



# **SRM Usage Assessment And Future Path**

**Arie Shoshani  
Alex Sim**

**Lawrence Berkeley National Laboratory  
May, 2009**

# How to make our meeting productive

---



- **Usage assessment**
  - What features and functions are used regularly
  - What features and functions are sometimes used
  - What features and functions are not used at all
- **Approach**
  - Determine what is the “core” functionality that all SRMs should support
  - The “core-SRM” should support all desired functions required for simple operation of current requirements
  - Determine desirable categories of functions, consider them optional
  - Optional functions should be discoverable

# Path forward



- **Longer term**
  - Define SRM 3.0 that supports “core” and “optional” functions
  - Determine features or functions now missing (e.g. return more informative information with “SRM busy”)
  - Determine framework that is more flexible than WSDL, for introducing additional downward compatible features over time
  - Use a security model other than httpg (https?)
  - propose a time line for switching to SRM v3.0

# Determine core/optional features (1)



Functions/features	Castor	DPM	dCache	STORM	BeStMan
Data Movement functions					
prepareToGet					
prepareToPut					
BringOnline					
Copy					
Data Movement features					
file types					
multi-file requests					
directory requests					
SURL/TURL					
file lifetime negotiation					
protocol negotiation					
network transfer					
type of storage					

# Determine core/optional features (2)



Functions/features	Castor	DPM	dCache	STORM	BeStMan
Explicit Space Management					
static space with tokens					
dynamic space reservation					
dynamic space lifetime negotiation					
guaranteed/best-effort					
retention policy					
access latency					
Directory management					
Ls					
Mkdir					
Rmdir					
Mv					
Rm					
Cp					
ACLs					
Ping					