The following table presents a comparison of ERA project proposal with INFSO 'research challenges for computing systems' - ICT work programme 2009-2010, document of EC, dated 10th Nov 2007, Braga.

INFSO document will be commented statement-by-statement to clarify meaning and relevance and what is tangible there.

EC INFSO program vs. ERA

IFSO document	Comment	ERA proposal
p.7: 'Over decades ... Moore law has driven evolution of computing architectures...'	Nature and electronics at the technology level do not follow Moore or anybody else. Citing of irrelevant, clarifies nothing. Pseudoscience.	
p.7: 'the smaller the feature size of the components, the more power they leak'	Wrong statement. Power consumption is one story, performance is another. Minimum energy per elementary calculations was studied in 60's by Fradkin (Fradkin gates) and indicated achievable powerscale per calculation, including positioning technology limits vs theoretical limits.	
p.7: 'computing has about the same carbon footprint as commercial aviation'	Over-estimated in order of magnitude, on purpose to puff cheeks, using buzzword greening. Shame.	
p.8: '… designing reliable systems from unreliable components will become a key design technique'	No matter Neuman, or Weiner claimed this in 1950s - it is wrong statement and wrong direction. Read details, ' Reliability of Malfunction Tolerance '.	Combination and use of information, structure and time redundancies is a real way to make efficient systems.
p.9: 'parallel computing is based upon the concept of increasing performance by using division of labour among number of processors'	Ignored program structure limitations and complexity of data organization; they set the real limits.	
p.9: 'in future parallelism is the only way forward to continue the exponential performance growth'	Nothing near to the truth. Von Neuman organization of calculus is the limit; again Moore guess has nothing in connection with real physical phenomena involved.	Design of complex tools for simple object organizations and dependencies is excluded.
p.9: 'the future has already begun'	All examples are about SIMD or DSP, narrow areas, not able to generalize or flexible use or breakthrough.	Flexibility of monitoring of redundancy is the core.
p.10: 'a software developer who has been trained to think parallel'	Application software features are independent to the features of hardware availability and configuration. Parallel thinking is limitation by itself. World record of juggling is ...12 rings.	Special re-processing procedure for sequential program is proposed using several new models.
p.10: 'due to Amdahl law...'	It is not law at all. It is limitation of Von Neuman architecture use.	New data structures and theoretical models.
p.13: 'UK based ARM sold almost 3Bn microprocessors...'	Market domination and right things have nothing in common. See ONBASS 28/11/7 presentation with comparison of ARM, Intel and ERA performance.	Best existing theory (BET) will be implemented with best existing practice (BEP) this is called progress.

IFSO document	Comment	ERA proposal
p.14: 'Java based platform independent software development, network on chip...,'	Java RT? With hardware supported interpretation? Check Marchuk theorem about complexity of formal model transformations - every layer adds on exponentially in killing performance. NOC? Transactions decoding included in every processing? Nonsense!	Consistent development of application software, system software and hardware to use redundancy introduced at maximum efficiency.
p.15: '... EU program are backed by major commercial players'	This is killing any innovation in Europe actually. Funding by EC further domination of US industry? Something strange. Besides NIH syndrome will kill everything alive.	
p.15: '... performance, power and reliability will then be expected to be at forefront'	Symptoms declared as cure. Active redundancy applied for re-configurability is the core.	Read ERA proposal
p.16: '... containing cores optimized for a particular task or set of tasks'	Dynamic optimization? Since 1960s have not been effective? Tuning of the structure kills all benefit of parallelism.	
p.16: '... how to build reliable systems from unreliable components'	Again, Weiner was right on the paper, in real world of computer systems please read 'Reliability of Malfunction tolerance', the problem declared in EC program DOES NOT EXIST.	ERA proposes design of the elements with maximum efficiency at minimum redundancy - to be patented.
p.17: Figure 2 power, performance, reliability	- Three goals? Shooting three ducks at once? Ignoring philosophical principles, contradicts Redundancy classification for fault tolerant computer system.	Active redundancy introduction and monitoring to apply for SW and HW.
p.18: 'beyond static auto-parallelization ... run-time information use...'	Dynamic optimization? Have not been efficient yet (since 1950s). Tuning and rearrangements of resources structure kills all benefit.	Separated phases of software design, then recompilation then adjustment for existing hardware resources.
p.18: 'new programming languages to express computation in parallel.....'	Wrong statement. Hidden parallelism, hardware resources available will be ignored.	See ERA proposal. There is NO OTHER OPTION.
p.18: 'new support environment'	Human beings think sequentially, and still do mistakes. Making this process more complex? Unlikely that it is future.	Only sequential programming with efficient language and rum time system for user and then reprocessing to HW "friendly form".
P.18: 'the outcome of research...'	Voluntarism.	
p.19: 'continuous adaptation'	The whole page is wrong. To pick one: machine learning optimization? It is NP problem! In real time?	READ ERA proposal.
p.20: hardware/software support for dynamic instrumentation and optimization	Complexity of dynamic optimization was proven to be wrong even for single processor systems. TLB ruined performance and increased complexity of VAX, old and new Intel and other architectures.	
p.21: 'machine learning use or artificial intelligence for optimization of compilation'	Complexity of compilation will be increased astronomically. Hardware structure will be more complex, no need.	Different phases of compilation developed especially for the purpose solve the problem of fast execution.
p.23: 'all bullet points on the page'	Function of the system is the key, nothing else. Meta-structure is required, rigorously defined, which will be a core to derive structures of HW, or SW.	Redundancy theory, GLM; <C,D,P>, models will be applied for rigorous design of architecture and system software, to maximize gain.