Applications:

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

Multiscale simulation applications will benefit from extreme scale computing. These include turbulent reacting flows, molecular dynamics, geophysics, climate modeling, astrophysics, material science, etc.

Algorithmic advances that facilitate more asynchrony (allowing overlap of communication with computation and eliminate/minimize global reduction operations will be important. Furthermore, algorithms that reduce memory pressure and have multiple levels of concurrency will be advantageous.

Systems:

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

Many of the applications that can benefit from extreme scale computing are characterized by the solution of large systems of partial differential equations. These have memory bandwidth requirements (associated with stencil calculations) that are not represented by traditional HPC benchmarks (BLAS/LAPACK). It will, therefore, be important to measure system performance and benchmarks against PDE solvers and not just dense linear algebra.

Technologies:

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

A major challenge will be effectively utilizing heterogeneous systems (e.g., CPU/GPU). We will need programming models that enable heterogeneous systems to be used efficiently while minimizing the burden on application programmers.

Methodologies:

How should we perform interdisciplinary research that spans applications, systems, and technologies? How should abstraction layers evolve?

Interdisciplinary teams will be required to develop the hardware, operating systems, runtime environments, software, and algorithms for forthcoming architectures. Hardware/application co-design is an example of interdisciplinary partnerships that will help ensure that applications and hardware successfully evolve together. At the application level, computer scientists must be closely partnered
with domain scientists to ensure that software abstractions are adequate and robust. None of these partnerships should be one-sided.

**Risks:**

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

Many of the challenges will be associated with architectural uncertainty. Application codes require massive efforts to refactor. Hedging against this uncertainty will be important. Immediate investment in robust abstractions that can be used by segments of communities can help mitigate this risk. Such investment should be targeted to individual communities and, wherever possible, should include several options rather than placing too much reliance on any one technology.