An Architecture for Distributing the Computation of Software Clustering Algorithms

2001 Working Conference on Software Architecture (WICSA'01).

Brian S. Mitchell, Martin Traverso & Spiros Mancoridis

Math & Computer Science, Drexel University

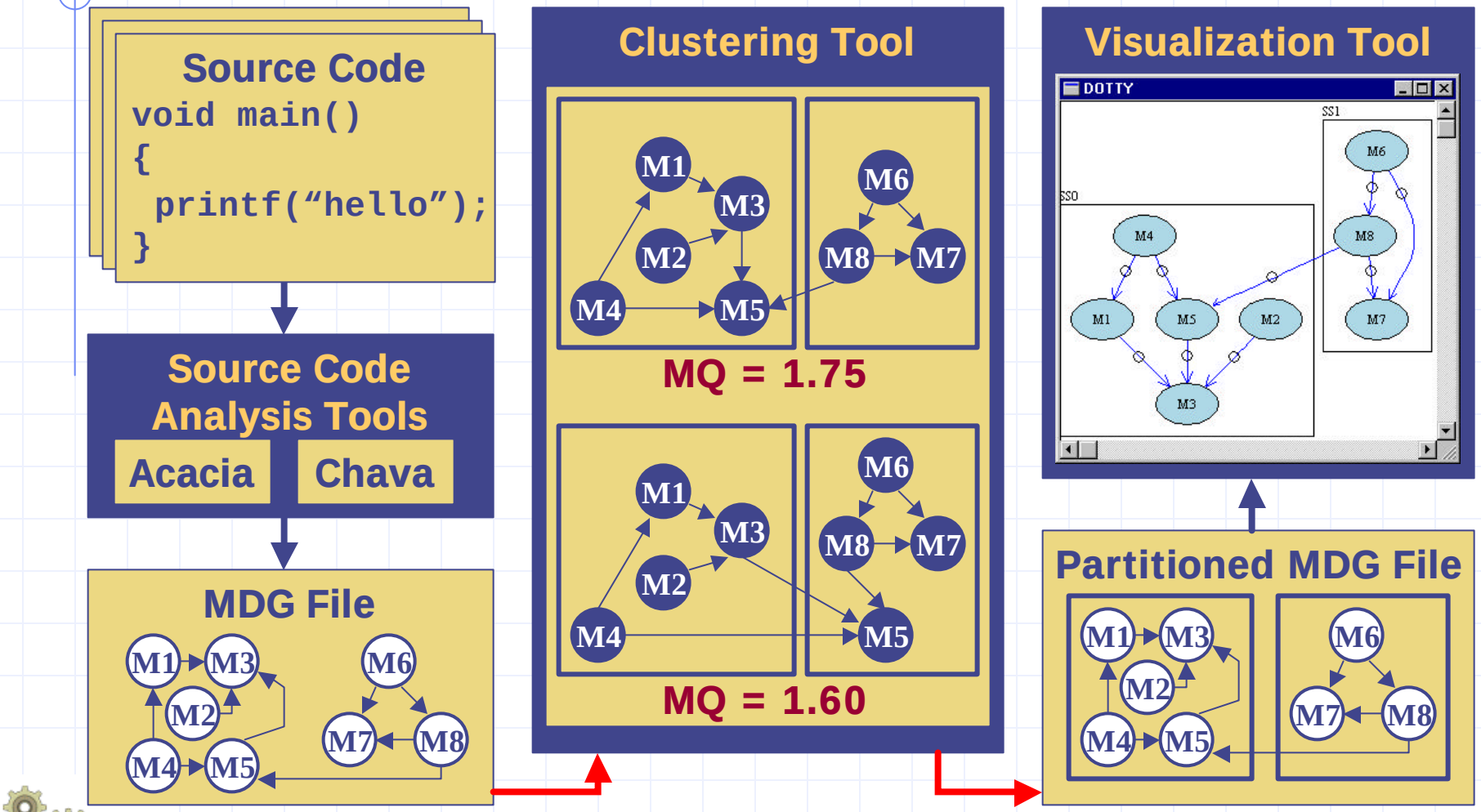
AT&T Labs-Research

Software Architecture

- Software Architecture describes the:
  - System elements
  - Interaction between the system elements
  - Patterns that guide the composition of the elements
  - Constraints on the patterns

[Shaw & Garlan 1996]

Reverse Engineering Environment



Software Architecture Challenges

- Determining the software architecture
  - Designer knowledge, and/or
  - Up to date documentation, and/or
  - Automated tooling
    -  Source Code Analysis
    -  Clustering
    -  Visualization
    -  Evaluation
    -  Design Constraint Validation

Bunch Clustering Tool Evolution

	Semi-Automatic	Automatic	User Tooling
Bunch V 1.x 1998		✓	
Bunch V 2.x 1999-2000	✓	✓	✓
Bunch V 3.x 2000-2001	✓	✓	✓

Distributed Clustering Added in Bunch Version 2.x

Bunch is the clustering tool produced by the Drexel University Software Engineering Research Group.

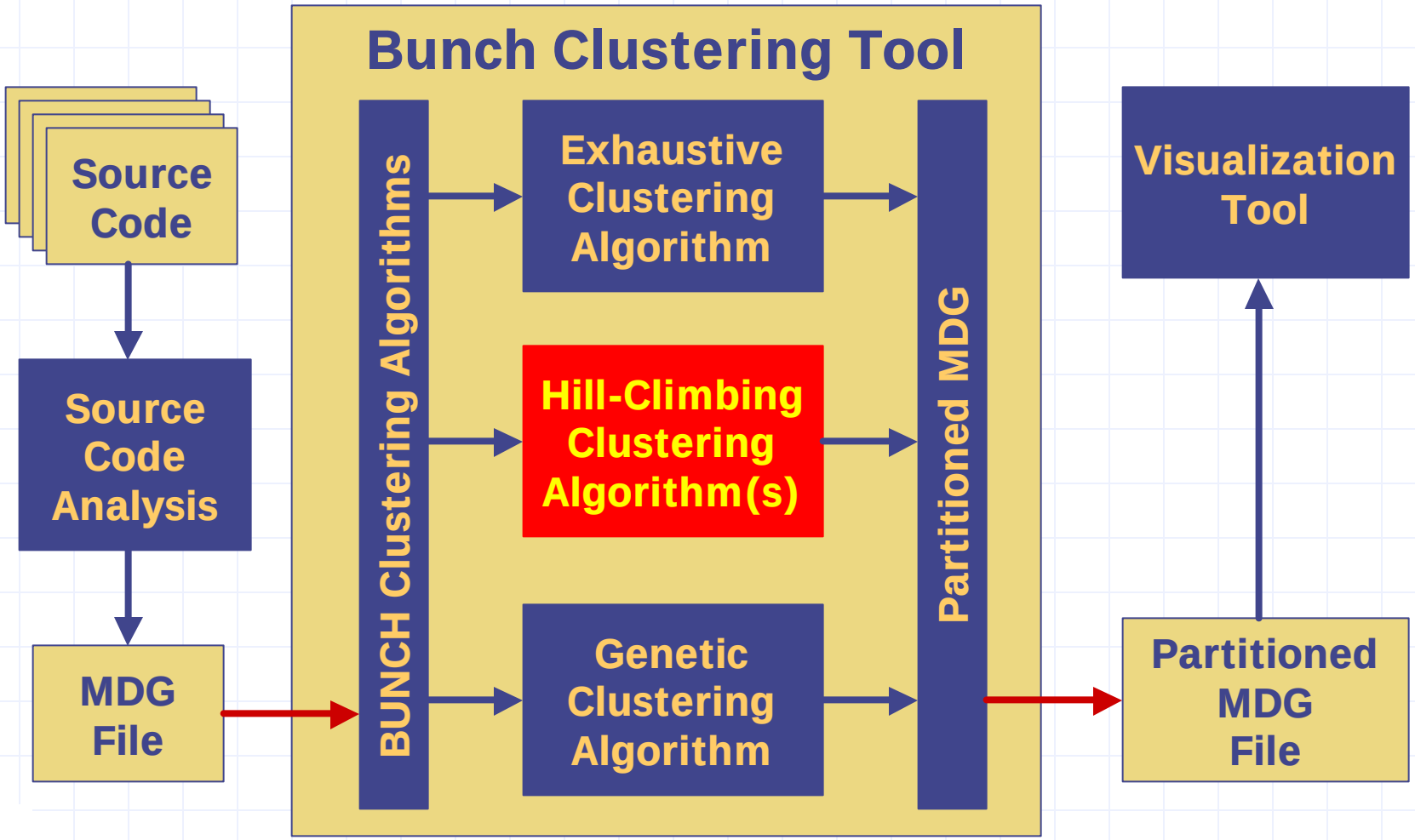
Clustering Tool Requirements

- Pluggable Algorithms
- User Knowledge Integration
- Programming Language Independence
- Tool Integration
  - Source Code Analysis
  - Visualization
  - Evaluation
  - API
- PERFORMANCE to handle large and complex systems

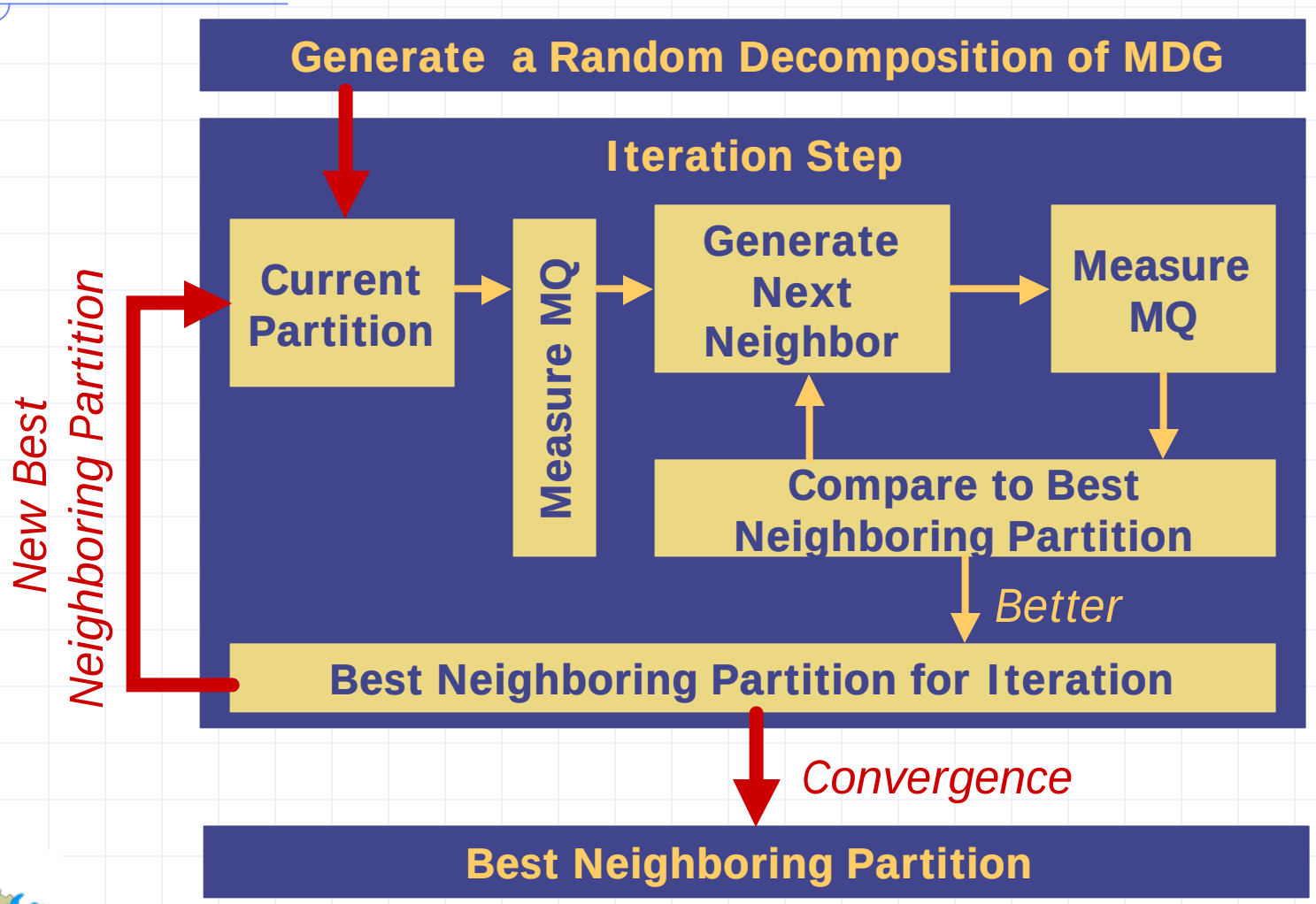
Bunch Challenges

- Performance well-suited to small and intermediate sized systems (< 250 modules)
- Design/Architecture changes were required to improve performance
  - Clustering Algorithm and Implementation Enhancements
  - Distributed Processing Capabilities

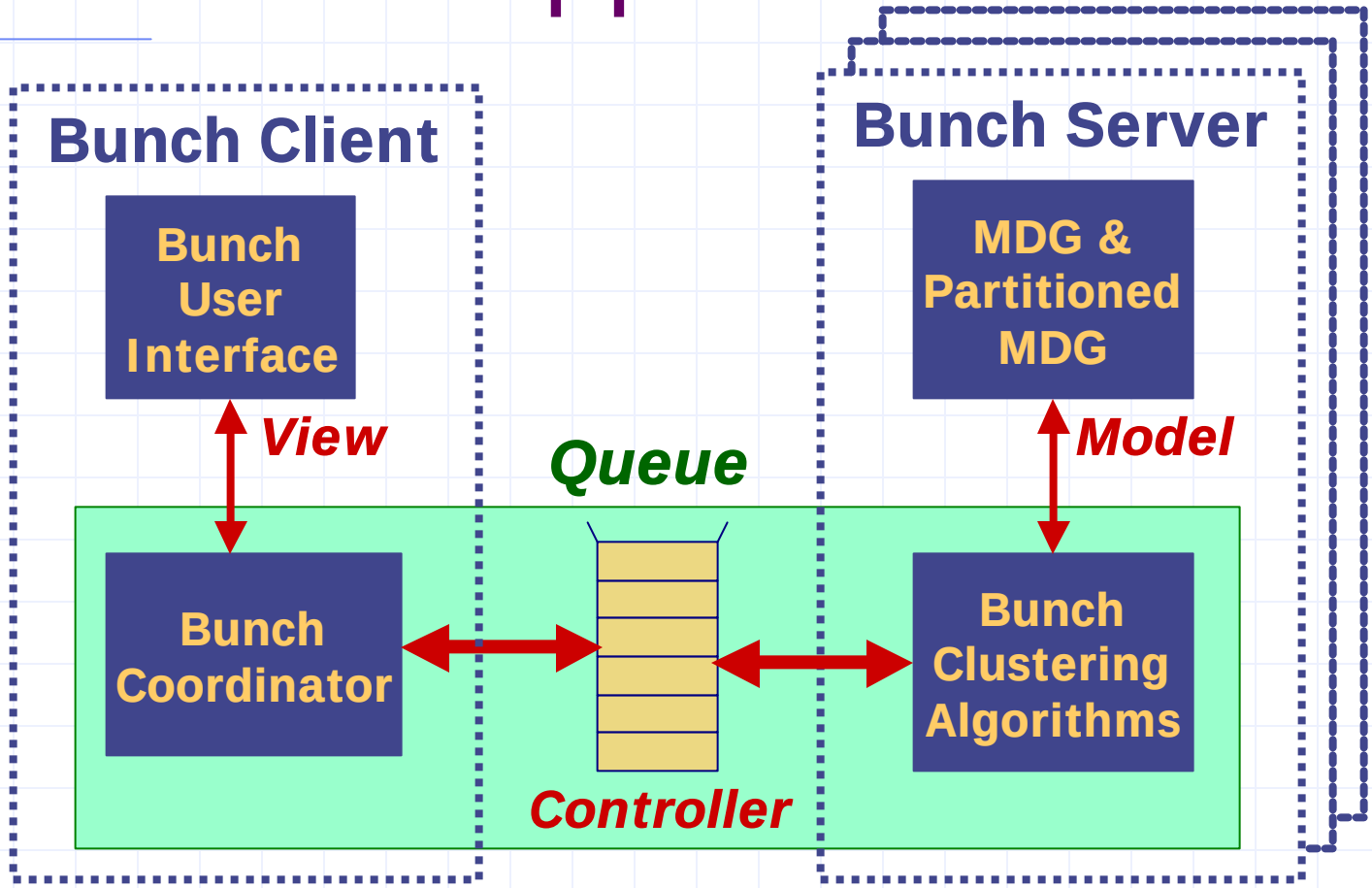
Bunch Environment



Bunch Hill Climbing Clustering Algorithm

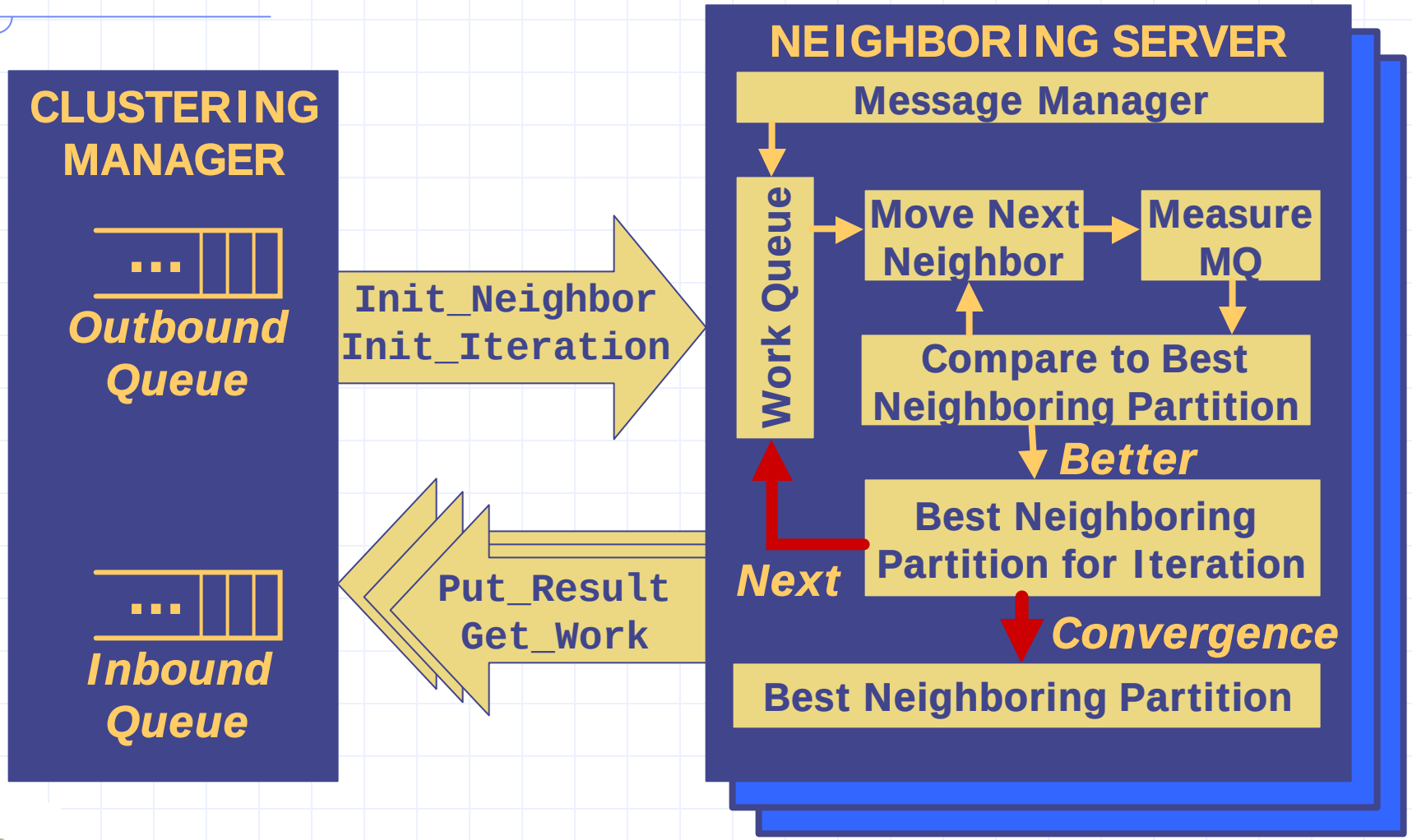


Bunch's MVC Architecture and Algorithms Support Distribution

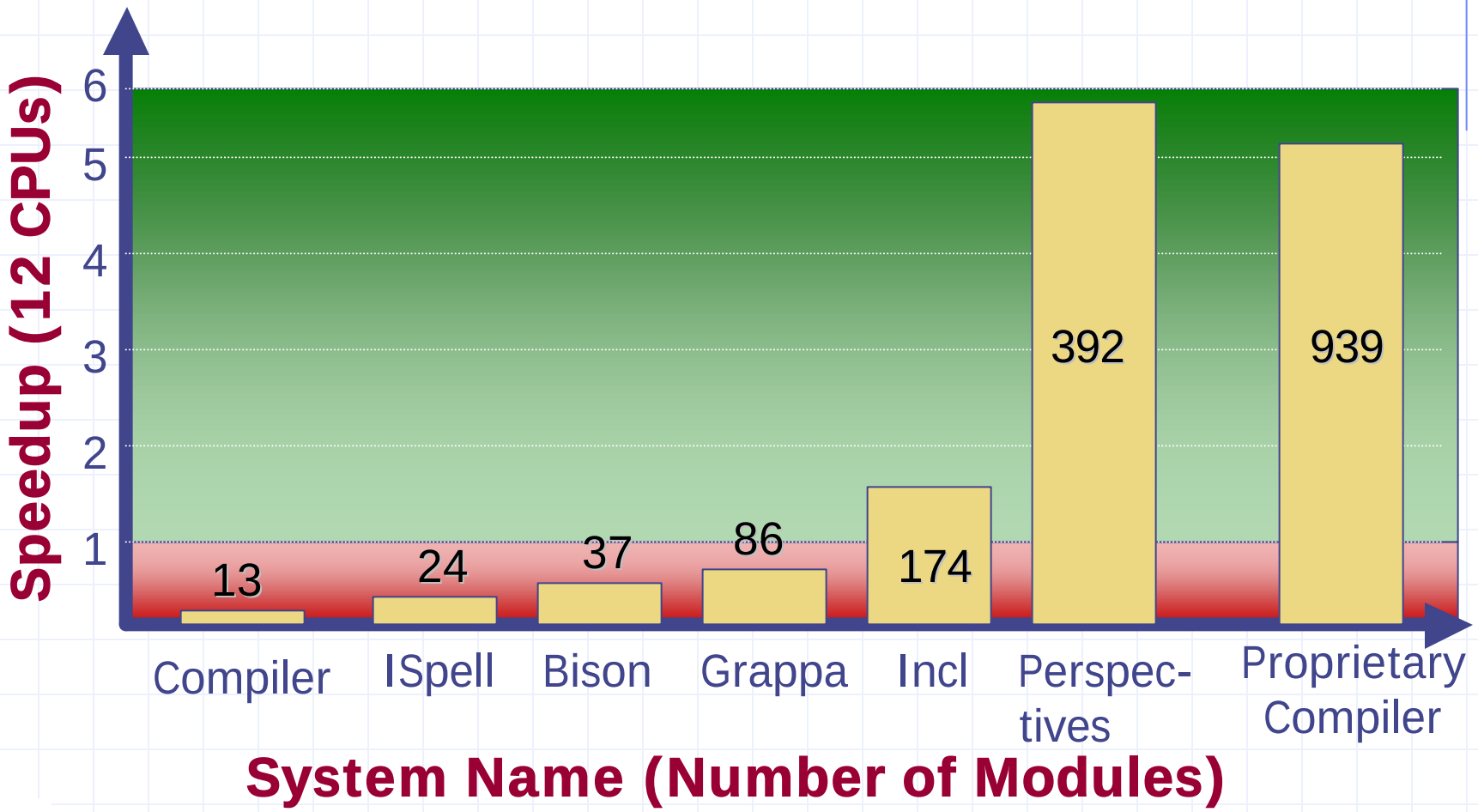


- *Clustering Activity Messages: Producer/Consumer Pattern*
- *Status & Management Messages: Publish/Subscribe Pattern*

Bunch Distributed Hill Climbing Clustering Algorithm



Case Study Results



Concluding Remarks

- Distribution approach based on:
  - Optimization of clustering approach
  - Bunch's MVC Architecture
- Performance improved for large systems, further improvement still possible
- Future improvement based on additional implementation optimizations
- Bunch written in 100% Java, DBunch uses RMI/IIOP Infrastructure

Visit Bunch Online: http://serg.mcs.drexel.edu/bunch