

CLAIMS

What is claimed is:

1. An apparatus for optimizing code comprising:
5 a memory to hold program code; and
a processor including,
hardware resources adapted to support transactional execution and to
provide a representation of an availability of the hardware
resources; and
10 execution logic adapted to execute the program code to cause the processor
to dynamically resize a transactional region including an optimized
portion of the program code based on the representation of the
availability of the hardware resources.
2. The apparatus of claim 1, wherein execution logic adapted to execute the program
15 code to cause the processor to dynamically resize a transactional region including
an optimized portion of the application code based on the representation of the
availability of the hardware resources comprises: the execution logic being adapted
to execute a conditional commit instruction within the program code that is before
an end of the transactional region, the conditional commit instruction to commit
20 the transaction before the end of the transactional region in response to the
representation of the availability of the hardware resources indicating insufficient
resources to complete execution of the transactional region.
3. The apparatus of claim 2, wherein the memory is further adapted to hold dynamic
25 optimization code, and wherein the execution logic is further adapted to execute
the dynamic optimization code to optimize a portion of the code to obtain the
optimized portion of the program code including inserting, during runtime, the
conditional commit instruction before the end of the transactional region.
4. The apparatus of claim 1, wherein execution logic adapted to execute the program
30 code to cause the processor to dynamically resize a transactional region including
an optimized portion of the application code based on the representation of the
availability of the hardware resources comprises: the execution logic being adapted
to execute a transactional write from the transactional region, wherein the
transactional region is to be rolled back to a recent checkpoint and committed in
response to the transactional write overflowing a hardware store buffer.

5. The apparatus of claim 1, wherein the memory is selected from a group consisting of an on-processor cache memory, a system memory directly coupled to the processor, and a system memory indirectly coupled to the processor.
6. An apparatus comprising:
 - 5 decode logic adapted to decode a region check instruction associated with a transaction; and
 - hardware adapted to support execution of the transaction, wherein the hardware is further adapted to provide a representation of a hardware usage metric to a usage metric storage element in response to the decode logic decoding the region check instruction, wherein the usage metric storage element is adapted to be utilized in determining if a conditional commit of the transaction is to be performed before an end of the transaction.
7. The apparatus of claim 6, wherein the hardware is selected from a group consisting of a speculative cache memory, a store buffer, a speculative register file, and a speculative checkpoint register file, and wherein.
8. The apparatus of claim 6, wherein the region check instruction is to include a representation of an expected usage metric; and wherein the usage metric storage element is adapted to be utilized in determining if a conditional commit of the transaction is to be performed before an end of the transaction comprises the hardware being further adapted to compare the representation of the expected usage metric with the representation of the hardware usage metric to be provided to the usage metric storage element to obtain a usage comparison result; and determining if a conditional commit of the transaction is to be performed before the end of the transaction based on the usage comparison result.
9. The apparatus of claim 8, wherein the region check instruction is further to include a branch target address; the hardware is adapted to perform the determining if the conditional commit of the transaction is to be performed before the end of the transaction based on the usage comparison result utilizing a predefined algorithm, and wherein execution is adapted to jump to the branch target address in response to determining the conditional commit of the transaction is to be performed before the end of the transaction based on the usage comparison result utilizing the predefined algorithm.
10. The apparatus of claim 8, wherein the hardware is adapted to perform the determining if the conditional commit of the transaction is to be performed before

the end of the transaction based on the usage comparison result utilizing a predefined algorithm, and to expose a representation of the determining if the conditional commit of the transaction is to be performed before the end of the transaction based on the usage comparison result utilizing a predefined algorithm to conditional code associated with the transaction, the conditional code, when executed, is to:

jump to a conditional commit branch address in response to the hardware exposing a representation of determining that the conditional commit of the transaction is to be performed before the end of the transaction,
commit the transaction; and
start a new transaction.

11. The apparatus of claim 6, wherein the usage metric storage element includes a register that is adapted to be read by conditional code associated with the transaction, the conditional code, when executed, to evaluate the hardware usage metric in regard to a software expected usage metric of hardware, and to determine if the conditional commit of the transaction is to be performed before the end of the transaction based on the evaluation of the hardware usage metric in regard to the software expected usage metric.

12. The apparatus of claim 11, wherein the conditional code, when executed, is further to jump to a commit path, commit the transaction before the end of the transaction, and start a new transaction in response to determining the conditional commit of the transaction is to be performed before the end of the transaction based on the evaluation of the hardware usage metric in regard to the software expected usage metric.

13. An apparatus comprising:

decode logic adapted to decode a conditional commit instruction to be associated with a conditional commit point in a transaction, the conditional commit instruction to reference a destination address; and

hardware adapted to determine, in response to the decode logic decoding the conditional commit instruction, if a hardware resource is to include enough available space to support execution of a region of the transaction and to jump execution to the destination address in response to determining the

hardware does not include enough available space to support execution of the region of the transaction.

- 5 14. The apparatus of claim 13, wherein the hardware resource includes a memory device, and wherein the hardware is further adapted to determine if the memory device includes enough available space to support execution of the region of the transaction based on an expected number of entries to be utilized before a second conditional commit point in the transaction and a number of available entries in the memory device, wherein the hardware is adapted to determine the memory device does not include enough available space to support execution of the region of the transaction in response to the number of available entries being less than the number of expected entries.
- 10 15. The apparatus of claim 13, wherein the conditional commit instruction is further to reference the expected number of entries to be utilized before the second conditional commit point in the transaction.
- 15 16. A system comprising:
a memory adapted to hold program code, the program code to include an atomic region with optimized code comprising a conditional commit instruction before an end of the atomic region; and
a processor coupled to the memory, the processor including decode logic adapted to recognize the commit instruction and hardware adapted to determine, in response to the decode logic recognizing the conditional commit instruction, if the atomic region should be committed before the end of the atomic region.
- 20 17. The system of claim 16, wherein hardware adapted to determine, in response to the decode logic recognizing the conditional commit instruction, if the atomic region should be committed before the end of the atomic region comprises the hardware being adapted to track an available amount of a hardware resource in the processor to be utilized to execute the atomic region, to determine an expected amount of the hardware resource to be utilized in executing a portion of the atomic region, and to determine the atomic region should be committed before the end of the atomic region in response to the available amount of hardware resource being less than the expected amount of the hardware resource.
- 30 18. The system of claim 16, wherein the conditional commit instruction includes a reference to an expected amount of a hardware resource to be utilized in executing

a portion of the atomic region, and wherein hardware adapted to determine, in response to the decode logic recognizing the conditional commit instruction, if the atomic region should be committed before the end of the atomic region comprises the hardware being adapted to track an available amount of the hardware resource in the processor to be utilized to execute the atomic region and to determine the atomic region should be committed before the end of the atomic region in response to the available amount of hardware resource being less than the expected amount of the hardware resource.

19. The system of claim 18, wherein the hardware resource is selected from a group consisting of a cache memory, a speculative cache memory, a store buffer, a register file, a speculative register file, and a speculative checkpoint register file.
20. The system of claim 16, wherein the memory is selected from a group consisting of a Synchronous Dynamic Random Access Memory (SDRAM), a Read Only Memory (ROM), and a flash memory.
21. An apparatus comprising:
execution logic adapted to execute a transaction including a loop;
a counter adapted to count a number of iterations of the loop; and
hardware adapted to initiate a commit of a transaction before the end of the transaction in response to the number of iterations reaching an iteration threshold.
22. The apparatus of claim 21, wherein the iteration threshold is to be dynamically adjusted based on execution analysis of the loop.
23. The apparatus of claim 21, wherein the iteration threshold is to be initially set to a default value, to be increased in response to no roll-backs due to hardware constraints before the hardware initiates a commit of the transaction before the end of the transaction in response to the counter reaching the default value, and to be decreased in response to a roll-back due to hardware constraints before the hardware initiates a commit of the transaction before the end of the transaction in response to the counter reaching the default value, and wherein the default value is selected from a group of values consisting of a number of speculative stores in the loop, a number of entries available in a memory device divided by a number of entries estimated to be utilized during an iteration of the loop, a start value from software, and a number of loop iterations before a roll-back due to hardware constraints during a previous execution of the loop.

- 5 24. The apparatus of claim 21, wherein the counter adapted to count a number of iterations of the loop comprises the counter being adapted to be set to a default value and counting down towards zero, and wherein the hardware adapted to initiate a commit of the transaction before the end of the transaction in response to the number of iterations reaching an iteration threshold comprises the hardware adapted to initiate a commit of the transaction before the end of the transaction in response to the counter reaching zero.

CLAIMS

What is claimed is:

1. An apparatus for optimizing code comprising:

a memory to hold program code; and

a processor including,

hardware resources adapted to support transactional execution and to provide a representation of an availability of the hardware resources; and

execution logic adapted to execute the program code to cause the processor to dynamically resize a transactional region including an optimized portion of the program code based on the representation of the availability of the hardware resources.

2. The apparatus of claim 1, wherein execution logic adapted to execute the program code to cause the processor to dynamically resize a transactional region including an optimized portion of the application code based on the representation of the availability of the hardware resources comprises: the execution logic being adapted to execute a conditional commit instruction within the program code that is before an end of the transactional region, the conditional commit instruction to commit the transaction before the end of the transactional region in response to the representation of the availability of the hardware resources indicating insufficient resources to complete execution of the transactional region.

3. The apparatus of claim 2, wherein the memory is further adapted to hold dynamic optimization code, and wherein the execution logic is further adapted to execute the dynamic optimization code to optimize a portion of the code to obtain the optimized portion of the program code including inserting, during runtime, the conditional commit instruction before the end of the transactional region.

4. The apparatus of claim 1, wherein execution logic adapted to execute the program code to cause the processor to dynamically resize a transactional region including an optimized portion of the application code based on the representation of the availability of the hardware resources comprises: the execution logic being adapted to execute a transactional write from the transactional region, wherein the transactional region is to be rolled back to a recent checkpoint and committed in response to the transactional write overflowing a hardware store buffer.

5. The apparatus of claim 1, wherein the memory is selected from a group consisting of an on-processor cache memory, a system memory directly coupled to the processor, and a system memory indirectly coupled to the processor.

6. An apparatus comprising:

decode logic adapted to decode a region check instruction associated with a transaction; and

hardware adapted to support execution of the transaction, wherein the hardware is further adapted to provide a representation of a hardware usage metric to a usage metric storage element in response to the decode logic decoding the region check instruction, wherein the usage metric storage element is adapted to be utilized in determining if a conditional commit of the transaction is to be performed before an end of the transaction.

7. The apparatus of claim 6, wherein the hardware is selected from a group consisting of a speculative cache memory, a store buffer, a speculative register file, and a speculative checkpoint register file, and wherein.

8. The apparatus of claim 6, wherein the region check instruction is to include a representation of an expected usage metric; and wherein the usage metric storage element is adapted to be utilized in determining if a conditional commit of the transaction is to be performed before an end of the transaction comprises the hardware being further adapted to compare the representation of the expected usage metric with the representation of the hardware usage metric to be provided to the usage metric storage element to obtain a usage comparison result; and determining if a conditional commit of the transaction is to be performed before the end of the transaction based on the usage comparison result.

9. The apparatus of claim 8, wherein the region check instruction is further to include a branch target address; the hardware is adapted to perform the determining if the conditional commit of the transaction is to be performed before the end of the transaction based on the usage comparison result utilizing a predefined algorithm, and wherein execution is adapted to jump to the branch target address in response to determining the conditional commit of the transaction is to be performed before the end of the transaction based on the usage comparison result utilizing the predefined algorithm.

10. The apparatus of claim 8, wherein the hardware is adapted to perform the determining if the conditional commit of the transaction is to be performed before the end of the transaction based on the usage comparison result utilizing a predefined algorithm, and to expose a representation of the determining if the conditional commit of the transaction is to be performed before the end of the transaction based on the usage comparison result utilizing a predefined algorithm to conditional code associated with the transaction, the conditional code, when executed, is to:

jump to a conditional commit branch address in response to the hardware exposing a representation of determining that the conditional commit of the transaction is to be performed before the end of the transaction,

commit the transaction; and

start a new transaction.

11. The apparatus of claim 6, wherein the usage metric storage element includes a register that is adapted to be read by conditional code associated with the transaction, the conditional code, when executed, to evaluate the hardware usage metric in regard to a software expected usage metric of hardware, and to determine if the conditional commit of the transaction is to be performed before the end of the transaction based on the evaluation of the hardware usage metric in regard to the software expected usage metric.

12. The apparatus of claim 11, wherein the conditional code, when executed, is further to jump to a commit path, commit the transaction before the end of the transaction, and start a new transaction in response to determining the conditional commit of the transaction is to be performed before the end of the transaction based on the evaluation of the hardware usage metric in regard to the software expected usage metric.

13. An apparatus comprising:

decode logic adapted to decode a conditional commit instruction to be associated with a conditional commit point in a transaction, the conditional commit instruction to reference a destination address; and

hardware adapted to determine, in response to the decode logic decoding the conditional commit instruction, if a hardware resource is to include enough available space to support execution of a region of the transaction and to jump execution to the destination address in response to determining the hardware does not include enough available space to support execution of the region of the transaction.

14. The apparatus of claim 13, wherein the hardware resource includes a memory device, and wherein the hardware is further adapted to determine if the memory device includes enough available space to support execution of the region of the transaction based on an expected number of entries to be utilized before a second conditional commit point in the transaction and a number of available entries in the memory device, wherein the hardware is adapted to determine the memory device does not include enough available space to support execution of the region of the transaction in response to the number of available entries being less than the number of expected entries.

15. The apparatus of claim 13, wherein the conditional commit instruction is further to reference the expected number of entries to be utilized before the second conditional commit point in the transaction.

16. A system comprising:

a memory adapted to hold program code, the program code to include an atomic region with optimized code comprising a conditional commit instruction before an end of the atomic region; and

a processor coupled to the memory, the processor including decode logic adapted to recognize the commit instruction and hardware adapted to determine, in response to the decode logic recognizing the conditional commit instruction, if the atomic region should be committed before the end of the atomic region.

17. The system of claim 16, wherein hardware adapted to determine, in response to the decode logic recognizing the conditional commit instruction, if the atomic region should be committed before the end of the atomic region comprises the hardware being adapted to track an available amount of a hardware resource in the processor to be utilized to execute the atomic region, to determine an expected amount of the hardware resource to be utilized in executing a portion of the atomic region, and to determine the atomic region should be committed before the end of the atomic region in response to the available amount of hardware resource being less than the expected amount of the hardware resource.

18. The system of claim 16, wherein the conditional commit instruction includes a reference to an expected amount of a hardware resource to be utilized in executing a portion of the atomic region, and wherein hardware adapted to determine, in response to the decode logic recognizing the conditional commit instruction, if the atomic region should be committed before the end of the atomic region comprises the hardware being adapted to track an available amount of the hardware resource in the processor to be utilized to execute the atomic region and to determine the atomic region should be committed before the end of the atomic region in response to the available amount of hardware resource being less than the expected amount of the hardware resource.

19. The system of claim 18, wherein the hardware resource is selected from a group consisting of a cache memory, a speculative cache memory, a store buffer, a register file, a speculative register file, and a speculative checkpoint register file.

20. The system of claim 16, wherein the memory is selected from a group consisting of a Synchronous Dynamic Random Access Memory (SDRAM), a Read Only Memory (ROM), and a flash memory.

21. An apparatus comprising:

execution logic adapted to execute a transaction including a loop;

a counter adapted to count a number of iterations of the loop; and

hardware adapted to initiate a commit of a transaction before the end of the transaction in response to the number of iterations reaching an iteration threshold, wherein the iteration threshold is to be initially set to a default value, to be increased in response to no roll-backs due to hardware constraints before the hardware initiates a commit of the transaction before the end of the transaction in response to the counter reaching the default value, and to be decreased in response to a roll-back due to hardware constraints before the

hardware initiates a commit of the transaction before the end of the transaction in response to the counter reaching the default value, and wherein the default value is selected from a group of values consisting of a number of speculative stores in the loop, a number of entries available in a memory device divided by a number of entries estimated to be utilized during an iteration of the loop, a start value from software, and a number of loop iterations before a roll-back due to hardware constraints during a previous execution loop.

22. The apparatus of claim 21, wherein the iteration threshold is to be dynamically adjusted based on execution analysis of the loop.

23. The apparatus of claim 21, wherein the counter adapted to count a number of iterations of the loop comprises the counter being adapted to be set to a default value and counting down towards zero, and wherein the hardware adapted to initiate a commit of the transaction before the end of the transaction in response to the number of iterations reaching an iteration threshold comprises the hardware adapted to initiate a commit of the transaction before the end of the transaction in response to the counter reaching zero.