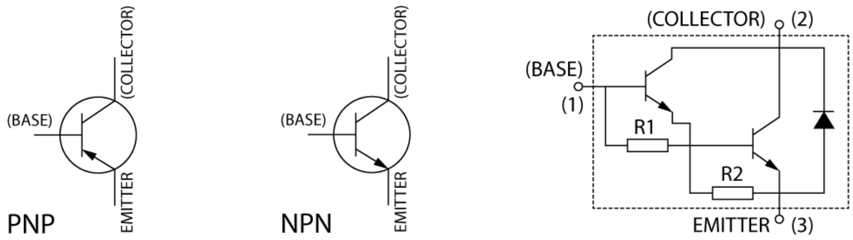
eng.time_for_d

Alexandr Kirilov (https://github.com/alexandrkirilov)

Time for D? 2D? 3D? Even 500D or nD?

From the time of inventing transistor nothing in principle new has developed yet. All of further inventions have been inheriting the transistor. It was HUGE technological breakthrough.



The transistor related technology for now starting to hinder global technological progress. If there will not be act for improving technology this kind of affection will be more critical. Why? The reply on this question might be found in the architectural explanation of HDD or RAM procedures of writing/reading data.

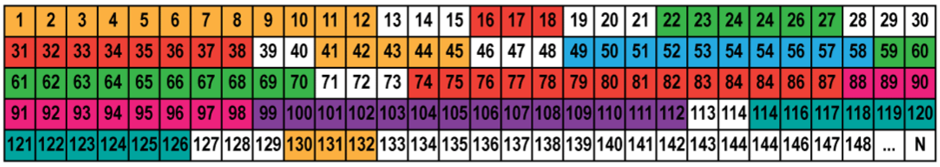
To be detailed of this technology - not the goal of this article. You might to find it by your own in the Internet at least by visiting Wikipedia:

- Memory Storage
- HDD

For being brief - all of this kind of devices using statical linear physical algorithms and architecture for storing and operating data and it's no any abilities to be adopted for the data that going to be stored or operated. This is the main problem and huge bottleneck:

- in case of RAM you have the chain of transistors in specially organised order and you have no any ability to reorganise it by depend in case of necessity
- in case of HDD you have the chain of sectors within sector's magnetic field value and you have no any abilities to reorganise the sectors within data, you capable only reorganise data by changing value of magnetic field
- etc

The list of points might be prolonged further and further. The global problem looks like: we have no ability to reorganise or adopt physically the data storage for the nature of data that we re going to operate. For the RAM - transistors, for the HDD - sectors. For the case of Big-Data handling it's HUGE problem.



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	24	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	54	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	84	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	114	116	117	118	119	120
121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	144	146	147	148	...	N

For illustrating this problem there are two examples:

- Just imagine the spiral staircase inside of skyscraper within huge numbers of floors, where every floor is one office. You are the courier inside of this building and got a task to collect the personal signs of the group on different floors. One group is on 5,34,190,1113, another on 5,78,1,23456,34,7819. And now - just notice how many floors you need to pass away without paying any attention for getting floor that you need. Have you got this imagination? This process is similar to what is going on inside of HDD.
- Just imagine the Christmas, you need to switch on garland. For making it possible you are plugging the lamps into slots in special order that is required. At time of finishing this process, you've plugged in 1234 lamps and you forgot to plug one green lamp between 35 and 36 lamps. As a result of this mistake you need to plug out 1199 lamps, plug in one green lamp and again plug in 1199 in the special order for getting 1235 lamps in garland. This process is similar to what is going on in RAM.

Why this problem became so critical today? Because for now the human beings started to operate huge amount of data and this amount going to be much more, but the technology hasn't been improving on physical level a long time. 10-15 years ago the HDD of 1Tb of capacity was something fantastical - for now 5-10Tb for home usage is normal. For corporate storage thousands of terabytes is normal and increasing further.

How to solve this problem? For deeper understanding the nature of this problem, developers shell/should/have to change the measure system in mind:

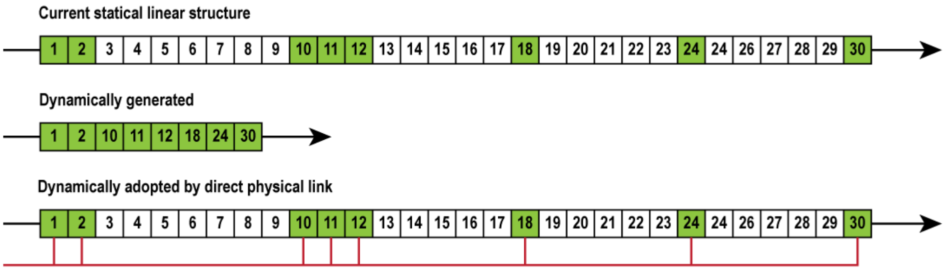
- **The first and most important point: every data carrier will always require**

the format that is based on physical abilities of carrier and statical linear architecture is only one from huge variety and without upgrading, developing, improving of data carriers physical abilities we going to stubbed by our own, everything has limit, even transistor based technology.

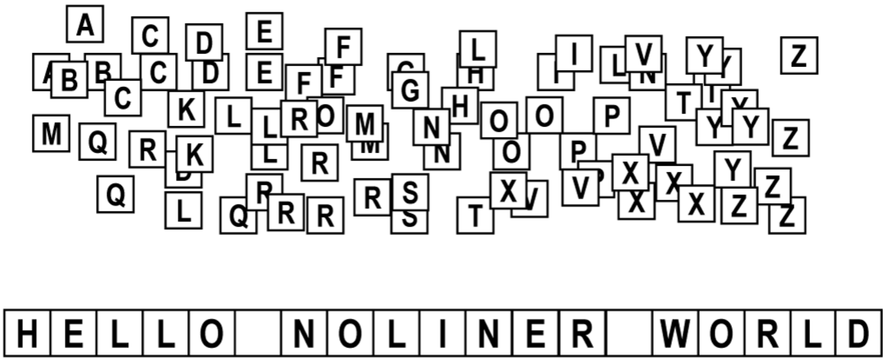
- **Shift the focus out of developing "based on transistors" and be focused on carriers new physicals abilities that is improving informational abilities itself to describe events and processes**: for now we have tremendous list of code library that is developed 20-30 years ago and have no intention to be rewritten only because they are very optimal for this physical platform
- **Have to understand the place of IT industry in a global schema around, the IT presented only because of necessity from another fields of human beings actions and stop developing "yet-another-super-progressive-language" that is based on the C and ASM only**: for IT industry is looking like overheated engine that is closely to maximum of abilities but the current location is very very far from finish but resources closely to the end, there are created a lot of things - but everything is working on transistors

There are might be added few points more, but they are not so important in context of this article. Most important - this 3 points.

Just after changing our understanding of what we are naming like IT will be gotten free resources that might be focused on something very new and important - for example nonlinear (vector) memory architecture that will have the ability to be adopted automatically for the data or data size that going to be stored and operated by making direct-one-step links between the objects that will exclude the necessity "to pass away a lot of floors". This idea on architectural level directly opposite towards we are doing right now - we are adopting data towards the carrier and always limited by carrier itself.



We have to develop the mechanism of data storing and operating, that will looking like the numbers of empty memory cells that is produced unlinked before or equipped by sub mechanism to produce the memory cells by demand in following of data architecture that we going to store. Or at least developing on-step-links between cells and by this excluding "passing away the floors"



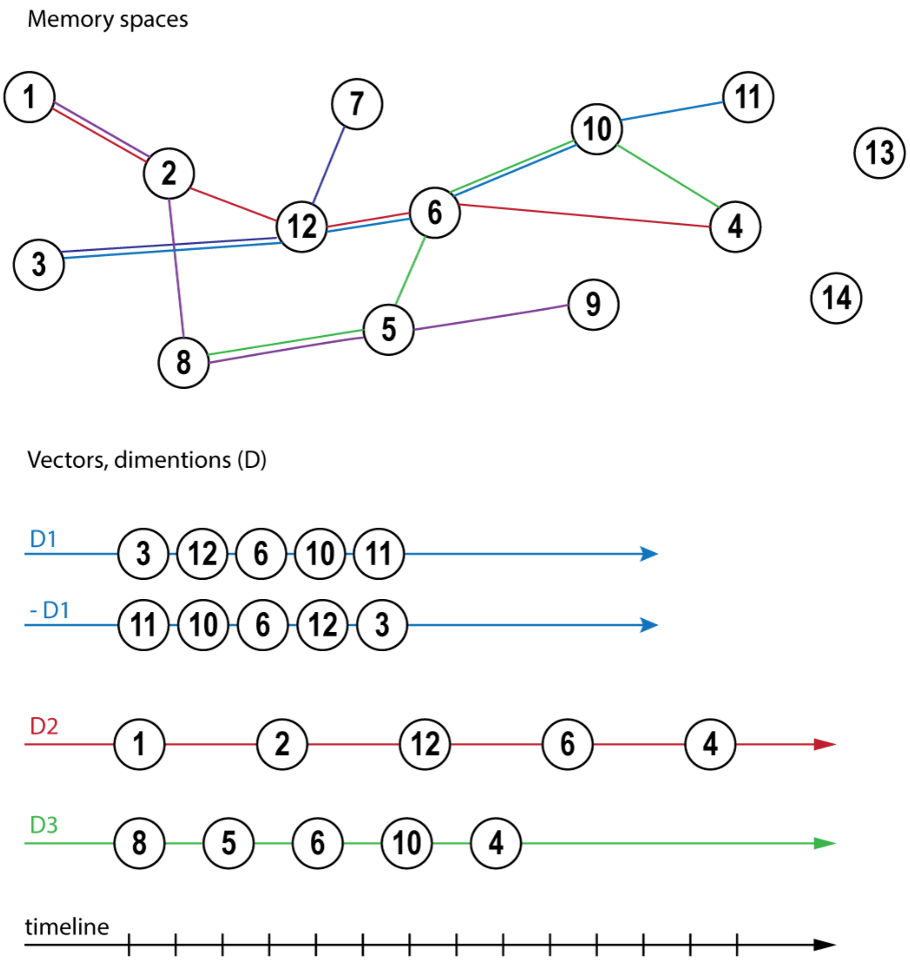
One of illustration of this mechanism might be childish game "Words" when the child is making words by letters in case of studying grammar. The child own idea or the idea of parents that teaching him creating the structure of word and the child unfollowing this structure composing the word from free cubes within letters and starting to use this numbers of letter like on - the word for making sentences. The process of uniting numbers of elements to one solid described in article "Understanding blockchain". Sometimes the meaning of word might be changed by context that is the sentence where this word used.

After defining the problem, question - with materials will allow to produce this king of mechanism? For getting reply on it we have to look around - the organic cells. There proven scientifically fact that the organic cell has the memory. Beside all of it organic cell might be created by demand in following different architectures. Why we are not focused on it? For example this kind of tech will allow to produce memory cell from elements in liquid by demand and by electromagnetic field of special frequency and in case of not using it - to decompose it by high voltage for the level of elements that will allow to produce new one in following another architecture. Partly this approach described in article "Frequency"

If you are going to draw this schema - it will looks like Neuron links inside of brain - the objects that connected to each other by links and through this links passing electricity.

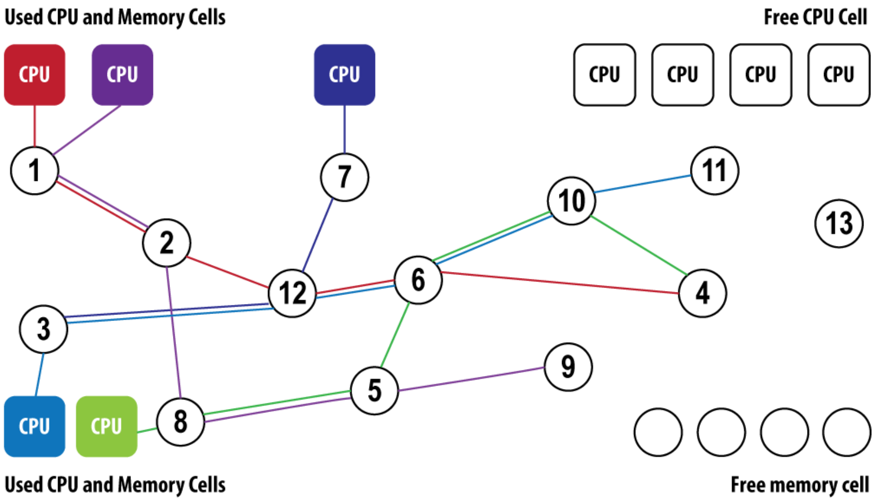
For now - a little futurology: it would be funny to came to the computer store and ask about 3 litres of memory for your computer, but for the future it might normal question.

Why this article named like "Time for D? 2D? 3D? Even 500D or nD?"? The D - is from dimension, for being precise, because of using vectors.



The image above is the explanation of the cause of why this article named how it named.

We have memory space where grown the memory cells that is donating the objects. The objects in memory space is constants and do not require any additional calculation for organising objects (moving, recalculating and etc). There are only one thing is going over there - making links between the objects in following specially defined order, for each vector one order. In context of vector the memory allocated space is not bigger or less then it required by vector linear order. Yes we've turned back to linear schema but equipped by ability to reorder it in case of necessity by very less calculations then it was before.



If you will connect the CPU-cell to every vector that is producing electromagnetic energy by specially adjusted frequency (the schema of adjusting might be gotten from cell phone networks, where defined range of frequencies and every user using one from it) and only in amount that is required by vector you will increase power efficiency and avoid the collusion at one moment of time for using one object. If there will be performed actions for analysing objects structure might be possible to define kind of "Mendeleev Periodic Table" of objects that we are using in case of describing events and actions - kind of organic cells libraries that will require you only do the links between them. For the case of big-data optimisation might be very very important.

This kind of technology has huge list of advantages, but this advantages for being in real world have to created first.

Is there time for D?