Non von Neumann Architectures
The "von Neumann" in von Neumann architecture refers to Hungarian-American mathematician John von Neumann, Von Neumann became a consultant to the E NIAC (Electronic Numerical Integrator and Computer) project, which upon its completion in 1945 became the world's first general purpose, electronic computer.

This computer architecture uses a computer program stored in memory to control a computer.

A von Neumann Architecture computer has five parts: an arithmetic-logic unit, a control unit, a memory, some form of input/output and a bus that provides a data path between these parts.
The Von Neumann Architecture

Memory

Processor (CPU)
- Control Unit
  - ALU
  - Do arithmetic/logic operations requested by program

Input-Output

Bus

- Communicate with "outside world", e.g.
  - Screen
  - Keyboard
  - Storage devices
  - ...

Store data and program

Execute program
Characteristics (von Neumann’s Architecture)

- Both data and instructions (control sequences) are stored in a single read-write memory. (Cannot tell the difference between data and instructions by examining a memory location)
- Memory contents are addressable by location without regard for the type of data contained there
- Execution occurs in a sequential fashion by reading consecutive instructions from memory
von Neumann or control flow computing model:

• A program is a series of addressable instructions, each of which either specifies an operation along with memory locations of the operands or it specifies (un)conditional transfer of control to some other instruction.
• Essentially: the next instruction to be executed depends on what happened during the execution of the current instruction.
• The next instruction to be executed is pointed to and triggered by the PC.
What is the principal contribution?

- To von Neumann, the key to building a general purpose device was in its ability to store not only its data and the intermediate results of computation, but also to store the instructions, or orders, that brought about the computation.
- In a special purpose machine the computational procedure could be part of the hardware.
- In a general purpose one the instructions must be as changeable as the numbers they acted upon.
- Therefore, why not encode the instructions into numeric form and store instructions and data in the same memory?
- This frequently is viewed as **the principal contribution provided by von Neumann's insight into the nature of what a computer should be.**
• He then defined the control organ as that which would automatically execute the coded instructions stored in memory.

• Interestingly he says that the orders and data can reside in the same memory "if the machine can in some fashion distinguish a number from an order."

• And yet, there is no distinction between the two in memory. The control counter (what we now usually call the program counter) contains the address of the next instruction, and that word is fetched to be executed.
Von Neumann bottleneck

- The separation between the CPU and memory leads to the von Neumann bottleneck, the limited throughput (data transfer rate) between the CPU and memory compared to the amount of memory.
- In modern machines, throughput is much smaller than the rate at which the CPU can work.
- This seriously limits the effective processing speed when the CPU is required to perform minimal processing on large amounts of data.
- The CPU is continuously forced to wait for vital data to be transferred to or from memory.
- As CPU speed and memory size have increased much faster than the throughput between them, the bottleneck has become more of a problem.
• As speeds have increased, the processor has spent an increasing amount of time idle, waiting for data to be fetched from memory.
• No matter how fast a given processor can work, in effect it is limited to the rate of transfer allowed by the bottleneck.
• Often, a faster processor just means that it will spend more time idle.
• The von Neumann bottleneck has often been considered a problem that can only be overcome through significant changes to computer or processor architectures.
Why need non Von Neumann architecture

• Instructions and data are distinguished only implicitly through usage.
• The memory is a single memory, sequentially addressed.
• The memory is one-dimensional, these are in conflict with our programming languages
• It is not possible to tell by looking at a set of bits whether that set of bits represents an integer, a floating point number or a character string