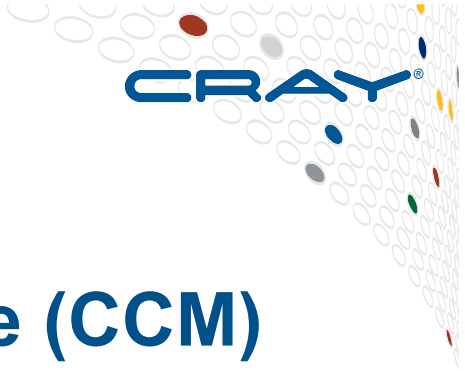
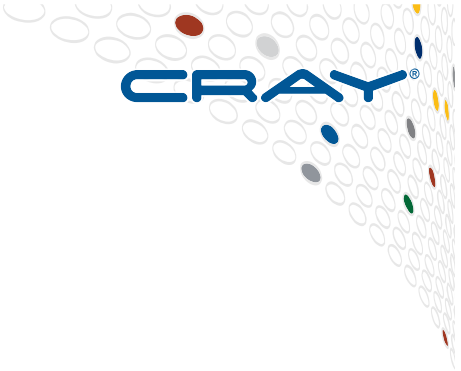


Reducing Cluster Compatibility Mode (CCM) Complexity

Marlys Kohnke
kohnke@cray.com
Presented by Andrew Barry
abarry@cray.com
Cray Inc.

4/30/15

Agenda



- What CCM does
- Services Provided by CCM
- CCM Today
- Shortcomings of the Current Design
- CCM Revised

What CCM Does

- Cray CLE software stack supports extreme scalability mode (ESM) applications
  - Must be relinked against Cray libraries
  - Some applications must/should be recompiled
- Many applications, written for whitebox clusters, cannot be relinked
- CCM allows ISV and third party MPI applications to run out of the box on Cray systems
- CCM does not impact concurrent or sequential ESM applications

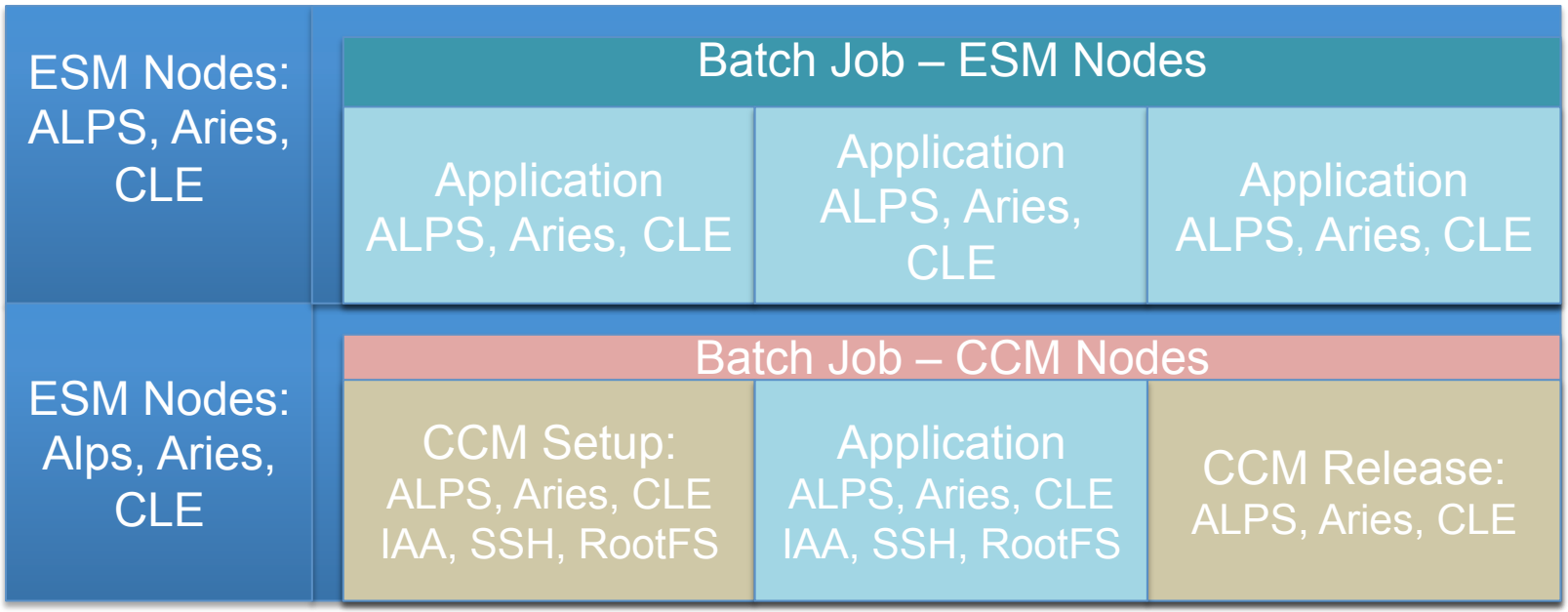
Cray CLE Design

- CLE designed for large scale
- Minimalist compute node image
- No disk on compute nodes
- Limited external network connectivity to compute nodes

CCM Supported Functionality

- **Make Aries network behave like whitebox cluster**
  - Third party MPIs can set up Aries network
  - High speed TCP and IBVerb MPI
  - `SSH/RSH` authentication to/between compute nodes
  - External network connectivity for licensing
  - User authentication
- **Make compute node behave as cluster compute node**
  - Location of some executables
  - Writeable /tmp
  - Shared libraries from /dsl filesystem

CCM Environment Create/Release



CCM Environment Create/Release

CCM Create:
Start Network Services: **rpcbind, xinetd, nscd, ypbind, dbus-daemon**
Setup Aries Network: **authentication keys, cookies**
Setup RootFS: **writeable /tmp, /dev/urandom, /etc/ssh, /var/***

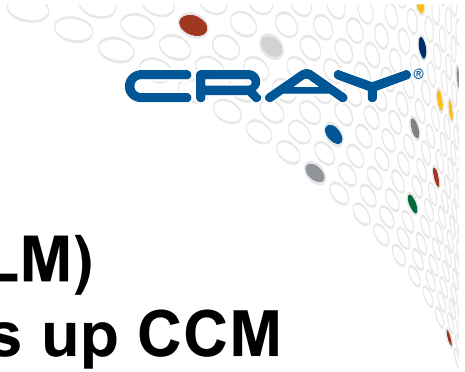
Application Runtime:
High Speed TCP with IAA, IBVerbs, application launch with SSH/RSH, Dynamic libraries from /dsl fs, writeable /tmp, normal binary location

CCM Release:
Stop Network Services, Remove RootFS bindmounts, Deconfigure Aries:
Reset the compute nodes for next ESM job

CCM Application Runtime Notes

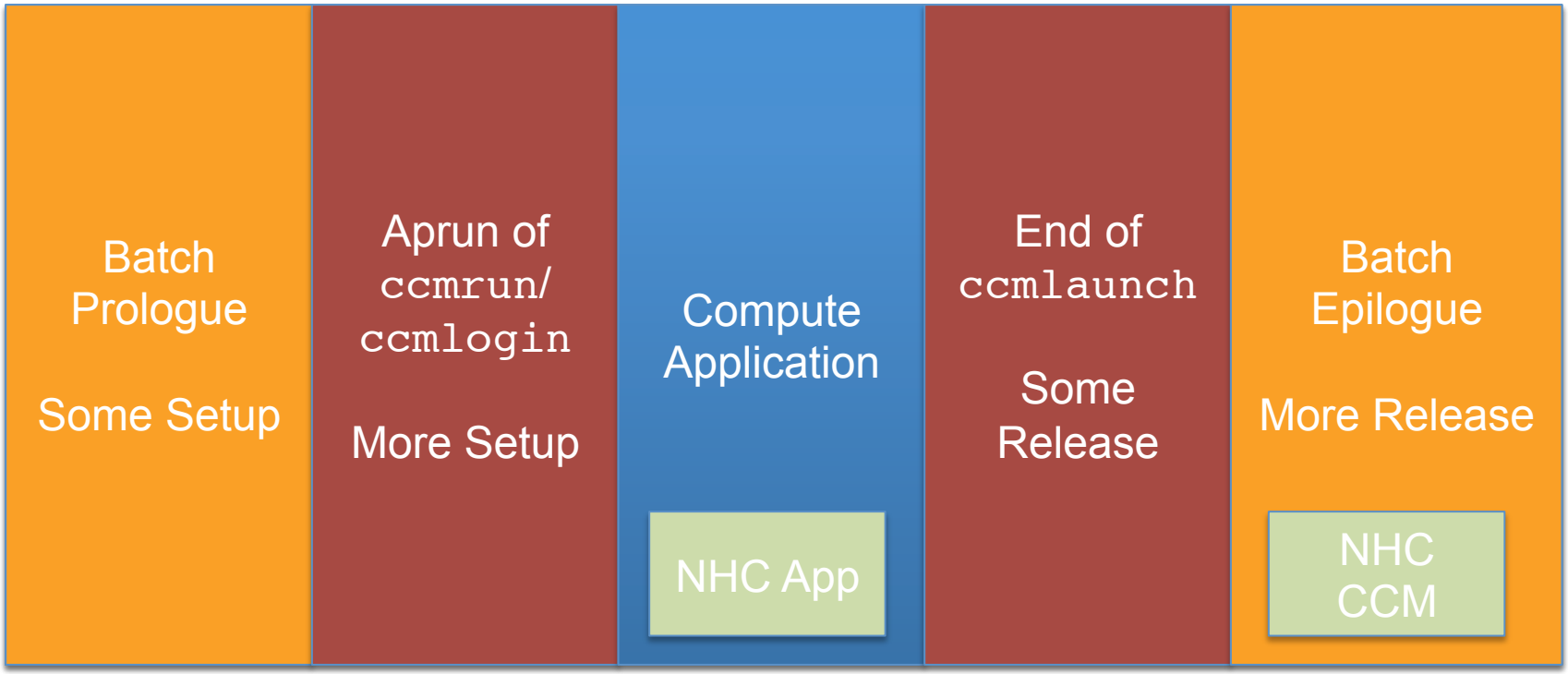
- `ccmrun` **uses Cray ALPS to run** `ccmlaunch` **on head node, which runs 3rd party MPI launcher.**
- **Application launcher (e.g.** `mpiexec`**) starts actual application**
- `ccmlogin` **call Cray ALPS to run** `ccmlaunch` **program on compute nodes, starts interactive** `SSH` **shell on compute node to act as head node.**

CCM Today



- **Tight coupling with workload manager (WLM)**
  - **WLM Prologue calls CCM prologue, sets up CCM environment**
  - **WLM Epilogue calls CCM epilogue, removes CCM environment**
- `ccmrun`/`ccmlogin` **call ALPS, which runs** `ccmlaunch` **program on compute nodes, finishing setup**

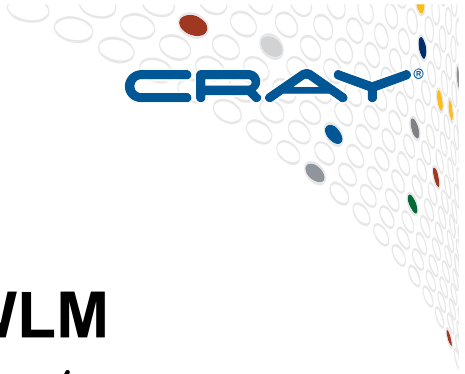
CCM Today - Workflow



Shortcomings of the Current Design

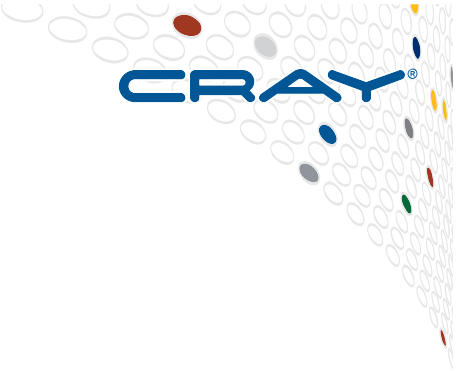
- CCM prologue uses screen scraping of stdout from ALPS `apstat` and WLM status commands
  - Required customized code for each WLM
  - Changes in command output can cause failure
- Nodelist file limited by kernel
- Timeouts can be a challenge
  - WLM Prologue/Epilogue timeouts are short
  - NodeHealthChecker (NHC) likes long timeouts
  - Epilogue calls NHC concurrent with ALPS NHC

CCM Revised



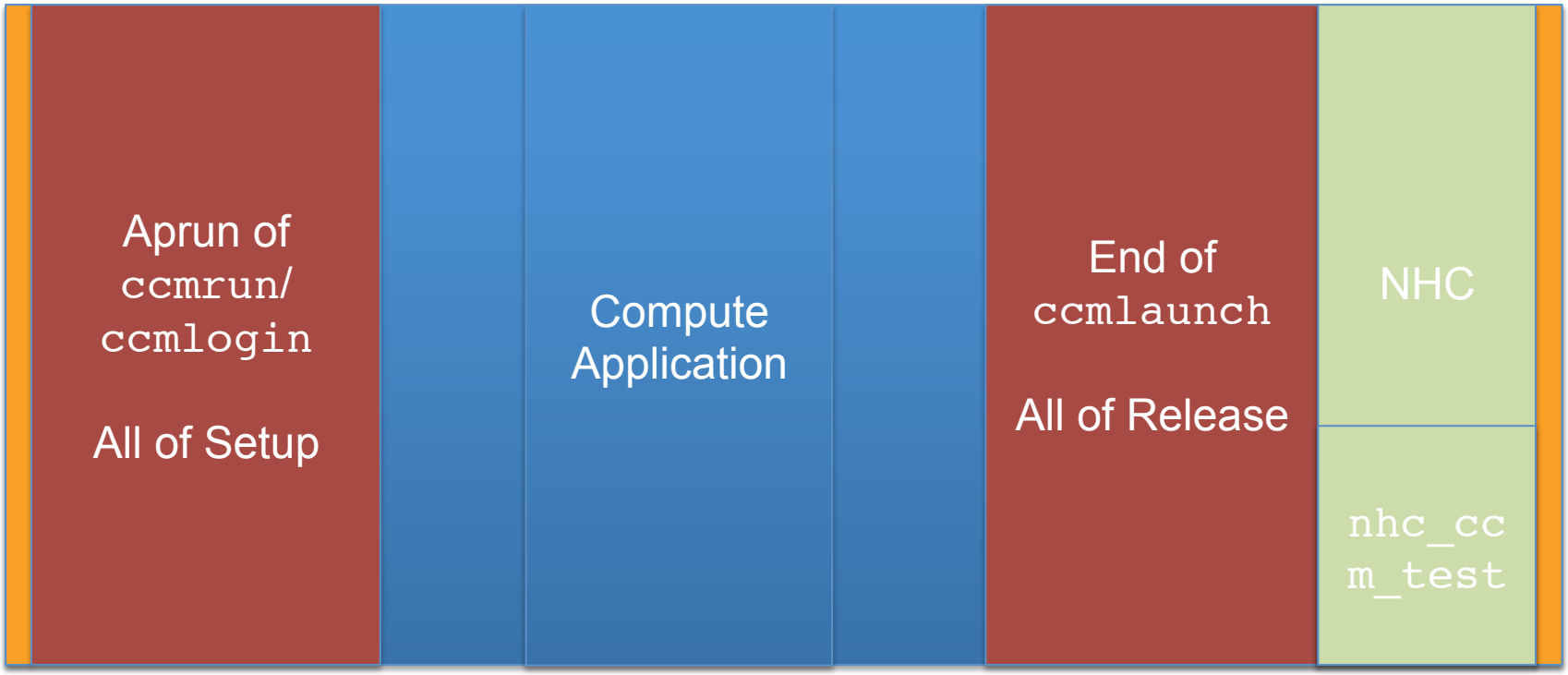
- CCM Prologue/Epilogue decoupled from WLM
  - CCM prologue invoked with first `ccmrun/ccmlogin`
  - CCM epilogue invoked when ALPS is ready to cancel the batch job reservation
- Data Path through ALPS, no screen scraping
- Nodelist file generated by ccmlaunch, on compute node, with data from ALPS

CCM Revised - NHC



- **NHC no longer run during CCM epilogue**
- **At the end of all reservations, NHC runs `nhc_ccm_test.sh` plugin**
  - **No-op for non-CCM jobs**
  - **CCM cleanup checking for CCM jobs**
- **Normal NHC timeouts**
- **`ccmrun` and `ccmlogin` interface remains the same**

CCM Today - Workflow



CCM Revised – Logging Differences

- CCM prologue output no longer written to batch job stdout/stderr
- Apsys writes CCM prologue/epilogue start and end information to apsys logfile
  - Contains batch jobid and `ccmrun/ccmlogin` apid
- CCM prologue/epilogue write additional information to ccm-YYYYMMDD file on SMW
- NHC still writes to console logfile, but now tagged with `nhc_ccm_test.sh` plugin name

CCM Revised - Configuration

- **Configurable ALPS timeout for CCM prologue and epilogue in alps.conf**
  - **prologTimeoutCCM, default 120 seconds**
  - **epilogTimeoutCCM, default 120 seconds**
- **New ccm.conf setting used by `ccmlogin` to allow more `ssh` attempts, SSH_MAX_CONNECTION_TIMEOUT**

CCM Release Schedule

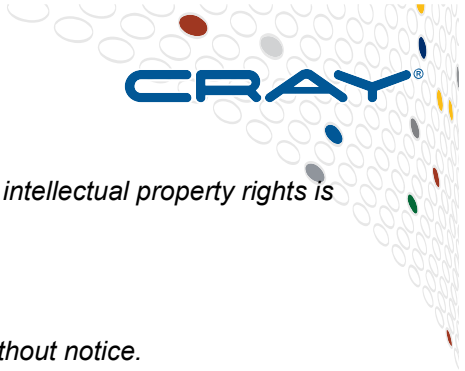
- **Both ALPS and WLM CCM prologue/epilogue services supported for CLE5.2UP04**
- **Subsequent releases contain only Revised CCM**

- **Note: CCM is not currently available for Native Slurm, but similar `SSH` functionality is planned for Rhine**

CCM Summary

- **CCM allows users to run cluster applications on Cray systems**
- **Revised CCM design removes integration with WLMs**
- **ALPS controls all of CCM setup/release, and the full path of control data**
- **NHC `nhc_ccm_test.sh` plugin for CCM cleanup**
- **More resilient, more supportable**
- **Coming soon**

Legal Disclaimer



Information in this document is provided in connection with Cray Inc. products. No license, express or implied, to any intellectual property rights is granted by this document.

Cray Inc. may make changes to specifications and product descriptions at any time, without notice.

All products, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.

Cray hardware and software products may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Cray uses codenames internally to identify products that are in development and not yet publically announced for release. Customers and other third parties are not authorized by Cray Inc. to use codenames in advertising, promotion or marketing and any use of Cray Inc. internal codenames is at the sole risk of the user.

Performance tests and ratings are measured using specific systems and/or components and reflect the approximate performance of Cray Inc. products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance.

The following are trademarks of Cray Inc. and are registered in the United States and other countries: CRAY and design, SONEXION, URIKA, and YARCDATA. The following are trademarks of Cray Inc.: ACE, APPRENTICE2, CHAPEL, CLUSTER CONNECT, CRAYPAT, CRAYPORT, ECOPHLEX, LIBSCI, NODEKARE, THREADSTORM. The following system family marks, and associated model number marks, are trademarks of Cray Inc.: CS, CX, XC, XE, XK, XMT, and XT. The registered trademark LINUX is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis. Other trademarks used in this document are the property of their respective owners.