What is claimed :

1. A method, implemented at a computer system that includes at least one processor, for indexing a trace with a diffgram, the method comprising:

accessing a replayable trace that records a prior execution of one or more threads of an entity;

identifying, within a trace segment recording a prior execution of a sequence of a plurality of instructions of a first thread of the one or more threads, a subset of the sequence of instructions, the subset of instructions beginning with a first instruction and ending with a second instruction ;

replaying the subset of instructions to obtain state of one or more named resources, the one or more named resources having a first state at the first instruction and a different second state at the second instruction;

based on replaying the subset of instructions, generating a diffgram for the subset of instructions, the diffgram comprising one or more reversible mathematical operations on the first state that transform the first state into the second state, such that addition of the diffgram at the first instruction by applying the one or more reversible mathematical operations to the first state brings the one or more named resources to the second state, and subtraction of the diffgram at the second instruction by applying a reversal of the one or more reversible mathematical operations to the second state brings the one or more named resource to the first state; and

storing the diffgram in connection with the trace.

2. The method of claim 1, storing the diffgram in connection with the trace comprising at least one of (i) storing the diffgram into durable storage associated with the trace, or (ii) storing the diffgram into volatile memory associated with replay of the trace.

3. The method of claim 1 , the diffgram using a key frame as a baseline, addition of the diffgram to the key frame bringing the one or more named resources to the second state, and subtraction of the diffgram at the second instruction bringing the one or more named resources to a state stored in the key frame.

4. The method of claim 3, wherein the state stored in the key frame comprises the first state.

5. The method of claim 4, wherein the key frame is associated with the trace segment.

6. The method of claim 1, wherein the second state includes a state change made by a second thread of the plurality of threads after the first instruction.

7. The method of claim 1, wherein the second state omits a state change made by a second thread of the plurality of threads after the first instruction.

8. The method of claim 1, the subset of instructions comprising a first subset of nstructions, and the diffgram comprising a first diffgram, the method further comprising:

identifying, within the trace segment, second a subset of the sequence of instructions, the second subset of instructions beginning with a third instruction and ending with a fourth instruction

replaying the second subset of the sequence of instructions to obtain state of the one or more named resources, the one or more named resources having a third state at the third instruction and a different fourth state at the fourth instruction; and

based on replaying the second subset of instructions, generating a second diffgram for the second subset of instructions, the second diffgram being structured such that addition of the diffgram at the third instruction brings the one or more named resources to the fourth state, and subtraction of the diffgram at the fourth instruction brings the one or more named resources to the third state.

9. The method of claim 8, the second diffgram using a key frame as a baseline, subtraction of the second diffgram at the fourth instruction bringing the one or more named resources to a state stored in the key frame.

10. The method of claim 8, the second diffgram using the first diffgram as a baseline, and subtraction of the second diffgram at the fourth instruction bringing the one or more named resources to the third state.

11. The method of claim 8, addition of the first and second diffgrams at the first instruction bringing the one or more named resources to the fourth state, and subtraction of the first and second diffgrams at the fourth instruction bringing the one or more named resources to the first state.

12. A method, implemented at a computer system that includes at least one processor, for utilizing a diffgram to replay a trace, the method comprising:

accessing a replayable trace that records a prior execution of one or more threads of an entity;

identifying, within the replayable trace, a particular instruction within a trace segment recording a prior execution of a sequence of a plurality of instructions of a first thread of the one or more threads, the plurality of instructions including the particular instruction;

identifying a set of one or more diffgrams for restoring a first state of one or more named resources that is needed for emulating the prior execution of the particular instruction, the set of one or more diffgrams comprising one or more reversible mathematical operations on the first state that transform the first state into a second state, such that addition of the diffgram by applying the one or more reversible mathematical operations to the first state brings the one or more named resources to the second state, and subtraction of the diffgram by applying a reversal of the one or more reversible mathematical operations to the second state brings the one or more named resources to the first state ;

applying the set of one or more diffgrams to restore the first state of the one or more named resources; and

after applying the set of one or more diffgrams, emulating the prior execution of the particular instruction based on the restored first state of the one or more named resources.

13. The method of claim 12, applying the set of one or more diffgrams to restore the first state of the one or more named resources comprising adding the set of one or more diffgrams to a state of the one or more named resources that is obtained from a key frame associated with the trace segment, or within another trace segment.

14. The method of claim 12, applying the set of one or more diffgrams to restore the first state of the one or more named resources comprising adding the set of one or more diffgrams to a state of the one or more named resources that is obtained from another diffgram that restores state to an instruction that is at, or prior to, the particular instruction in the sequence of instructions.

15. The method of claim 12, applying the set of one or more diffgrams to restore the first state of the one or more named resources comprising subtracting the set of one or more diffgrams from a state of the one or more named resources at an instruction that is subsequent to the particular instruction in the sequence of instructions.

16. The method of claim 12, applying the set of one or more diffgrams to restore the first state of the one or more named resources comprising adding a plurality of diffgrams to a restored prior state of the one or more named resources, or subtracting the plurality of diffgrams from the restored prior state of the one or more named resources.

17. The method of claim 12, at least one diffgram being obtained from the trace, or from having previously replayed the trace segment.

18. The method of claim 12, further comprising identifying, from the set of one or more diffgrams, one or more named resources that were modified by instructions covered by the set of one or more diffgrams.

19. The method of claim 12, wherein applying the set of one or more diffgrams to restore the first state of the one or more named resources comprises restoring the first state of the one or more named resources without emulating any instructions.

20. A computer system comprising :

a processor, and

a computer-readable medium having stored thereon computer-executable instructions that are executable by the processor to cause the computer system to generate and utilize diffgrams during trace replay, the computer-executable instructions including instructions that are executable by the processor to cause the computer system to at least :

access a replayable trace that records a prior execution of an entity;

identify, within a trace segment recording a prior execution of a sequence of a plurality of instructions of the entity, a subset of the sequence of instructions, the subset of instructions beginning with a first instruction and ending with a second instruction;

replay the subset of instructions to obtain state of one or more named resources, the one or more named resources having a first state at the first instruction and a different second state at the second instruction;

based on replaying the subset of instructions, generate a diffgram for the subset of instructions, the diffgram comprising one or more reversible mathematical operations on the first state that transform the first state into the second state, such that addition of the diffgram at the first instruction by applying the one or more reversible mathematical operations to the first state brings the one or more named resources to the second state, and subtraction of the diffgram at the second instruction by applying a reversal of the one or more reversible mathematical operations to the second state brings the one or more named resources to the first state;

subsequent to generating the diffgram, identify, within the replayable trace, a particular instruction within the subset of instructions that is to be replayed;

identify the generated diffgram for reaching the particular instruction;

apply the diffgram at the second instruction to restore the first state of the one or more named resources;

and

after applying the diffgram at the second instruction, replay the subset of instructions, including replaying the particular instruction based on the restored first state of the one or more named resources.