Introduction:
- Primary Goal: secure virtual execution environment tailored for a specific application
- Application-specific trusted computing base (TCB) size minimization in a virtualized environment is possible
- VM size and composition matters: security (TCB, attack surface) and performance (storage, memory, CPU)

Approach:
- From declarative descriptions to systems deployed as VMs.

1. Capture user and application requirements in the form of a declarative description $D$
   
   Approach:
   Gather user/application requirements as a declarative description (D). D describes an execution environment and other requirements such as trust domains and storage issues. Steps 2-4 focus on the execution environment.

   Challenges:
   Language Design, Profiling Techniques

2. Analyze D to give a system constraint described as $C$
   
   Approach:
   Consult the ‘Global Up-to-date State’ – GUS (patches, updates, versions, standards as models…) to create a more precise description of an execution environment for the application.

   Challenges:
   GUS Representation, Semantic Reasoning

3. Create a security optimized machine $M$ based on $C$
   
   Approach:
   Use constraints in C to configure an OS, creating a minimal ‘Trusted Computing Base’ in the application’s execution environment $M$ – e.g. include only relevant drivers or libraries.

   Challenges:
   Constraint Satisfaction/Optimization Approaches, Implementations

4. Deploy $M$ as a target $T$
   
   Approach:
   Pack and deploy $M$ on different target cloud/HPC platforms.

   Challenges:
   Performance Issues, Predictability, Scalability, Interoperability

Advantages
- Flexibility.
- Reusability.
- Portability.
- Ease of Use.

Work in Progress
- Model refinement and description techniques.