

# SSS BCM Update 8/05

Narayan Desai  
desai@mcs.anl.gov



# Overview

---

- LRS Conversion
- Cobalt
  - What is it
  - Motivations
  - Architecture
  - Portability
  - Deployment

# LRS Conversion

---

- Targetted for SC05 release
- SSSlib changeover
  - Completed and included in the snapshot
- SDK support
  - Completed and tested
- Component conversion
  - Service Directory (completed)
  - Event Manager (completed)
  - BCM Stack (completed)
  - Process Manager (completed)
- Plan from here?

# Cobalt Overview

---

- System software and tools
  - Resource Management
  - System/Configuration Management
  - Parallel Tools
- Primary Motivations
  - Scalability
  - Flexibility
  - Simplicity
  - Support for research activities

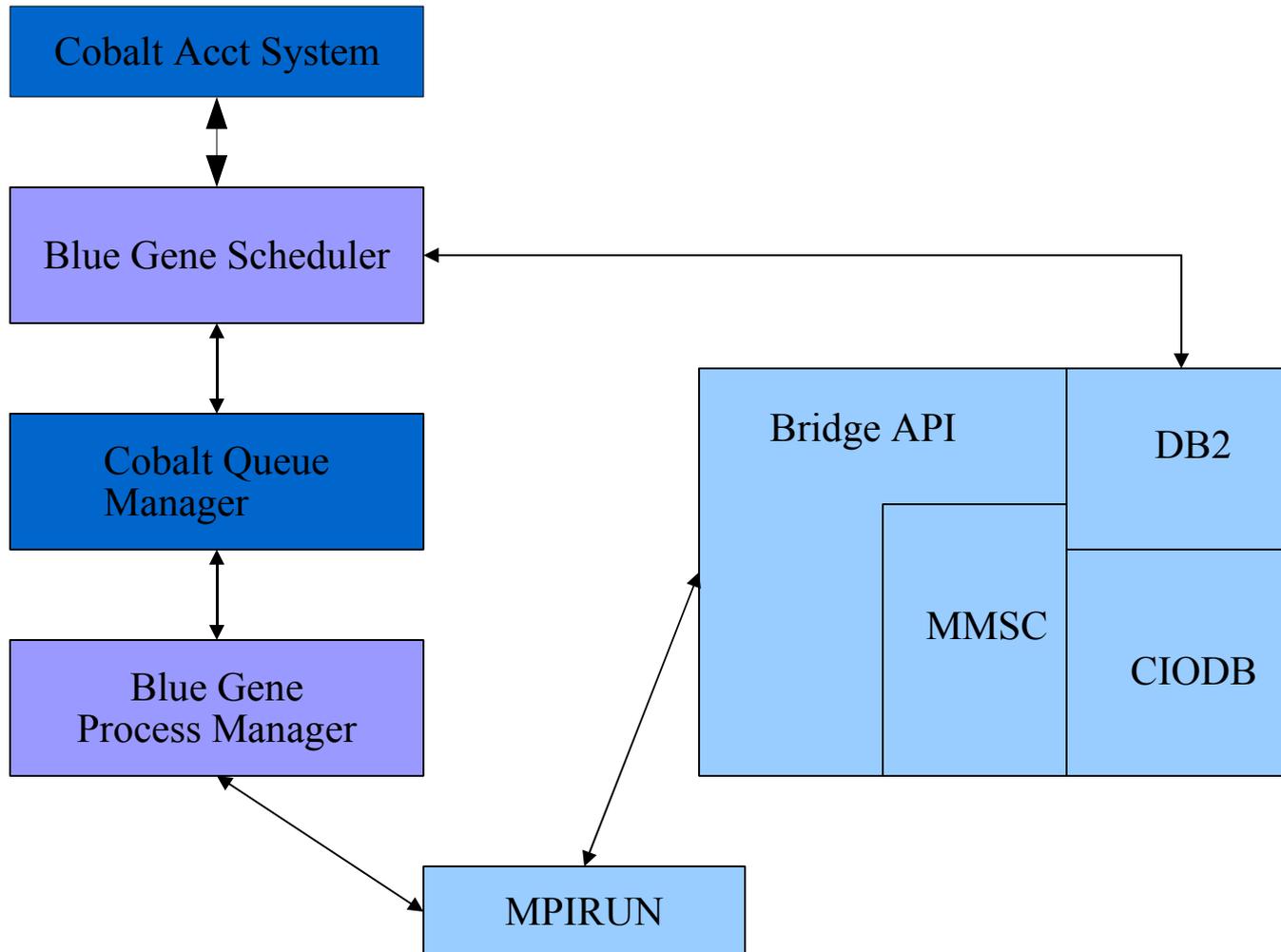
# Architecture

---

- Component architecture based on the SciDAC Scalable System Software project
  - Provides easy portability to new platforms, currently supports clusters running Linux and MacOSX, and BG/L systems
  - Highly customizable – only 5K lines of component code
  - Well-defined interfaces provide mechanism for ad-hoc usage of component data
- Parallelism provided by MPI, where needed

# Cobalt on BG/L

---



# Portability

---

- Running on
  - BG/L
  - Linux Clusters
  - OSX Clusters
  - Qemu
- Targetted for
  - Cray
    - XT3
    - X1
  - FastOS
    - ZeptoOS
    - Others?

# Unique Cobalt Features

---

- Small partition support on BG/L
- OS Specification support
- Agile
  - As we find easier ways to do things, the components can be easily modified to support new activities
  - Having readily changeable components provides a convenient location for vendor code workarounds
  - User requests can be easily fulfilled
  - More importantly, administrator requests can be easily satisfied

# Results and Experiences

---

- Running 2 BG/L systems (ANL and NCAR) and under evaluation at other Blue Gene sites
- Running several clusters at ANL
- Makes OS research on BG/L possible in a supportable manner
- Lightweight component environment makes for agile and well-suited components

# Future Developments

---

- Better scheduling policies
- Porting to new platforms
- More user front-ends (webpages, etc)
- Better docs