CLAIMS:

1. A processor that loads data elements, comprising: an upper level cache; and at least one processor core coupled to the upper level cache, including one or more registers and a plurality of instruction processing stages: a decoder unit to decode an instruction requiring an input of a plurality of data elements, wherein a size of each of the plurality of data elements is less than a cache line size of the processor, and responsive to the instruction, an execution unit to load the plurality of data elements to the one or more registers, without loading data elements spatially adjacent to the plurality of data elements, wherein the loading of the plurality of data elements is to: gather the plurality of data elements in a temporary buffer; and load the plurality of data elements from the temporary buffer to the one or more registers.

2. The processor of claim 1, wherein the instruction is specifically for execution of data elements without considering spatial locality.

3. The processor of claim 2, wherein the instruction specifies a stride value between data elements of the plurality of data elements.

4. The processor of claim 3, the stride value is bigger than a cache line size of the processor.

5. The processor of claim 2, wherein the instruction specifies an index corresponding to locations of each of the plurality of data elements.

6. A method for loading data elements in a processor, comprising: decoding an instruction, wherein the instruction requires an input of a plurality of data elements, wherein a size of each of the plurality of data elements is less than a cache line size of the processor; and after decoding the instruction, responsive to the instruction, loading the plurality of data elements to one or more registers of the processor for execution, without loading data elements spatially adjacent to the plurality of data elements, wherein the loading of the plurality of data elements comprises: gathering the plurality of data elements in a temporary buffer; and loading the plurality of data elements from the temporary buffer to the one or more registers.

7. The method of claim 6, wherein the instruction is specifically for execution of data elements without considering spatial locality.

8. The method of claim 7, wherein the instruction specifies a stride value between data elements of the plurality of data elements.

9. The method of claim 8, wherein the stride value is bigger than a cache line size of the processor.

10. The method of claim 7, wherein the instruction specifies an index corresponding to locations of each of the plurality of data elements.

11. A computer system that loads data elements, comprising: a memory for storing instructions; and a processor including: an upper level cache; at least one processor core coupled to the upper level cache, including one or more registers and a plurality of instruction processing stages: a decoder unit to decode an instruction, wherein the instruction requires an input of a plurality of data elements, wherein a size of each of the plurality of data elements is less than a cache line size of the processor, and responsive to the instruction, an execution unit configured to load the plurality of data elements to the one or more registers, without loading data elements spatially adjacent to the plurality of data elements, wherein the loading of the plurality of data elements is to: gather the plurality of data elements in a temporary buffer; and load the plurality of data elements from the temporary buffer to the one or more registers.

12. The computer system of claim 11, wherein the instruction is specifically for execution of data elements without considering spatial locality.

13. The computer system of claim 12, wherein the instruction specifies a stride value between data elements of the plurality of data elements.

14. The computer system of claim 13, the stride value is bigger than a cache line size of the computer system.