seeing
11 caches doing message passing, but you're not seeing a single unified entity
12 performing cache coherency for all of the cores.

13 And we know this because if there were some type of shared control
14 structure, even if it was distributed, but some type of shared control
15 structure, which you need for this logical unification, according to Dr.
16 Bagherzadeh, you wouldn't have the need for this kind of message passing.

17 And let me refer you, how do I know this? Why should you take my
18 word for it? Take a look at the '228 patent in column 6, lines 15 to 17.
19 What does it tell us? It tells us that the solution in the '228 patent, the
20 reason, the *raison d'etre* for a single caching entity is to eliminate the need
21 for divided tracking resources as you find across home tile structures as you
22 have in Agarwal.

23 JUDGE HOFFMAN: Could you repeat where that citation was?

24 MR. FAHMI: Yes, Your Honor. It's column 6, lines 15 to 17 in the
25 '228 patent.

26 JUDGE HOFFMAN: Thank you.

1 MR. FAHMI: All right. The specification is telling us that Agarwal's
2 distributed system is not the kind of system the '228 patent is directed to. In
3 fact, when you get right down to it, the architecture is really the opposite of
4 what the claims require. Agarwal has no unified control structure. It has no
5 shared decision-making, each directory of controllers acting autonomously
6 for its assigned memory region. And there's no one entity responsible for
7 coherence across the entire system. As I said, it's really just message
8 passing.

9 JUDGE PESLAK: So, if, just let me interrupt for one second. So
10 that's based on Patent Owner's claim construction, correct? If we were not to
11 adopt that, then is there some other issue with Agarwal that Patent Owner
12 could point to if we maintain the claim construction from institution decision
13 here?

14 MR. FAHMI: Well, your claim construction in the institution order
15 was that it's one caching agent for the processor, right? I think that's
16 consistent with what I'm saying. There is no one caching agent in Agarwal.
17 What you have is a collection of peer agents. They're not unified. There's
18 no single cache controller. There are many cache controllers. Independent
19 peers is I think how Dr. Bagherzadeh described them. So even if you don't
20 like the logical unity being part of the construction, I think you still have to
21 reach the conclusion that multiple peer controllers do not a single cache
22 controller make.

23 The Petitioner also had comments concerning the claims that include a
24 table of requests. These would be dependent claims 2 and 3, and also
25 independent claim 16, although Agarwal is not asserted against claim 16. So
26 let me speak to it in the context of claims 2 and 3.

1 The table of requests is a shared structure. How do we know that?
2 We know that because as Mr. Anderson pointed out, figure 2 shows what a
3 caching agent looks like, what a single caching agent looks like, and there is
4 one table of requests. It's not a situation where you have multiple different
5 tables. It is one table.

6 Now, I heard Petitioner's counsel say that we were arguing somehow
7 that each of the distributed portions had to have its own table of requests.
8 We've never made that statement, and that really wouldn't be our position.
9 Our position would be that while the caching agent may be made up of
10 distributed portions, the table of requests has to be a shared structure
11 amongst distributed portions.

12 And there's no such shared request table in Agarwal. Instead, each of
13 the individual directory controllers maintains its own local structure. Dr.
14 Bagherzadeh explains this in paragraphs 77 and 81 of his declaration.

15 And to the extent that the Petitioner is also citing Blumrich as
16 providing some kind of support for the idea of these tables, because
17 Blumrich describes a buffer, it's worth noting that the buffer that the
18 Petitioner points to does not share both IO and core transactions. It is
19 strictly course. And I don't think there's a dispute about that.

20 But regardless, one would, even if you adopt the use of buffers, which
21 I think is all that Petitioner is saying Blumrich should be looked for, or
22 looked to for, even if you adopt the use of buffers, you still have them at the
23 per-tile level. There's no indication that they would be shared amongst
24 different portions of a single cache controller, for there is no single cache
25 controller in Agarwal.

1 JUDGE ANDERSON: Counsel, I hate to interrupt, but I could lose
2 my computer at any moment here. So, I have a question, and I understand
3 that your demonstratives are not evidence. We say that all the time. But you
4 say this case, in your first demonstrative, you say that the case turns on
5 architecture, not labels. And let me ask you, initially, what do you mean by
6 that?

7 MR. FAHMI: Your Honor, what I'm referring to by that is that the
8 architecture of the single caching agent, rather than any labels, whether you
9 want to attach labels to it, as Petitioner does, trying to map it to Agarwal, is
10 what's important. So, in the case of the '228 patent, it is one caching agent
11 performing cache coherency for the entire processor, notwithstanding the
12 fact that it's made up of distributed portions. One caching agent.

13 Agarwal has the opposite. Agarwal has distributed caching agents.
14 That's an architectural distinction, and so that's what I'm referring to.

15 JUDGE ANDERSON: So, if we look at the '228 patent and figure 6,
16 and we can agree that that is part of the architecture of what is disclosed in
17 the '228 patent?

18 MR. FAHMI: Yes, Your Honor.

19 JUDGE ANDERSON: So, how does that differ from what is shown
20 in figure -- I want to go to figure 5 of Agarwal, which also shows a similar --
21 well, let me just say, that's an architecture too.

22 MR. FAHMI: Yeah, the difference lies in how those units are
23 operating, Your Honor. So, it's not just location. As I mentioned earlier, co-
24 location is not the distinguishing characteristic. It is the fact that the
25 distributed portions in the '228 are functioning as a single entity versus the
26 disparate cache controllers in Agarwal are not. So they are merely operating

1 independently, keeping one another advised of what they are doing through
2 the message passing. There's no --

3 JUDGE ANDERSON: Would you agree, though, that figures 5, 6a,
4 and 6b of Agarwal have the same architecture as figure 6 of the '228 patent?

5 MR. FAHMI: No, Your Honor, I would not. First of all, in figure 5,
6 you see that not all of the tiles even have a distributed caching agent. It's
7 only select ones. So, we're not talking about a situation where there's a one-
8 to-one mapping between all cores and caching agents. 6a and 6b are merely
9 showing the select two tiles, presumably somehow taken from that
10 arrangement on figure 5.

11 And here, again, what we're seeing is the caching agent in the left tile,
12 tile 1, is the sole caching agent that is acting for the requested data, and that's
13 a function of its assigned memory address range. So, no, I don't agree that
14 it's the same architecture.

15 JUDGE ANDERSON: Okay. Go ahead.

16 MR. FAHMI: Your Honors, I know I'm early. I'm happy to take
17 further questions. If you want me to address others of the references, I'm
18 happy to. But as I mentioned, since the Petitioner confined its remarks to
19 Agarwal and primarily claim 1, I've done likewise. As I say, the patent
20 itself, I think we looked in column 6, lines 15 to 19, provides all the
21 distinction you need in order to find the claims patentable over Agarwal.

22 JUDGE PESLAK: Can you go over your position on claim 2?

23 MR. FAHMI: Yeah. So, claim 2 speaks to the table of requests. As I
24 mentioned, it's a shared structure, and if you go to claim 3, it even includes a
25 dedicated entry for so-called posted transactions. And I think everybody

1 agrees that posted transactions are the ones that don't require an
2 acknowledgment.

3 With respect to claim 2, what does Agarwal say? Well, first, Agarwal
4 does not use any shared structure for a series of cache requests. Each
5 directory controller is operating independently. They're not storing any
6 requests related to anything outside their address range. So the concept of
7 the shared structure of the table of requests just doesn't exist because it's not
8 needed.

9 JUDGE PESLAK: So your position is that the '228 patent requires
10 one table of requests for all the cores?

11 MR. FAHMI: Well, I want to make sure -- we're not saying it has to
12 be a single physical table of requests. We've never made that argument. We
13 know it has to be shared, right? Because figure 2 shows one table of
14 requests for the single caching agent. Now, could it exist all in one place?
15 Yes, perhaps, but we know the caching agent has distributed portions.

16 And we're not precluding the table of requests from being distributed.
17 The patent doesn't say one way or the other. But even if it's distributed, it
18 has to be a single table of requests for the single caching agent that handles
19 coherency for the entire processor. So it's shared by the distributed portions.

20 JUDGE PESLAK: Can you go over again why Patent Owner like
21 you, Agarwal, doesn't need that?

22 MR. FAHMI: Well, Agarwal doesn't have it because Agarwal doesn't
23 need it. Looking at the example in figure 6, tile 1 handles the address range
24 that it's assigned. It doesn't need either itself or access to a table that has
25 caching requests related to address ranges that are outside of its assigned

1 region. It wouldn't make sense, right? It doesn't need to consult those when
2 it's making cache coherency decisions for its address range.

3 So somehow, as Dr. Bagherzadeh explained, I think between
4 paragraphs 77 and 81, each controller in Agarwal is maintaining its own
5 local structures and it's not relying on -- doesn't utilize any unified table for
6 storing requests simply because it's not necessary. And modifying Agarwal
7 in some fashion to accommodate that is not something a person of ordinary
8 skill in the art would do because it would simply introduce unnecessary
9 overhead, right?

10 Tile 1 is not going to be dealing with cache coherency outside of its
11 assigned address range. Why would it have to go and use a unified table that
12 includes transactions related to caching operations that are not within that
13 assigned address range? It would simply introduce unnecessary overhead.

14 JUDGE PESLAK: Okay, I think I understand your position. Judge
15 Hoffman, do you have any further questions?

16 JUDGE HOFFMAN: No questions. Thank you.

17 JUDGE ARTHUR: Judge Anderson, do you have any more questions
18 for Mr. Fahmi?

19 JUDGE ANDERSON: No, I do not. Thank you.

20 JUDGE PESLA: Okay. So, I think you have about 14 and a half
21 minutes for surrebuttal.

22 MR. FAHMI: Thank you.

23 JUDGE PESLAK: So, Mr. Anderson, I think you had about three
24 minutes left. But since I dragged you down the hole of collateral estoppel,
25 you have a couple of minutes. I'll give you six minutes for rebuttal.

26 MR. ANDERSON: I think I reserved 15 minutes, Your Honor.

1 JUDGE PESLAK: But I think -- oh, wait a minute. I'm sorry. Like
2 you reserved 15 minutes and we went three minutes over, so you have 12
3 minutes. I'm sorry.

4 MR. ANDERSON: Yeah. Thank you, Your Honor.

5 JUDGE PESLAK: My apologies. That's why I shouldn't be in charge
6 of this.

7 MR. ANDERSON: So let me just start with this issue of, you know,
8 what single means in the claims. And I'll turn to Claim 1. And what I'd
9 point out is, during their entire discussion, they did not point you to
10 anywhere in the '228 patent that describes any sort of architecture or how it
11 would work under their kind of exclusionary constructions.

12 And what I think the term single is read to mean is that it is it's with
13 reference to what the caching agent is doing. The caching agent and its
14 portions are managing the cache for the shared cache memory. The portions
15 are collectively managing the cache for the shared cache memory. I can't
16 emphasize that enough. That's what the claim requires, is managing for the
17 shared cache memory.

18 That's exactly what Agarwal's portions are doing. Each portion is
19 handling -- each tile in Agarwal is managing a portion of the shared cache
20 memory. Collectively, they are a single caching agent for the entirety of the
21 shared cache memory. That's exactly what the claim requires. It's the only
22 thing that's described in the '228 patent. And none of their kind of
23 exclusionary attempts there are supported by the patent or the claim
24 language itself.

25 Regarding their arguments about associated with, I think they said that
26 in order for it to be associated with, that distributed portion has to handle the

1 cache coherence for that entire core. First of all, that's not supported. They
2 didn't cite you to anything in the claim language at all. They didn't cite you
3 to anything in the specification. They cited to a background portion that
4 doesn't say anything about imposing that requirement on associated with.

5 But then I'll fall back on, that's exactly what Agarwal does. In
6 Agarwal for each core, there was a directory controller. Each tile has its
7 own directory controller. That directory controller manages L1 cache for
8 that core, it manages L2 cache for that core, and it manages the L3 cache
9 accesses either to its own portion of memory or to other portions of memory.
10 So Agarwal is doing exactly what they said, even under their restrictive
11 associated with, which isn't anywhere in the '228 patent.

12 I would say on their argument about figure 2 showing -- because
13 figure 2 shows a single TOR, therefore there must be a single TOR in the
14 LLC requirement here. Let me just tell you how that makes no sense.
15 Figure 2, which is what the caching agent writ large, also shows a single
16 LLC. If we go down to figure 6, figure 6 expressly shows multiple different
17 banks for the LLC. So under their view of the world, if their view were
18 correct, figure 6 would only show a single LLC.

19 And to be clear, the LLCs in figure 6 are exactly what is in Agarwal.
20 Figure 6, they're described as having an individual bank or slice. The
21 collective last level cache is made up of all the different banks or slices.
22 Each of them -- a bank is a particular portion of memory that is addressed by
23 a particular set of addresses. Collectively, those portions constitute the LLC
24 and the shared cache that's in the '228 patent. And that's exactly the same
25 architecture, exactly the same in Agarwal.

26 JUDGE PESLAK: Let me ask you a question about claim 2.

1 MR. ANDERSON: Yeah, sure.

2 JUDGE PESLAK: I thought at the end Mr. Fahmi said you were
3 proposing to modify Agarwal to meet claim 2. I may have misheard what he
4 said. Are you modifying that --

5 MR. ANDERSON: No, that's completely wrong. I mean, our view is
6 that when the claim says a table of requests and the specification teaches the
7 table of requests is within the caching agent and the claim requires a
8 distributed portions of the caching agent, you would have portions of the
9 table of requests distributed in the caching agent throughout. And that is
10 precisely what is shown in Agarwal.

11 So, the claim language here, and I'm looking at page 23 of the petition
12 which addresses claim 2, the claim language says a table of requests to store
13 pending requests. And then Agarwal has -- it keeps a small buffer
14 containing a list of outstanding transactions. That is the request. Agarwal
15 has exactly what is described in the '228 patent, a small buffer keeping the
16 pending or outstanding transactions there.

17 I will also -- I do also want to address they pointed to column 6, line
18 15 to 17, I think it was. What this passage is pointing out is --

19 JUDGE PESLAK: That's in the '228 patent?

20 MR. ANDERSON: Yes, I apologize. In the '228 patent, in columns
21 lines 15 to 17, or really 15 to 19. What this is pointing out is that part of the
22 invention is, I can reduce snooping. And just for context, what snooping is,
23 is when I decide that I'm going to access a particular portion of the shared
24 cache, I have to go out there and see who else might have it so that
25 everybody keeps -- everybody is keeping the same thing in their shared

1 cache. So the snooping is, I'm going out and I'm talking to everybody that
2 might have this.

3 And part of the invention of the '228 patent, they say is to reduce
4 snooping. Now they don't really tell you how, but they say that's part of it.
5 Precisely what Agarwal does, precisely. Because in Agarwal, I don't have to
6 go snoop. I don't have to go contact all the other places where the LLC is. I
7 know where it is because I have that address mapping that I talked about
8 earlier, which is the hash table. So when I, as tile 1, decide, hey, I need a
9 particular piece of memory from a particular address, I don't have to go
10 snoop tile 2, 3, 4, 5, 6, 7, 8. I know exactly where it is. It's in tile 3, because
11 I have that address mapping table.

12 And that address mapping is part of the collective action, and it's in
13 each of the cores. It's part of the collective action for all of the cores to
14 manage the overall shared cache, which again, if we go back to the claim
15 language, the single is referring to the management of the shared cache.
16 And these things collectively are a single entity that are managing the shared
17 cache. None of them are managing it on their own, and nor should they be.

18 On the issue of estoppel that came up, I think I heard Patent Owner's
19 counsel say it should have been raised in the petition. That would have been
20 quite difficult since the petition was filed before the final written decision
21 issued in that case. We did raise it to the extent that we could in footnote 1
22 on the very first page of the petition, where we apprised the Board that there
23 was a pending IPR with exactly the same prior art. I know there was some
24 citation to cases in there. Those cases didn't involve the exact same claim
25 language. And we showed that in our expert declaration, which is --

1 JUDGE PESLAK: Counsel, I'll save you some time here. I'm going
2 to ask the parties to submit some briefing on this after the hearing. So you
3 don't have to go over that anymore.

4 MR. ANDERSON: Okay, perfect. Although we would say that we
5 don't need collateral estoppel to win on Agarwal, obviously -- yes. In terms
6 of this issue of the architecture, the architecture of Agarwal is identical to
7 figure 6. Agarwal shows, as we showed in the petition, a set of cores. Each
8 of those cores has a caching agent, or in Agarwal's language, a directory
9 controller. And each of those cores has one bank or a portion of the LLC,
10 which is the shared cache in the claim language.

11 Similarly, Agarwal, just like the '228 patent, has an IO interface or an
12 IO module, which allows IO devices to use the same coherence mechanism
13 that is used to manage the shared cache in the -- as with accesses by the
14 other cores. And that was really actually the alleged novelty of this. And
15 there's no dispute that Agarwal has that sort of ability to allow IO devices to
16 access the caches and to ensure cache coherence for those IO device
17 accesses in the same manner as when the cores are accessing. Unless there's
18 any further questions.

19 JUDGE PESLAK: Judge Hoffman? Judge Anderson, any further
20 questions?

21 JUDGE ANDERSON: No, I got it.

22 JUDGE PESLAK: Okay, thank you.

23 MR. ANDERSON: Just want to emphasize again, the single means
24 the shared cache. That's what's in the claims.

25 JUDGE PESLAK: Okay. Mr. Fahmi, correct me if I'm wrong, I think
26 you're left with over 14 minutes.

1 MR. FAHMI: It's fine, Your Honor. I don't think I need that much.
2 Your Honors, regarding the Petitioner's discussion of what single means and
3 our so-called exclusionary constructions, they're not even happy with your
4 original claim construction, apparently, because they're not willing to
5 concede that single even means one. Your original claim construction is that
6 there's one cache controller for the processor. They're not even happy with
7 that. They want something else.

8 But we know that's not right. We know that the claim itself-- we just
9 have to read the claim. And it tells us that the caching agent is a single
10 caching agent for the multi-core processor. It's for the entire processor. And
11 it goes on to say that it includes distributed portions, each associated with a
12 corresponding one of the plurality of cores. Notice, associated with the
13 cores, not with memory regions, as in Agarwal.

14 Agarwal does not associate controllers, directory controllers, with
15 cores. It associates directory controllers with address regions in memory.
16 And that is why you see the tiles having to go back and forth with one
17 another.

18 And I disagree with Mr. Anderson about the reading in the '228 patent
19 concerning the passage in column 6. Let's read it together. Column 6,
20 beginning at line 15. By reducing the number of caching agents, right? That
21 means single caching agent. By reducing the number of caching agents,
22 much less resource division in the system, such as home agent tracker
23 entries, et cetera, occurs.

24 Now, Mr. Anderson wanted you to focus on what comes later. And
25 by reducing the amount of snoop overhead. Now, that's entirely relevant as
26 to Sinharoy, but we didn't get into Sinharoy. Let's just focus on the first part

1 of that statement in column 6. Reducing the number of caching agents, i.e.
2 going to a single, one caching agent, much less resource division in the
3 system, such as home agent tracker entries, et cetera, occurs.

4 Well, Mr. Anderson just told you that Agarwal is entirely based on
5 home agent tracking. He told you that these individual controllers have to
6 keep track of which tile is the home. Well, the '228 patent is an entirely
7 different architecture. It employed the single tracking agent precisely to
8 eliminate home agent tracking as in Agarwal.

9 JUDGE PESLAK: So, is tile 1 in Agarwal the home tile, is that what
10 you're saying?

11 MR. FAHMI: It will be for certain address regions. Tile 2 might be
12 for other address regions. The address regions get partitioned in Agarwal. It
13 offers up several methods, but one of them I recall is a hashing function. So
14 it takes the entire memory space, hashes those memory spaces into bins and
15 assigns the bins to the individual directory controller.

16 Your Honor, we can read claim 2 again for the table of requests.
17 Petitioner would like you to think that there can be lots of tables of requests.
18 Well, let's read the claim. The apparatus of claim 1, wherein the caching
19 agent includes, and the caching agent, by the way, their own expert, Dr.
20 Burnett is referring to the single caching agent. So we're talking about one.

21 The apparatus of claim 1, wherein the caching agent includes a table
22 of requests to store pending requests for the plurality of cores and the IIO
23 module. Not lots, not many, not multiple. A table of requests for the
24 plurality of cores, for all of them, single table of requests. Shared, yes,
25 single table. Petitioner --

1 JUDGE PESLAK: What about his argument that figure 2 shows that
2 the table of requests is associated with last-level cache two third. And
3 according to Mr. Anderson, that means there's a table of requests for each
4 last-level cache in figure 6.

5 MR. FAHMI: I like Mr. Anderson a lot. He's mistaken on this point.
6 We've already said that the last-level cache is a shared cache. It's depicted in
7 figure 6, according to its individual partitions, it's depicted in figure 2 as one
8 shared cache.

9 JUDGE PESLAK: Okay, I understand.

10 MR. FAHMI: You mentioned, Your Honor, you're going to ask for
11 briefing on the issue of estoppel, and if it's all right with you, I'll confine our
12 remarks then to whatever's in that briefing.

13 JUDGE PESLAK: Oh, yeah. I brought this up today, and I know you
14 need to go ahead and prepare, so let's not shoot in the dark, okay? And
15 waste time that way.

16 MR. FAHMI: Sounds good to us, Your Honor. What more can I tell
17 you? The patent itself makes the distinction from Agarwal. Single caching
18 agent reduces division, including home agent tracking resources,
19 diametrically opposed to Agarwal. I'm happy to take further questions.
20 Otherwise, we'll leave it in your capable hands.

21 JUDGE PESLAK: Judge Hoffman, any questions?

22 JUDGE HOFFMAN: Well, as you were talking about the home
23 agent, I was looking to the patent to understand that a little bit better. Can
24 you explain in the context of the '228 patent, the home agent. So in
25 particular, figure 6, reference number 770?

1 MR. FAHMI: The home agent in figure 6 is providing, I believe, the
2 path to and from the system memory, just as the home tile in Agarwal is
3 providing the path to and from the system memory in figures 6a and 6b. It's
4 a term of art that's used to show the home location of the requested data, so
5 it's the path to and from memory.

6 JUDGE HOFFMAN: Okay, thank you.

7 JUDGE PESLAK: Judge Anderson, any further questions?

8 JUDGE ANDERSON: No further questions.

9 JUDGE PESLAK: Okay. Thank you, Mr. Fahmi.

10 MR. FAHMI: Thank you.

11 JUDGE PESLAK: So before we go off the record here, as I said, the
12 Board would find it helpful to get some additional briefing on collateral
13 estoppel. And there's three issues that we've been considering. One is the
14 construction of single agent itself. Second, the factual finding in the prior
15 IPR that Sinharoy discloses that single caching agent. And then third,
16 whether the finding that Sinharoy disclosed all the limitations of the claims
17 at issue in the 1344 IPR collaterally stops Patent Owner from asserting that
18 the claims as a whole here aren't covered by Sinharoy.

19 So, we'd like the parties each to file an opening brief, not to exceed
20 seven pages, by July 6th, and then each side can file a response to the other
21 party's brief by July 13th, also not to exceed seven pages. So I want to ask
22 counsel, we have the 4th of July holiday in the middle of that. And is that a
23 problem for anyone's vacation to adhere to that schedule?

24 MR. ANDERSON: For counsel for Petitioner, that's not a problem.

25 JUDGE PESLAK: Mr. Fahmi?

26 MR. FAHMI: It's fine, Your Honor. Thank you.

1 JUDGE PESLAK: Okay. Well, thank you. It was a very informative
2 presentation for all of us today. I thank both sides for that. The hearing is
3 closed. We'll take the matter under advisement and issue our final written
4 decision in due course.

5 (Whereupon, the above-entitled matter went off the record at 2:37
6 p.m.)

PETITIONER:

Kevin Anderson
Patrick McPherson
DUANE MORRIS LLP
kpanderson@duanemorris.com
pdmcpherson@duanemorris.com

Cory Bell
Joseph Long
Alexander Boyer
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER, LLP
cory.bell@finnegan.com
jp.long@finnegan.com
alexander.boyer@finnegan.com

PATENT OWNER:

Tarek Fahmi
Jonathan Tsao
ASCENDA LAW GROUP, PC
tarek.fahmi@ascendalaw.com
jonathan.tsao@ascendalaw.