Discussion Questions

a) Applications: What are the applications that motivate future systems? What advances in algorithms and programming systems will support these applications?

Extracting meaningful/actionable information from data (analytics, IoT, drones), personalized health (sensing, diagnosis, treatment), security & privacy, virtual/augmented reality. Programming systems must be targeted at domain experts not computer experts; however, there is also a need for programming/runtime systems that can bridge the gap from domain experts down to various hardware substrates.

b) Systems: What hardware architectures and distributed systems will support future applications? What challenges do we face in design and management?

Hardware architectures are already evolving to the point of system-on-chip style designs with several different interacting specialized IP blocks. This trend will continue with the addition of configurable systems across many scales from traditional FPGA style LUTs to composition of general purpose and specialized cores (with new technologies) all the way to composition of compute units that are local vs. remote (i.e., in the cloud). This multi-scale use of resources presents challenges in programming abstractions and runtime resource management. The challenge is to go from a high-level specification of the problem and use both static and dynamic analysis to determine the best configuration for executing the workload in terms of appropriate metrics (e.g., energy, power, deadlines, throughput, etc.) while also optimizing across many workloads that share the computational fabric.

c) Technologies: What emerging technologies will change fundamental assumptions in hardware and software? What constraints disappear? What challenges arise?

Technologies can be divided into those with the potential for evolutionary change and those with the potential for revolutionary change. The computing research community must strike a balance between evolutionary technologies that produce incremental but important advances and revolutionary technologies that may offer tremendous advances but come with significant risk of failure. Evolutionary technologies include SoC style integration of general-purpose cores with throughput accelerators, configurable logic, and specialized cores. This will remove the constraints imposed by discrete components such as latency, cost, and may ease programming by allowing new shared memory models.

Molecular-scale on-chip photonics is an emerging technology that may enable larger steps forward. It can be used for communication to reduce latency across chip and to support efficient multi-cast and broadcast. The technology can also be used to create high-density storage, providing potential and specialized compute units to support probabilistic
computing applications. The challenge/limitations of this technology include integration with CMOS fabrication, limited number of useful wavelengths given currently available molecules, the size of the application class that can

**d) Methodologies:** How should we perform interdisciplinary research that spans applications, systems, and technologies? How should abstraction layers evolve?

Look to solve end-to-end problems that by their very nature require crossing many abstraction boundaries. Maintaining abstractions is powerful and has served the community well; however, if we don't consider breaking abstractions we are destined to incremental advances. Truly disruptive advances occur by tossing out old abstractions if needed; abstractions need to efficiently support mapping from high-level specifications by a domain expert to hardware configurations and may include adaptive algorithms, compilation, runtime and operating systems, hardware, etc.

**e) Risks:** What are the risks that threaten the success of XPS research directions? How do we guard and hedge against these threats?

- Being too conservative that only small incremental advances are achieved.
- Being too aggressive that all further out ideas end up not working.
- Insufficient funding such that ambitious projects simply can't be performed due to the significant engineering effort required.

Fund projects at all scales/aggressiveness, but prioritize the further out topics and ensure they are interdisciplinary. These interdisciplinary aspects do not need to come from separate PIs, it can be a single PI that cross boundaries.