

Document Number: P0047R0  
Date: 2015-09-25  
Authors: Michael Wong  
Project: Programming Language C++, SG5 Transactional Memory  
Reply to: Michael Wong <michaelw@ca.ibm.com>

## **SG5: Transactional Memory (TM) Meeting Minutes 2015/06/01-2015/09/21**

### **Contents**

Minutes for 2015/06/01 SG5 Conference Call .....	2
Minutes for 2015/06/29 SG5 Conference Call .....	6
Minutes for 2015/07/13 SG5 Conference Call .....	8
Minutes for 2015/08/10 SG5 Conference Call .....	14
Minutes for 2015/08/24 SG5 Conference Call .....	19
Minutes for 2015/09/21 SG5 Conference Call .....	24

## Minutes for 2015/06/01 SG5 Conference Call

Minutes by Michael Wong.

On Sun, May 31, 2015 at 7:06 PM, Michael Wong <[fraggamuffin@gmail.com](mailto:fraggamuffin@gmail.com)> wrote:  
Sorry for the late agenda as I just returned from a series of business trips that started with the Lenexa meeting.

But there is a call tomorrow Monday- mostly to report and organize for the future.

Start Time: Monday, June 1, 2015, 12:00 PM US Pacific Time (07:00:00 PM in GMT)

End Time: 1:00 PM US Pacific Time (duration: one hour)

Conference ID:

6754398

Local Dial-in number: 416-933-9400

Toll-free Dial-in number: 1 855 453 6951/866-576-2504

Help desk: 1 877 549-3145

Customer care: 1 800 667-3678

Global Toll-free Dial-in number: (use the following prefix for your country code-without a 1 between the prefix and this number) 800-4444-3030

[http://www.conferencing.bell.ca/en/ressource\\_centre/pdf/global\\_toll\\_free.pdf](http://www.conferencing.bell.ca/en/ressource_centre/pdf/global_toll_free.pdf)

All IBM locations: use internal tie line to call

With large numbers of participants, audio interference can be a problem. Please try to keep your phone muted whenever possible. If your phone does not have a mute button, the bridge will mute or un-mute your line if you dial \*6.

The current secretary rota list is (the person who took notes at the last meeting is moved to the end)

Torvald, Tatiana, Mike Spear, Justin, Jens Maurer, Michael Wong, Michael Scott, Victor, Maged, Hans

Agenda:

1. Opening and introductions

1.1 Roll call of participants

Tatiana, Mike Spear, Michael Wong, Michael Scott, Victor, Maged, Hans

## 1.2 Adopt agenda

OK

1.3 Approve minutes from previous meeting, and approve publishing previously approved minutes to ISOCPP.org  
Already done.

## 1.4 Review action items from previous meeting (5 min)

1.4.1. All: Consider attending Kona Meeting in Oct 19-24, 2015. There is no mailing document yet, but here is an email from Thomas Plum of PlumHall who is hosting the meeting along with ISOCPP.org.

[The hotel's web site is here:](#)

<http://www.royalkona.com>

*(This was the location of the February 2012 meeting, as well as others before that.) The hotel is located in downtown Kailua-Kona, which has remained the most-requested location for Big Island meetings. In addition to the hotel restaurant, there are many other eating places within walking distance. Because of the downtown location, a rental car is unnecessary; however, it can be useful for island excursions.*

*The Keahole Airport (Kailua-Kona, airport code KOA) is about 8 miles north from downtown; a shared taxi is the best airport transport. Don't allow your travel agent to book you inadvertently to Hilo Airport (ITO), unless you have intentionally planned the scenic three-hour drive into your itinerary. If your travel agent books you to Honolulu, you will still need another flight to get to Kona (KOA).*

*Reservations for the hotel room can be made online at this web page:*

<http://www.royalkona.com/Groups>

*Then enter Groupcode plumhall*

*Then be sure that your arrival and departure dates are correct!*

*The October 2015 WG21 (C++) meeting will be sponsored by Plum Hall and the Standard C++ Foundation. That meeting will take place at the same venue (Royal Kona Resort), from Monday Oct 19 to Saturday Oct 24. The C meeting is the week before.*

This meeting is where we will consider future features, and resolving defects to the current published PDTS. We intend to meet formally as an SG group.

Michael: Ideal time to attend is Tuesday, Wednesday and Thursday.

## 2. Main issues (50 min)

### 2.1 Review PDTS publication from Lenexa

N4514 should be linkable in another week.

These form C++17 candidates

TM TS2 likely will be cumulative (not union) on top TS1

### 2.2 Review future plans.

-C specification?

-additional constructs? see:

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4438.pdf>

Michael Scott+Spear: No I/O for synchronized block, and nothing for condition variables (retry), C++ atomics inside atomics constructs

Hans: may not be solvable but need to talk about

Tatiana: failure atomicity and tx in context of system memory, non-volatile memory.

Michael Scott: are there proposals for scoped parallelism,

This is the task region proposal, might be a hook for considering concurrency inside transactions,

Maged: Non-blocking domain transactions; not compatible with synchronized blocks, a separate option, when someone need that level of progress guarantees

Victor: explicit abort; escape actions; synchronization inside

Michael Spear: transaction pure is not in our spec; custom assembly code that you know is side-effect free, but compiler can not prove it is side-effect free; network programming, host-endian to network-endian code

Victor: many are special cases of escape actions

### 2.3 Change time of this call?

No. Not now.

Academics may need to revisit callin time at end of August.

### 2.4 Change chair?

Continue current chair.

3. Any other business

4. Review

4.1 Review and approve resolutions and issues [e.g., changes to SG's working draft]  
N4513 is the official working draft (these links may not be active yet until ISO posts these documents)

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4513.pdf>

N4514 is the published PDTS:

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4514.pdf>

N4515 is the Editor's report:

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4515.html>

Github is where the latest repository is (this is still the pre-Lenexa; I will upload changes soon):

<https://github.com/cplusplus/transactional-memory-ts>

4.2 Review action items (5 min)

5. Closing process

5.1 Establish next agenda

Wyatt paper author review on June 29. Michael Scott can't make it.

5.2 Future meeting

Next call : June 15.

Changed to June 29 because many people are away.

Past and future agendas

June 1: First post Lenexa review

June 15: Transact/PLDI/FCRC

June 29: Wyatt paper review.

July 13:

July 27:

Aug 10:

Aug 24:

Sept 7: Labor Day

Sept 21: CPPCon. Michael away.

Oct 5: OpenMP conf; Michael away

Oct 19: C++ Std meeting Kona.

## Minutes for 2015/06/29 SG5 Conference Call

Meeting minutes by Torvald

Brett Hall, Jens Maurer, Michael Wong, Victor Luchangco, Tatiana Shpeisman, Michael Scott, Hans Boehm, Torvald Riegel, [please fill in if I missed somebody].

> 2. Main issues (50 min)

>

> 2.1 Brett Hall of Wyatt to present TM in industry usage

>

> <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4438.pdf>

[Please correct the minutes if I got something wrong or missed something. Thanks.]

Brett presenting N4438.

- \* Current implementation: Uses global reader/writer lock; library interface; not-always-consistent, in some of their use cases explicit validation was required; has post-commit hooks; no equivalent of `transaction_safe`.

- \* Have a retry feature and use it, but less need for `or_else`.

- \* Use a simple contention manager that allows one transaction to reserve commit rights. Concern was starvation.

- \* Weakly atomic.

- \* User experience: Takes about a month for a programmer to get used to this TM system. A few rare cases may require more experienced programmers. Significantly easier to use than locks; problem is locks at large (code base) scale.

- \* Most customers probably use Intel 2-8 core machines with hyperthreading. Have done experiments with up to 24 cores.

- \* Have not tried lock elision yet.

Michael/Victor: Great to get this real-world perspective and user feedback for TM.

Brett: Really glad they decided to use TM and that it worked this well.

[Brett explains "after" method feature; is an after-commit hook.]

[Brett explains uses for retry.]

[Brett and Victor discuss library interface vs. compiler support.]

Brett: Don't use thread-locals in transactions.

Brett: Can't use move semantics because they can't roll this back in their implementation.

Victor: Biggest reasons to want compiler support?

Brett: Some cases where programmers do non-transaction-safe things in transactions; compiler checking for this would be nice.

Tatiana: How large are transactions (memory accesses)?

Brett: Many read 1-2 txnal vars and write to another one. But varies a lot between transactions and use cases.

Tatiana: And access to many nontransactional memory locations in transactions?

Brett: Varies. Some transactions do big calculations in transactions.

Victor: `tm_pure` could be a means to do the calculations nontxnally.

Brett: Would be nice to have for this use case.

[Time was up. If someone has more questions for Brett, please send email.]

## Minutes for 2015/07/13 SG5 Conference Call

Meeting minutes by Mike Spear

On Fri, Jul 10, 2015 at 1:03 PM, Michael Wong <[fraggamuffin@gmail.com](mailto:fraggamuffin@gmail.com)> wrote:  
Start Time: Monday, July 13, 2015, 12:00 PM US Pacific Time (07:00:00 PM in GMT)  
End Time: 1:00 PM US Pacific Time (duration: one hour)

Conference ID:

6754398

Local Dial-in number: 416-933-9400

Toll-free Dial-in number: 1 855 453 6951/866-576-2504

Help desk: 1 877 549-3145

Customer care: 1 800 667-3678

Global Toll-free Dial-in number: (use the following prefix for your country code-without a 1 between the prefix and this number) 800-4444-3030

[http://www.conferencing.bell.ca/en/ressource\\_centre/pdf/global\\_toll\\_free.pdf](http://www.conferencing.bell.ca/en/ressource_centre/pdf/global_toll_free.pdf)

All IBM locations: use internal tie line to call

With large numbers of participants, audio interference can be a problem. Please try to keep your phone muted whenever possible. If your phone does not have a mute button, the bridge will mute or un-mute your line if you dial \*6.

The current secretary rota list is (the person who took notes at the last meeting is moved to the end)

Tatiana, Mike Spear, Justin, Jens Maurer,,Michael Scott, Victor, Maged, Hans, Michael Wong, Torvald,

Tatiana, Justin, Jens Maurer, Michael Scott, Victor, Maged, Hans, Michael Wong, Torvald, Mike Spear

Agenda:

1. Opening and introductions

1.1 Roll call of participants

Michael Wong, Michael Scott, Maged Michael, Michael Spear, Jens Maurer, Victor Luchangco



## 1.2 Adopt agenda

Adopted

1.3 Approve minutes from previous meeting, and approve publishing previously approved minutes to ISO CPP.org

Approved

1.4 Review action items from previous meeting (5 min)

1.4.1. All: Consider attending Kona Meeting in Oct 19-24, 2015.

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4544.htm>

This meeting is a Future features for TM2 TS Kickoff, and resolve defects to the current published PDTs. We intend to meet formally as an SG group.

Michael: Ideal time to attend is ?

This is an important meeting to attend. If we are going to have a face-to-face SG5 at Kona, it will be tricky since many members are going to be in other meetings during the rest of the week in Hawaii.

We could have a meeting fairly early in the week before being pulled into other committees.

Michael Scott: we've been doing well on the conference calls, so perhaps we don't need a face-to-face.

Victor: agree

Jens: past experience is that the face-to-face wasn't more successful than a call, so face-to-face is probably not worth it.

Michael W: Portland 2012 is probably the only time we had a lot of face-to-face. We traditionally need to work through papers that document the ideas, gather feedback, summarize the reflector, and discuss advantages/disadvantages of our chosen direction.

Victor: when/where is meeting after Kona?

Michael W: Jacksonville Florida February 2016.

Victor: Hawaii is nice, but it's hard to justify, especially since we won't have many papers to discuss by then. Unless we have a definite plan for it, let's not do a face-to-face in Kona.

Jens: TS expires after a few years, but they can be reconfirmed once. There is a timetable, though, for moving beyond a TS and into the standard. Face-to-face time is very expensive time, too, unless we actually have papers to discuss. Would prefer we don't spend regular daytime on

that sort of discussion. Evening time (to progress a paper of some sort) would be OK, but unless we have specific papers, there are more important things to do during the 9-5 time at Kona. And so far there has been no deluge of papers.

Outcome: reconsider if we get a lot of papers in August.

## 2. Main issues (50 min)

2.1 Discuss Hawaii logistics, when to have the F2F SG5 meeting.

2.2 Logistics of new feature proposal moving forward. Do we continue with same meeting schedule?

Only 3-4 calls until Michael W starts travelling heavily. Mailing deadline is between Sep 7 and Sep 21. If we have papers by then, we should put them up as an interim checkpoint. We may have defect fixes to put in by then, too.

Jens: do we have implementations of anything?

Michael W: Torvald not on call to discuss GCC. Michael W is trying to work on a clang implementation, but it's not started yet.

Jens: it would be great to have more progress here, since we're all interested and there is demand. Maybe a vendor will pick it up, too?

Mike S: how much of what is in the TS but not GCC is just spelling?

Jens: but spelling matters! Companies don't want to buy into the cost of refactoring the code in the future.

Michael Scott: Mike is saying the cost is small... if so, it should be easy to make it happen.

Michael W: will follow up with Torvald and Tatiana.

Michael Scott: is there any update on IBM support?

Michael W: Nothing to report yet. It's not that he's hiding anything, just that nothing is implemented yet. The BG implementation was different syntax. And IBM has now switched to a clang-based front-end compiler, so there won't be xlc work yet.

## 2.3 Review future Features for TM2 TS

commit/abort actions

Non-atomic commit actions have consensus. There was only one case from Wyatt Tech about atomic commit actions.

Mike S: summarized reflector discussion. non-atomic commit actions make the most sense, then non-atomic abort, but only if we have escape actions. Two open issues for commit actions: can they use transactions, and can they use oncommit actions.

Victor: question is whether there is any guarantee about the order of these actions (or could they be in parallel). If we guarantee an order (in the order they are registered), then a transaction from a handler should run its handlers when it commits. Most flexible for implementer is to not specify the order, but people may expect certain behaviors.

Mike S: asked for clarification.

Victor: If we guarantee an order (which he doesn't advocate), then a nested transaction's handler should finish before the next top-level handler.

Mike S: agrees with Victor if we do order, and favors order.

Jens: also favors order, opposes parallel execution of commit handlers.

Mike: if a commit handler forks threads, does that behave oddly with the memory model?

Victor and Jens: No.

Victor: concerned when there is a nested transaction, and the child has a handler, when does it run? When the parent commits?

Mike: when the parent commits.

Victor: ok.

Mike: this does encourage sequential thinking, but we've already drank that kool-aid.

Victor: ok, now what about the order of the handlers I register in a transaction?

Mike: FIFO makes the most sense.

Jens: FIFO make sense for registering log entries at commit time.

Michael Scott: FIFO / registration order makes the most sense for commit. OnAbort is probably the other way (LIFO), if we ever discuss it.

Jens: but we need more examples.

Maged: we were symmetric in commit/abort for malloc/free.

Michael S: should malloc be a special case, or something the programmer has to deal with. If we don't build it in, we need onabort.

Mike S: prefers an implementation where malloc/free are not special case, and we have onAbort.

Victor: sympathetic toward this view, but doesn't want us to go there yet. There are a lot of dragons in that space.

Mike S: for example, can an escape action use transactions.

Victor: yes, among other things. And Victor is a big proponent of escape actions, but doesn't want us to confront them yet.

Mike: Should abort wait until after Kona?

Victor: no, once we finish commit, move to escape actions (and abort comes with it).

Mike and Victor will write a paper on commit actions.

Jens: be sure to document why we think it is a great addition to C++ TM, what is the syntax, and how does it work.

Jens, Michael W: syntax can be rough, but should have something right away.

Michael W: out of time

retry

or\_else

Have I missed any?

3. Any other business

4. Review

4.1 Review and approve resolutions and issues [e.g., changes to SG's working draft]

N4513 is the official working draft (these links may not be active yet until ISO posts these documents)

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4513.pdf>

N4514 is the published PDTS:

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4514.pdf>

N4515 is the Editor's report:

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4514.html>

Github is where the latest repository is (this is still the pre-Lenexa; I will upload changes soon):

<https://github.com/cplusplus/transactional-memory-ts>

#### 4.2 Review action items (5 min)

Michael Wong: will follow up with Torvald and Tatiana about TS support in GCC and ICC.

Mike and Victor will write a paper on commit actions.

#### 5. Closing process

##### 5.1 Establish next agenda

##### 5.2 Future meeting

Next call : July 27

##### Past and future agendas

June 1: First post Lenexa review

June 15:Cancelled due to Transact

June 29: Brett Hall of Wyatt to call in

July 13:Logistics of future meetings; new proposal writeups

July 27:

Aug 10:

Aug 24:

Sept 7: Labor Day, no call

Sept 21: CPPCon. Michael away.

Oct 5: OpenMPCon; Michael away

Oct 19: C++ Std meeting Kona.

## Minutes for 2015/08/10 SG5 Conference Call

Minutes by Tatiana Shpeisman.

Start Time: Monday, Aug 10, 2015, 12:00 PM US Pacific Time (07:00:00 PM in GMT)

End Time: 1:00 PM US Pacific Time (duration: one hour)

Conference ID:

6754398

Local Dial-in number: 416-933-9400

Toll-free Dial-in number: 1 855 453 6951/866-576-2504

Help desk: 1 877 549-3145

Customer care: 1 800 667-3678

Global Toll-free Dial-in number: (use the following prefix for your country code-without a 1 between the prefix and this number) 800-4444-3030

[http://www.conferencing.bell.ca/en/ressource\\_centre/pdf/global\\_toll\\_free.pdf](http://www.conferencing.bell.ca/en/ressource_centre/pdf/global_toll_free.pdf)

All IBM locations: use internal tie line to call

With large numbers of participants, audio interference can be a problem. Please try to keep your phone muted whenever possible. If your phone does not have a mute button, the bridge will mute or un-mute your line if you dial \*6.

The current secretary rota list is (the person who took notes at the last meeting is moved to the end)

Jens Maurer, Michael Scott, Victor, Maged, Hans, Michael Wong, Torvald, Mike Spear,

Agenda:

1. Opening and introductions

1.1 Roll call of participants

Tatiana, Hans, Michael Wong, Michael Scott, Maged, Michael Spear, Joe McFallen, Brett

1.2 Adopt agenda

OK

1.3 Approve minutes from previous meeting, and approve publishing previously approved minutes to ISOCPP.org  
OK

1.4 Review action items from previous meeting (5 min)

1.4.1. Contact Torvald on GCC implementation:

Done. Torvald responded that they will be starting the implementation within 6 months

1.4.2. Mike and Victor will write a paper on commit actions.

The paper has been written.

1.4.3. All: Consider attending Kona Meeting in Oct 19-24, 2015.

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4544.htm>

We have decided in the TM July 13 call that we will not require folks to attend this meeting since we already have telecon calls to triage and review papers. We plan to ask people to write new proposal papers and review them through the calls. These proposals can also be published through the mailing deadlines.

If people would still like to attend the meeting, that is fine and if we have enough people, we can also have a brief impromptu informal meeting to discuss any hot issues. But since most of the meeting has time allocated to so many SGs now, it is not a guarantee that a meeting time will be found. It was felt by our SG that we can efficiently conduct our business on telecons.

This is the same guidance as at the previous meeting.

2. Main issues (50 min)

2.1 Review future Features for TM2 TS

commit/abort actions

Discussion on the paper “Extending the Transactional Memory Technical Specification to Support Post-Commit Actions”.

Michael Spear: We went through exhaustive list of options and based on those conversation the document proposes relatively straightforward solution. In the current TM TS, irrevocable actions are supported in synchronized blocks. But, synchronized blocks cannot abort. Using local variables to express deferred actions does not work either. We propose a new construct - `transaction_defer(lambda_function)` – that defers the action till transaction commit.

Open questions:

- Is this a mechanism for atomic blocks only or also synchronized blocks?
- Is passing lambda as an argument the right approach or should we have more traditional API?

Michael Scott: It might be possible to solve this with local variables if we had a predicate that says if you are in or out of outermost transactions. On abort, though, we cannot solve it using local variables.

Victor: First suggestion is good. What happens on abort is an orthogonal question. I think there is a broad agreement that we want such a predicate.

Michael Scott: Having a predicate that indicates if execution is in or out of outermost transaction does not really solve the problem of such a solution being clunky.

Everybody agrees that we need to add this to the motivation discussion in the introduction section.

Michael Scott: Is the purpose of the paper to justify the mechanism that we could implement today or also provide a roadmap for where we might want to go?

Michael Wong: Describing the mechanism is best.

Michael Scott: I misunderstood – transaction deferred lambda executes only in case of commit. Please ignore everything I said about aborts. With the current specification abort is not observable by the application.

Victor: Michael's point is well taken. We should make clear that deferred action happens only on commit.

[Discussion on whether we need deferred semantics for synchronized block.]



Hans: In synchronized block, we should support it too. Actions should be deferred after the synchronized region.

Tatiana: Actions should be deferred in synchronized block.

Victor: What happens when you use `transaction_defer()` outside of either atomic or synchronized block?

Brett: Is it OK to use synchronized block inside atomic block?

[The answer is yes. A question on where deferred code executes. ]

Victor: You never know how far it is going to get deferred. It's always deferred to the outermost transaction.

Michael Scott: I'd like to take a strong stance against runtime error behavior. If we allow it, the attribute becomes `transaction only` rather than `transaction safe`.

[Hans and Victor agree that whether `transaction_defer()` should be deferred in a synchronized block depends on how much similarity we see between synchronized and atomic blocks.]

Michael Spear: If we see synchronized block as a transition to atomic block for legacy code we should defer `transaction_defer()` execution in synchronized blocks.

Victor: An alternative argument is that synchronized block should behave as non-transactional code.

Tatiana: Motivating example of `transaction_defer()` usage could help us decide on the right behavior for synchronized blocks.

Victor agrees that such example would be nice to have.

Hans: I am still concerned with what happens if you execute it outside of transaction.

[Hans proposes that deferred action should execute only as a result of an explicit call to a special handler to execute such calls if `transaction_defer()` was called not in a transaction. But, never at the call of deferral.]

Michael Scott: This could lead to empty transaction afterwards.

Victor: An empty transaction after the code would not affect it.

Hans: It would be an explicit call to run it for non-transactional case.

Victor: It makes me uncomfortable. ...

Brett: In our system, we are going through the transaction. If the transaction has committed it's OK for this stuff to run

Michael Spear: Can somebody do a sanity check on lambdas?

New Action items:

Michael Wong: Help sanitize lambda description posted by Michael Scott.

Victor: Elaborate on an example of `transaction_defer()`

Michael Wong: Somebody should start writing a paper on future items for the next version of TM TS.

Michael Scott: I do not think we need to start writing such papers now.

## Minutes for 2015/08/24 SG5 Conference Call

### Minutes by Michael Scott

- > With large numbers of participants, audio interference can be a
- > problem. Please try to keep your phone muted whenever possible. If
- > your phone does not have a mute button, the bridge will mute or
- > un-mute your line if you dial \*6.

Michael Scott taking minutes. New Secretary Rota is  
Jens Maurer, Victor, Maged, Hans, Michael Wong, Torvald, Mike Spear, Tatiana, Michael Scott

- > Agenda:
- >
- > 1. Opening and introductions
- >
- > 1.1 Roll call of participants

Roll call:

Michael Wong, Maged Michael, Michael Scott, Brett Hall, Mike Spear,  
Hans Boehm, Victor Luchangco, Tatiana Shpeisman.

- > 1.2 Adopt agenda

done

- > 1.3 Approve minutes from previous meeting, and approve publishing
- > previously approved minutes to ISOCPP.org

done

- > 1.4 Review action items from previous meeting (5 min)
- > 1.4.1. All: Consider attending Kona Meeting in Oct 19-24, 2015.
- > <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4544.htm>
- >
- > We have decided in the TM July 13 call that we will not require folks
- > to attend this meeting since we already have telecon calls to triage
- > and review papers. We plan to ask people to write new proposal papers
- > and review them through the calls. These proposals can also be
- > published through the mailing deadlines.
- >
- > If people would still like to attend the meeting, that is fine and if
- > we have enough people, we can also have a brief impromptu informal
- > meeting to discuss any hot issues. But since most of the meeting has

- > time allocated to so many SGs now, it is not a guarantee that a
- > meeting time will be found. It was felt by our SG that we can
- > efficiently conduct our business on telecons.
- >
- > 2. Main issues (50 min)
- >
- > 2.1 Review future Features for TM2 TS
- >
- > commit/abort actions from Mike Spear paper

Michael W. looked at syntax for lambda; now incorporated into Mike Spear's paper draft.

Still need to clarify that none of this applies to aborts.

Victor: also need to clarify that nothing happens until the end of (the dynamic extent of) an `_outermost_ atomic/synchronized` block.

Note also that capture issues apply only to capture by reference. This is a standard issue with C++ lambdas; we just introduce a new context in which to shoot yourself.

Victor: Should it be an error to capture a variable by reference in an on-commit lambda?

Michael Scott: not all capture by reference is unsafe. In the general case, "is this capture safe?" is undecidable.

Hans / Michael Scott: Do we really want to require a lambda expression, or perhaps an `std::function`? Or maybe make it a templated feature with a lambda version and an `std::function` version?

Mike Spear: Do we need to be compatible with C?

Michael Wong: Not sure. If we ever want TM to be added to C, we'll need a non-lambda equivalent.

Victor: Prefer not to invest too much thought in this now; can expect feedback from the committee.

Mike Spear: Maybe provide a simple C solution now, on which to expect feedback.

Michael Wong: This sounds a lot like the current gcc facility.

Hans: C compatibility is tricky -- want to facilitate without making the C++ case uglier. In particular, does the function you pass to `transaction_defer` ("on commit") take zero, one, or a variable number of parameters? There are places in the language now (e.g., `std::thread`) that use a function pointer and "argument pack" rather than a lambda.

Mike Spear: Are we worried about pre-C++11 code?

Hans/Victor: Don't think so.

Michael Scott: So what about Hans's suggestion of making `transaction_defer` a templated feature, with `lambda`, `std::function`, and `function/arg` pack versions?

Hans: has "gotchas" wrt by-reference parameters, but people are getting used to these -- use `std::reference_wrapper`.

Consensus seems to be with the templated approach; use language from `std::thread`.

Motivating examples?

The "after actions" of Brett's paper were used mainly when a txn needed to do I/O. For SG5 TM, it's a way to do this without resorting to synchronized blocks that execute serially.

> `retry`  
> `or_else`  
> Have I missed any?

Escape actions.

Michael Scott: Note, also, that `retry` is semantically simpler than `or_else`.

Brett: Provided both, and had use for `retry`, but never used `or_else`.

Victor: Escape actions seem more important.

Maybe also the related notion of "transaction waiver" -- things that are "fine" without instrumentation.

Mike Spear: Is `retry` really the right way to do condition synchronization? Should we consider any of the more efficient alternatives?

- `transaction-safe` condition variables
- Atomos 'await condition-variable-set'
- conditional critical section equivalent (as in, say, X10): check only at beginning.

Brett: Would be surprised if there were uses of `retry` that weren't inside a fairly trivial 'if'.

Michael Scott: Lean pretty strongly toward the 'await condition\_variable' approach. (Lots of precedent -- all the way back to Kessels's work on monitors [CACM 1977].)

Hans: Want to avoid anything that becomes viral.

Michael Wong: Anybody want to do a paper on this? Or maybe do escape actions first?

Victor: Do escape actions first.

Michael Wong: Or whatever somebody who cares about the issue proposes first!

- > 3. Any other business
- >
- > 4. Review
- >
- > 4.1 Review and approve resolutions and issues [e.g., changes to SG's working draft]. N4513 is the official working draft (these links may not be active yet until ISO posts these documents)
- > <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4513.pdf>
- >
- > N4514 is the published PDTS:
- > <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4514.pdf>
- >
- > N4515 is the Editor's report:
- > <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4514.html>
- >
- > Github is where the latest repository is (I have updated for latest PDTS published draft from post-Lenexa):
- > <https://github.com/cplusplus/transactional-memory-ts>
- >
- > 4.2 Review action items (5 min)

Michael Wong: Only pending action item is re-write of Mike Spear's paper on `transaction_defer`. Victor to work on a motivating example (see old minutes).

- > 5. Closing process
- >
- > 5.1 Establish next agenda
- > 5.2 Future meeting
- > Next call : Sept 7?

Labor Day in US and Canada.

Would be tight for planning for Kona meeting.  
No TM presentation yet planned for that...  
How about Sept. 14th?

Michael Scott: Can't join any fall meetings (after Labor Day) until 3:30.  
Mike Spear: Can't go later.

Consensus: Meet on the 14th, at 3:00.  
Mike Spear: Unless we reach agreement on-line.

- > Past and future agendas
- > June 1: First post Lenexa review
- > June 15: Canceled due to Transact

- > June 29: Brett Hall of Wyatt to call in
- > July 13: Logistics of future meetings; new proposal writeups
- > July 27: Canceled
- > Aug 10: More future TM2 feature triage
- > Aug 24:
- > Sept 7: Labor Day, no call?
- > Sept 21: CPPCon. Michael W. away.
- > Oct 5: OpenMPCon; Michael W. away

# Minutes for 2015/09/21 SG5 Conference Call

## Minutes by Jens Maurer

2015-09-21 19:19 hrs

### Attendees:

Maged Michael, Michael Wong, Michael Spear, Hans Boehm, Jens Maurer

1.2 Adopting the agenda  
no objection

1.3 Approve minutes from previous meeting, and approve publishing previously approved minutes to ISOCPP.org  
no objection

1.4 Review action items from previous meeting  
none at this time

### 2.1 Review TM for C draft

Michael Wong: Thanks for the comments on the C draft. Should we put out something with synchronized and atomic blocks without cancellation, or just with synchronized?

Michael Spear: I'm an advocate of atomic blocks with cancel, but we can't justify the burden on the C committee without exceptions.

Hans Boehm: The main argument for keeping it is compile-time support for speculative execution.

Michael Spear: Agreed.

Michael Spear: A transaction-safe function needs two versions, without name mangling in C?

Jens Maurer: The C compiler would need to do something like name mangling to differentiate tx-enabled from tx-disabled versions.

Jens Maurer: I'm uncomfortable spending a lot of SG5 time on a C feature, given that SG5 is a C++ study group.

Hans Boehm: WG14 will decide for or against atomic blocks anyway, regardless of the discussion in this group.

Michael Spear: We're between a rock and a hard place here: Without atomic blocks, people will be disappointed by the outcome; with atomic blocks, people will be disgusted by the heavy machinery.

Consensus: Michael Wong's paper is a good starting point, but WG14 will probably come back to us to tap into the TM expertise.

### 2.2 on\_commit paper

Jens: We should plan to publish the paper on Friday and form an opinion inside SG5 before Kona, so that people present in Kona can represent a consolidated SG5 opinion.

Michael Wong: Sometime down the road, we'll get an ISO project number to create a TM V2 specification. This will probably contain on\_commit. The on\_commit paper might come before EWG in Kona.

Action item for Michael Spear: finalize on\_commit paper by Friday.



### 2.3 retry discussion

Hans Boehm: We need a mechanism to interact with other threads by waiting for them. But atomic transactions expressly do not interact with other threads. Retrying a transaction that is nested inside another transaction would retry the outer transaction, which is probably unexpected. Since you can already implement retry nowadays (cancelling by exception), we already have the same issue today.

Michael Spear: At the point of retry, you wait on the read-set to change by another thread, yielding the processor meanwhile.

The bigger issue is "any nontrivial synchronization mechanism is going to end up in deadlocks" (Mark Moir). A transaction inside a synchronized block that became irrevocable cannot meaningfully retry.

Hans Boehm: We can prevent a retry inside a nested transaction by making it non-tx-safe.

Michael Spear: Retry inside a synchronized block is equivalent to waiting on a condition variable.

Michael Spear leaves at 20:00 UTC.

Hans Boehm: Retry is very useful for transactionalizing real code. I hope we can hide this inside a library. Other approaches are being investigated, but don't seem to be ready yet. It would be good to have Victor on the call for future discussions about this.

### 4.2 Review action items

Michael Spear to finish on\_commit paper.

Everybody to read and comment on the paper prior to Kona; absent substantiated opposition, the on\_commit paper will become an SG5 proposal for Kona.

Next meeting: Nov 2, 2015.

Adjourned at 20:07 UTC