

NMI Build & Test Laboratory:

Continuous Integration Framework for Distributed Computing Software

Andrew Pavlo, Peter Couvares, Rebekah Gietzel, Anatoly Karp,
Ian D. Alderman, and Miron Livny
University of Wisconsin-Madison

Charles Bacon
Argonne National Laboratory



THE UNIVERSITY
of
WISCONSIN
MADISON





Outline

- Motivation
- Overview
- Ongoing Research



The NMI Build & Test Framework

- Framework for building/testing software in a heterogeneous, multi-user, distributed computing environment.
- Abstracts the build/test procedures from the technology needed to execute on multiple resources.
- Apart of NSF's Middleware Initiative (NMI)



Brief History

- Condor is a distributed batch system developed at the University of Wisconsin-Madison.
- The Condor team was building and testing software by hand:
 - Every release took weeks/months to complete.
 - Developers were assigned platforms to “shepherd”.
- Oracle shamed/inspired us



Oracle's Build/Test System

- Oracle used distributed computing to automate their build/test cycle, with great success.
- Oracle selected Condor as the resource manager underneath their build and test system for their flagship database server product:
 - Automatic nightly builds.
 - Extensive regression testing.
- If Oracle can do it, why can't we?



NMI Laboratory Goals

- Design, develop, and deploy a complete system capable of performing build and tests of disparate software packages on a heterogeneous collection of platforms.
- Key tenets:
 - Dependability
 - Traceability
 - Manageability
 - Portability
 - Extensibility



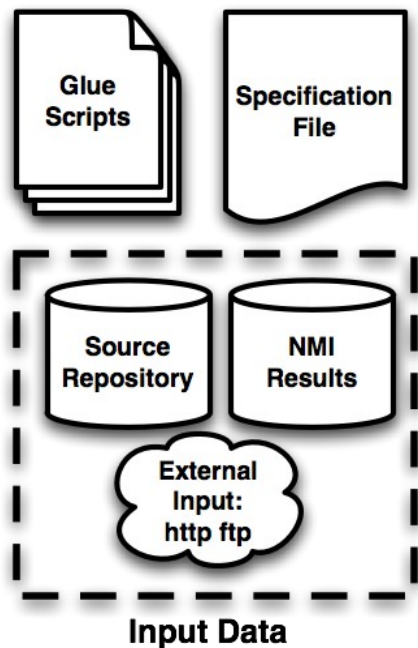
Framework Design Principles

- Tool independent
- Lightweight
- Well-controlled environments
- Central results repository
- Fault tolerant
- Explicit task separation



Using the Framework

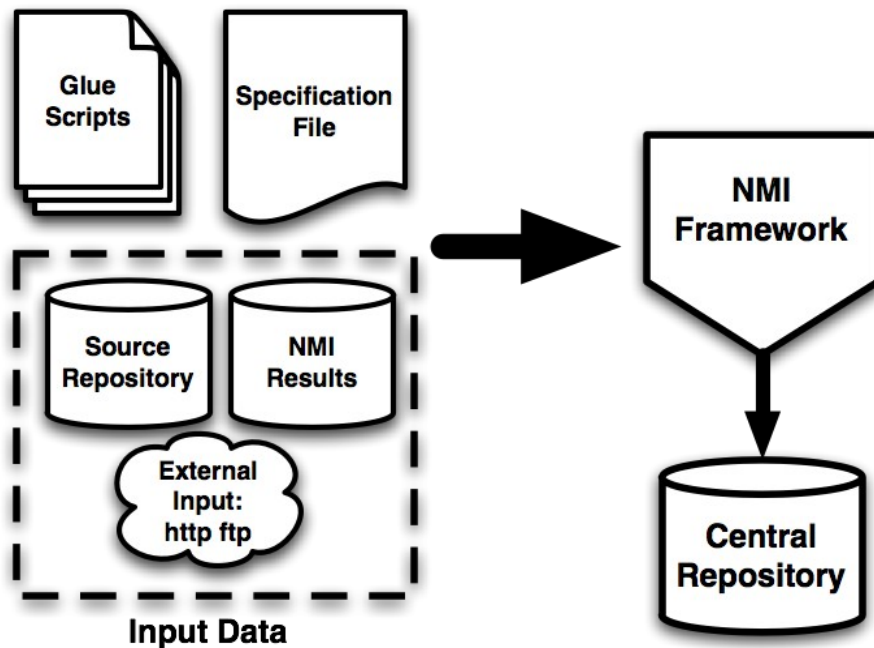
- Users define workflow of build/test procedures, software dependencies, and target platforms.





Using the Framework

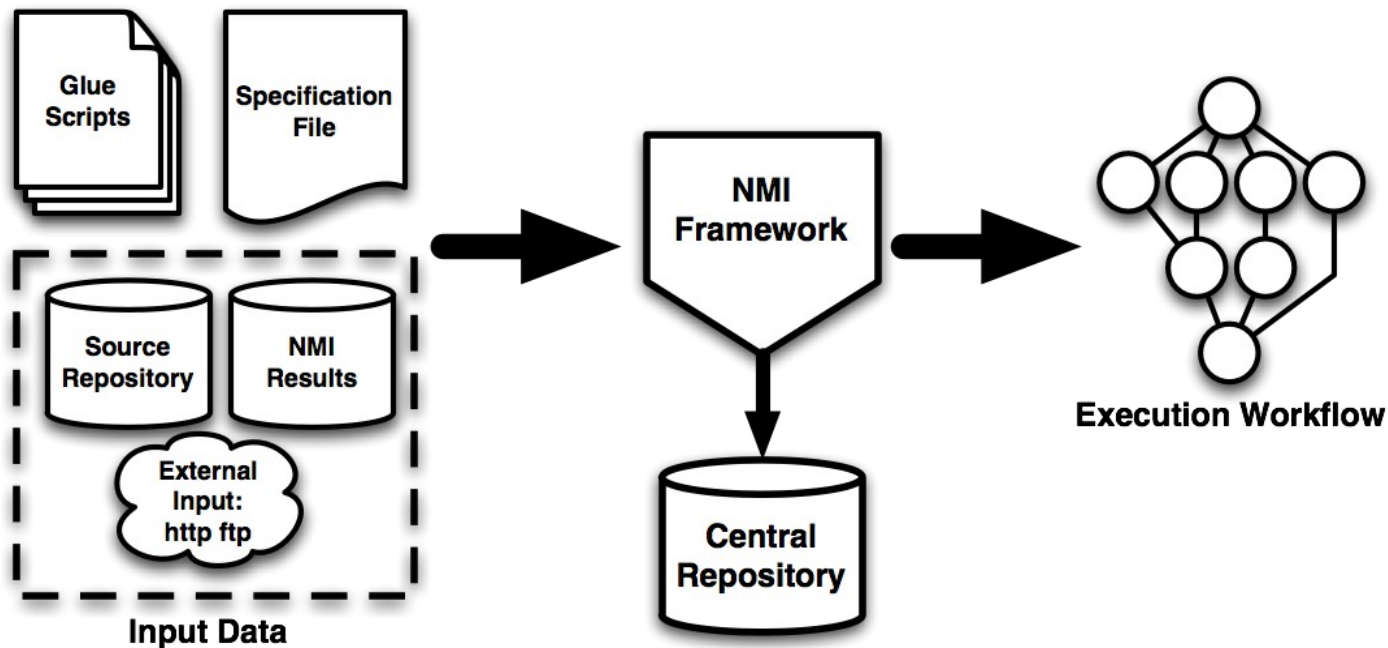
- This information is submitted to the framework and stored in the central repository.





Using the Framework

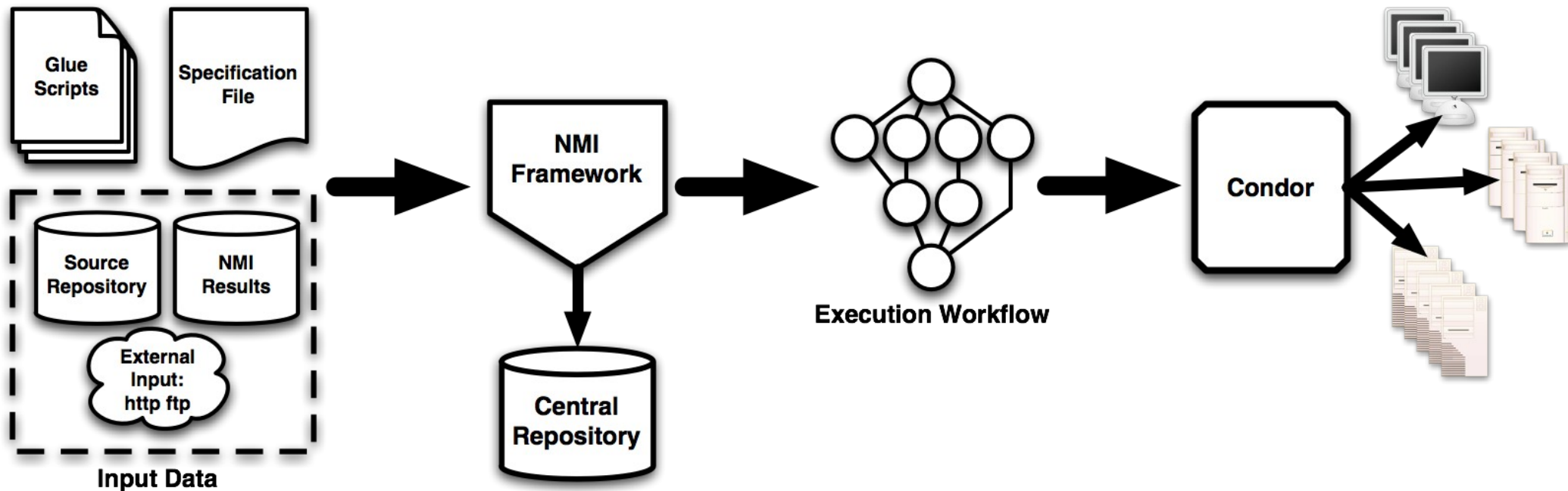
- The build/test procedures are then translated into an execution workflow.





Using the Framework

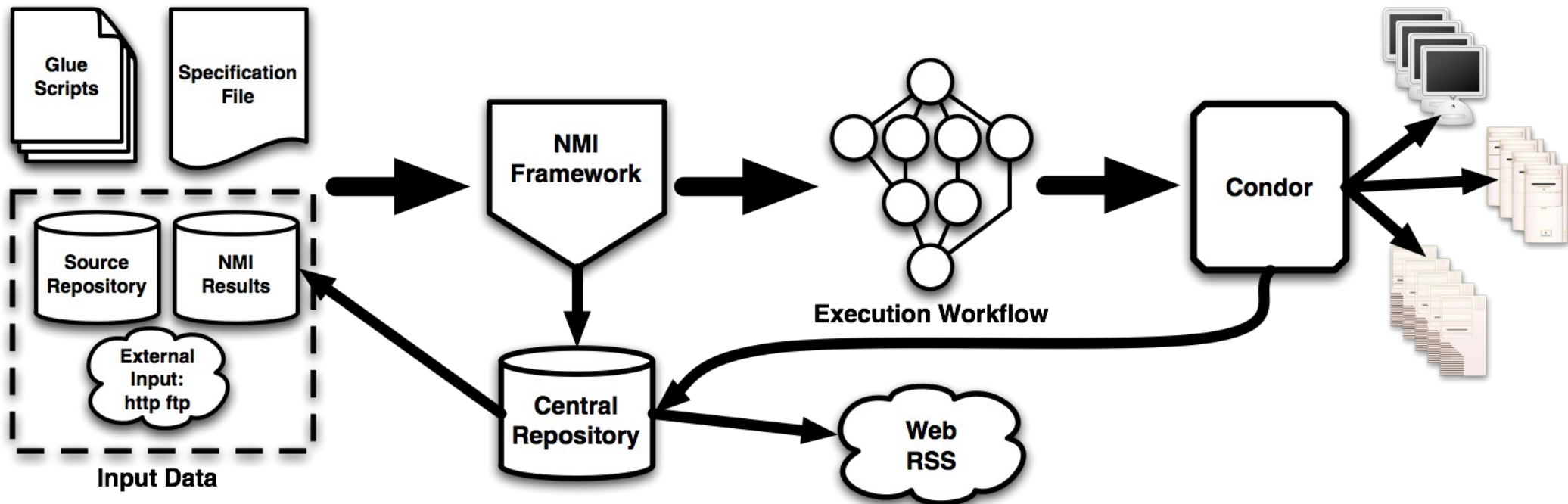
- This workflow is then submitted to Condor, which then submits the various jobs along with special framework software to one or more computing resources.





Using the Framework

- When the build/test is complete, the results are sent back to the central repository and made available from a user interface or for future builds and tests.





Building Software

- Automation
 - Always more to it than just “configure” & “make”
- Reproducibility
 - Well-managed & comprehensive source repository
- Portability
 - No dependencies on “local” capabilities
-



Testing Software

- Use any testing harness/suite.
- Cross-site Testing
 - Test services across administrative boundaries.
- Cross-platform binary compatibility testing
 - Example: “Deploy Linux binaries on FreeBSD”
- Backlog Testing
 - Run new tests on old binaries.



Current Research

- Parallel Testing
 -
- Automatic Cross-site Job Migration
 - Route submissions to resources outside of local administrative domains.



Current Research

- Virtual Machine Support
 - Local cache of pre-configured VM images.
 - Submission is automatically injected into VM image.
 - Automatic discovery of image configuration.
- Integration with other software quality projects
 - ETICS Project at CERN
 - OMII-UK/Japan
 - ???



Acknowledgments

- This research is supported in part by NSF Grants
 - No. ANI-0330634
 - No. ANI-0330685
 - No. ANI-0330670



Availability

- The NMI Build & Test Laboratory continuous integration framework is available for download at our website under a BSD-like license:

`http://nmi.cs.wisc.edu`