What is claimed is:

1. A method comprising:

accumulating a reduced representation of geometry, input data, and state for tile when a triangle list is created;

before per tile rasterization, determining whether the representation for the current tile in the current frame is the same as the representation for the previous frame and the corresponding tile; and

if the representations for the tile in previous and current frames are the same, then reusing the prior tile's pixels as the current tile's pixels when rendering the current tile.

2. The method of claim 1 including using a hash function to produce the reduced representation.

3. The method of claim 1 including using a bloom filter to produce the reduced representation.

4. The method of claim 3 including using two bloom filters, one for the current tile and one for a previous tile.

5. The method of claim 4 including using one pair of bloom filters for each of geometry, texture addresses, and shaders.

6. The method of claim 1 including accumulating in a sort-middle architecture.

7. The method of claim 1 wherein reusing includes reusing the color values.

8. The method of claim 1 wherein reusing includes reusing depth or stencil buffers.

9. The method of claim 1 including refraining from sending data for display when the compact representations are the same.

10. The method of claim 1 including changing the current compact representation to the previous compact representation.

11. The method of claim 1 including determining if the same draw call is used for the same tile in successive frames to determine if the representations are the same.

12. One or more non-transitory computer readable media storing instructions to implement a sequence comprising:

accumulating a reduced representation of geometry, input data, and state for tile when a triangle list is created;

before per tile rasterization, determining whether the representation for the current tile in the current frame is the same as the representation for the previous frame and the corresponding tile; and

if the representations for the tile in previous and current frames are the same, then reusing the prior tile's pixels as the current tile's pixels when rendering the current tile.

13. The media of claim 12, said sequence including using a hash function to produce the reduced representation.

14. The media of claim 12, said sequence including using a bloom filter to produce the reduced representation.

15. The media of claim 14, said sequence including using two bloom filters, one for the current tile and one for a previous tile.

16. The media of claim 15, said sequence including using one pair of bloom filters for each of geometry, texture addresses, and shaders.

17. The media of claim 12, said sequence including accumulating in a sort-middle architecture.

18. An apparatus comprising:

a processor to accumulate a reduced representation of geometry, input data, and state for tile when a triangle list is created, before per tile rasterization, determine whether the representation for the current tile in the current frame is the same as the representation for the previous frame and the corresponding tile, and if the representations for the tile in previous and current frames are the same, then reuse the prior tile's pixels as the current tile's pixels when rendering the current tile; and

a storage coupled to said processor.

19. The apparatus of claim 18, said processor to use a hash function to produce the reduced representation.

20. The apparatus of claim 18, including a bloom filter to produce the reduced representation.

21. The apparatus of claim 20 including two bloom filters, one for the current tile and one for a previous tile.

22. The apparatus of claim 21 including one pair of bloom filters for each of geometry, texture addresses, and shaders.

23. The apparatus of claim 18, said processor to accumulate in a sort-middle architecture.

24. The apparatus of claim 18, said processor to refrain from sending data for display when the compact representations are the same.

25. The apparatus of claim 18, said processor to determine if the same draw call is used for the same tile in successive frames to determine if the representations are the same.

What is claimed is:

1. A method comprising:

accumulating a reduced representation of geometry, input data, and state for tile when a triangle list is created;

before per tile rasterization, determining whether the representation for the current tile in the current frame is the same as the representation for the previous frame and the corresponding tile; and

if the representations for the tile in previous and current frames are the same, then reusing the prior tile's pixels as the current tile's pixels when rendering the current tile.

2. The method of claim 1 including using a hash function to produce the reduced representation.

3. The method of claim 1 including using a bloom filter to produce the reduced representation.

4. The method of claim 3 including using two bloom filters, one for the current tile and one for a previous tile.

5. The method of claim 4 including using one pair of bloom filters for each of geometry, texture addresses, and shaders.

6. The method of claim 1 including accumulating in a sort-middle architecture.

7. The method of claim 1 wherein reusing includes reusing the color values.

8. The method of claim 1 wherein reusing includes reusing depth or stencil buffers.

9. The method of claim 1 including refraining from sending data for display when the compact representations are the same.

10. The method of claim 1 including changing the current compact representation to the previous compact representation.

11. The method of claim 1 including determining if the same draw call is used for the same tile in successive frames to determine if the representations are the same.

12. One or more non-transitory computer readable media storing instructions to implement a sequence comprising:

accumulating a reduced representation of geometry, input data, and state for tile when a triangle list is created;

before per tile rasterization, determining whether the representation for the current tile in the current frame is the same as the representation for the previous frame and the corresponding tile; and

if the representations for the tile in previous and current frames are the same, then reusing the prior tile's pixels as the current tile's pixels when rendering the current tile.

13. The media of claim 12, said sequence including using a hash function to produce the reduced representation.

14. The media of claim 12, said sequence including using a bloom filter to produce the reduced representation.

15. The media of claim 14, said sequence including using two bloom filters, one for the current tile and one for a previous tile.

16. The media of claim 15, said sequence including using one pair of bloom filters for each of geometry, texture addresses, and shaders.

17. The media of claim 12, said sequence including accumulating in a sort-middle architecture.

18. The media of claim 12 wherein reusing includes reusing the color values.

19. The media of claim 12 wherein reusing includes reusing depth or stencil buffers.

20. The media of claim 12, said sequence including determining if the same draw call is used for the same tile in successive frames to determine if the representations are the same.

21. An apparatus comprising:

a processor to accumulate a reduced representation of geometry, input data, and state for tile when a triangle list is created, before per tile rasterization, determine whether the representation for the current tile in the current frame is the same as the representation for the previous frame and the corresponding tile, and if the representations for the tile in previous and current frames are the same, then reuse the prior tile's pixels as the current tile's pixels when rendering the current tile; and

a storage coupled to said processor.

22. The apparatus of claim 21, said processor to use a hash function to produce the reduced representation.

23. The apparatus of claim 21, including a bloom filter to produce the reduced representation.

24. The apparatus of claim 23 including two bloom filters, one for the current tile and one for a previous tile.

25. The apparatus of claim 24 including one pair of bloom filters for each of geometry, texture addresses, and shaders.

26. The apparatus of claim 21, said processor to accumulate in a sort-middle architecture.

27. The apparatus of claim 21, said processor to refrain from sending data for display when the compact representations are the same.

28. The apparatus of claim 21, said processor to determine if the same draw call is used for the same tile in successive frames to determine if the representations are the same.

29. The apparatus of claim 21 including a battery.

30. The apparatus of claim 21 including firmware and a module to update said firmware.