

70 YEARS OF CREATING TOMORROW



**Los Alamos**  
NATIONAL LABORATORY

# The State of CBTF

CScADS 2013 - Petascale Tools Workshop

July 15, 2013

J. Green, HPC-3 LANL

on behalf of the Open|Speedshop Engineering Team

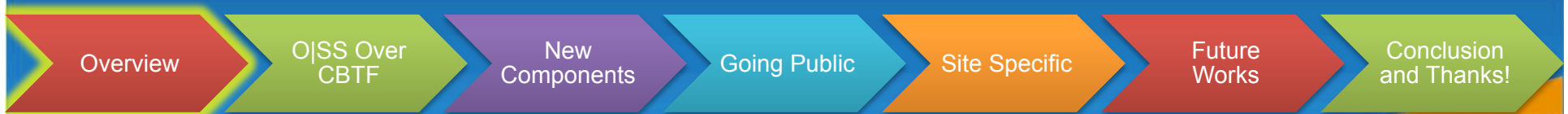
LA-UR-13-25207

UNCLASSIFIED



# Component Based Tool Framework

- Brief Overview of CBTF
- Project Status
- Discuss Open|Speedshop Implemented with CBTF Framework
- Going Public
- Site Specific Tools and Tests



UNCLASSIFIED



# Component Based Tool Framework

- Framework tailored to rapid, scalable cluster tool development with Reusable Components
- C++ / XML Code
- Dataflow Programming Model
- MRNet (Multicast Reduction Network) communication transport layer

Overview

OJSS Over  
CBTF

New  
Components

Going Public

Site Specific

Future  
Works

Conclusion  
and Thanks!

UNCLASSIFIED



# Open|Speedshop Built on Component Based Tool Framework

- Supports Same Features, Increased Scalability while Maintaining Ease of Use
- New O|SS Experiments Under Development:
  - Memory Experiment
  - Threading Experiment
  - I/O Profiling Experiment
  - GPU Experiment

Overview

O|SS Over  
CBTF

New  
Components

Going Public

Site Specific

Future  
Works

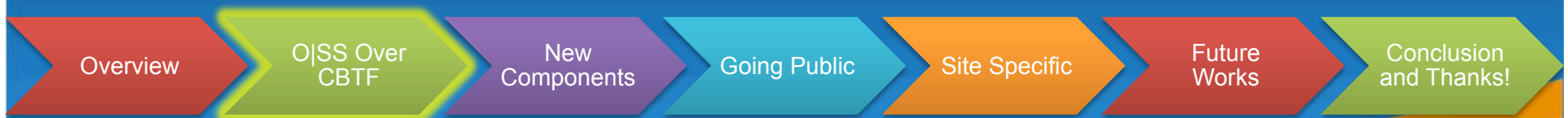
Conclusion  
and Thanks!

UNCLASSIFIED



# Open|Speedshop Built on Component Based Tool Framework

- Production Ready Open|Speedshop Using CBTF Framework Slated for Fall 2013
- “Friendly-testing” Versions Available on LANL Production Clusters
- All Current O|SS collectors work with CBTF version



UNCLASSIFIED



# CBTF Memory Analysis Collector

- Memory Analysis
  - Memory Consumption Information
  - Map Memory Allocations Back to Source Code
  - Top Ten Malloc(s) and New(s)
  - Top Ten Malloc(s) and New(s) Not Freed
  - Allocation Lifetimes and Sizes

Overview

OJSS Over  
CBTF

New  
Components

Going Public

Site Specific

Future  
Works

Conclusion  
and Thanks!

UNCLASSIFIED



# CBTF Threading Analysis Collector

- Statistics on Pthread Wait
- OpenMP (OMP) Blocking Times
- Relate Performance to Threads
- Alias to Shorten POSIX Thread IDs for Improved Readability
- Synchronization Overhead Mapping to Threads

Overview

OJSS Over  
CBTF

New  
Components

Going Public

Site Specific

Future  
Works

Conclusion  
and Thanks!

UNCLASSIFIED



# Other New CBTF O|SS Collectors

- Lightweight Tracing of I/O Functions
  - Capability to Efficiently Profile I/O Time Spent in Applications
- CUDA/GPU Collector
  - Support for Performance Analysis of Applications Built with Cuda / OpenCL for Nvidia GPUs

Overview

O|SS Over  
CBTF

New  
Components

Going Public

Site Specific

Future  
Works

Conclusion  
and Thanks!

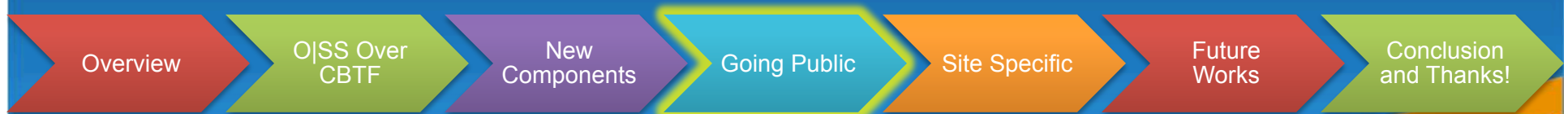
UNCLASSIFIED





# Public Repository

- CBTF Source Code to be Moved to SourceForge Publicly Accessible Repository
- Documentation and Tutorials Available at new site for Demonstrating Tool Development Techniques

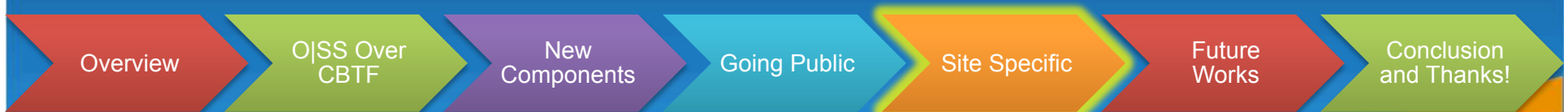


UNCLASSIFIED



# Tools Created at Los Alamos Nat'l Lab

- Tool Implementations Using CBTF
- Tools Will Be Available in /contrib Directory
- Proof of Concept that CBTF Enables Rapid Scalable Tool Development
- CBTF Tools Scale



UNCLASSIFIED



# GPU Monitoring with CBTF

- Six tools Developed
  - checkGpuMemory
  - checkConfigs
  - checkPctUsage
  - checkPstate
  - checkPstateOnly
  - checkAll
- NVIDIA Management Library
- Works with MRNet Trees of Depth 3 or More

Overview

OJSS Over  
CBTF

New  
Components

Going Public

Site Specific

Future  
Works

Conclusion  
and Thanks!

UNCLASSIFIED



# Pstool Scaling Study - Success!

- PStool performs `ps` command on all nodes
  - Reports common processes
  - Reports nodes running “rogue” processes
- 1550 pes returned in under twenty seconds
  - LANL’s Mustang
  - Correctly identified:
    - “rogue” ping process manually injected on node
    - *slurmd* and *munge* processes on head node and node targeted to run `ping`

Overview

OJSS Over  
CBTF

New  
Components

Going Public

Site Specific

Future  
Works

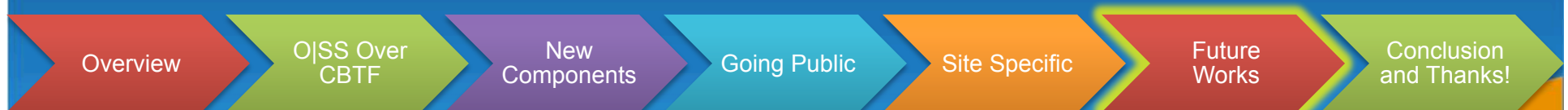
Conclusion  
and Thanks!

UNCLASSIFIED



# Future Works

- CBTF Components Support Python
- New QT4 Based Framework
  - O|SS GUI Views Under Development
- Improving Documentation for System Administrators, Tool Developers and End Users
- Goal: Production Ready O|SS/CBTF by SC'13



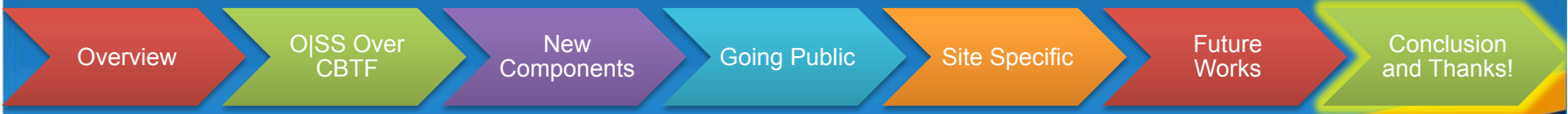
UNCLASSIFIED



# Thank you!



## To our audience, sponsors and affiliates.



UNCLASSIFIED

