

Distributed BLAST with ProActive

Santosh Anand
Richard Christen*
Claude Pasquier*

***UMR 6543 CNRS & University of Nice
Virtual Biology Lab, Campus Valrose**





Plan



- ◆ Sequence Similarity Search Problem and **BLAST**: Overview and Issues
- ◆ Parallel Distributed BLAST: Various Approaches

GeB: Grid-enabled BLAST

- ◆ Grid-enabled BLAST Architecture
 - ◆ GeB Implementation
 - ◆ Merging of partial results
 - ◆ Benchmark results
-
- ◆ Conclusions and Future roadmap

Sequence Similarity Search Problem

>Q9GJY8 Q9GJY8 GAMMA2-GLOBIN.

MSNFTAEDKAAITSLWGKVNVEDAGGETLGRLLVVYPWTQRFFDSFGSLCSPSAIMGNPKVKAHG.VKVLT
SLGEAIKNLDDLKGTFGQLSELHCDKLHVDPEDFRLGNVLVTVLAILHGKEFTPEVQASRQKMOVAGSAL
ASRYH

A representation of a sequence of the protein called globin (Query-Sequence)

>Q9XT16 Q9XT16 EPSILON GLOBIN (FRAGMENT).

MVHFTAEEKAAITNTWVGKVNVEEAGGEALGRLLVVYPWNQKFFDNFGNLSSSSAIMGNPQ
VKAHGKKVLT.SFGDAVRNMDNLKAAFAKLSELHCDKLYVDPENFRL

>Q9TUY5 Q9TUY5 EPSILON GLOBIN (FRAGMENT).

MVHFTAEEKAAITNEWVGKVNVEEAGGEALGRLLVVYPWNQKFFDNFGNLSSSSAIMGNPQ
VKAHGKKVLT.SFGDAVK.NMDNLKAAFAKLSELHCEKLHVDPENFRL

>Q9XT20 Q9XT20 EPSILON GLOBIN (FRAGMENT).

MVHFTAEEKAAITNKWGKVNVEEAGGEALGRLLVVYPWNQKFFDNFGNLSSSSAIMGNPQ
VKAHGKKVLT.PFGDAVK.NMDNLKAAFAKLSELHCDKLHVDPENFRL

>Q9R1N1 Q9R1N1 BETA GLOBIN (FRAGMENT).

LLGNMIVIVLGHMHMGKDFT.PAAQ.EAFQKVVGGVATALADKYH

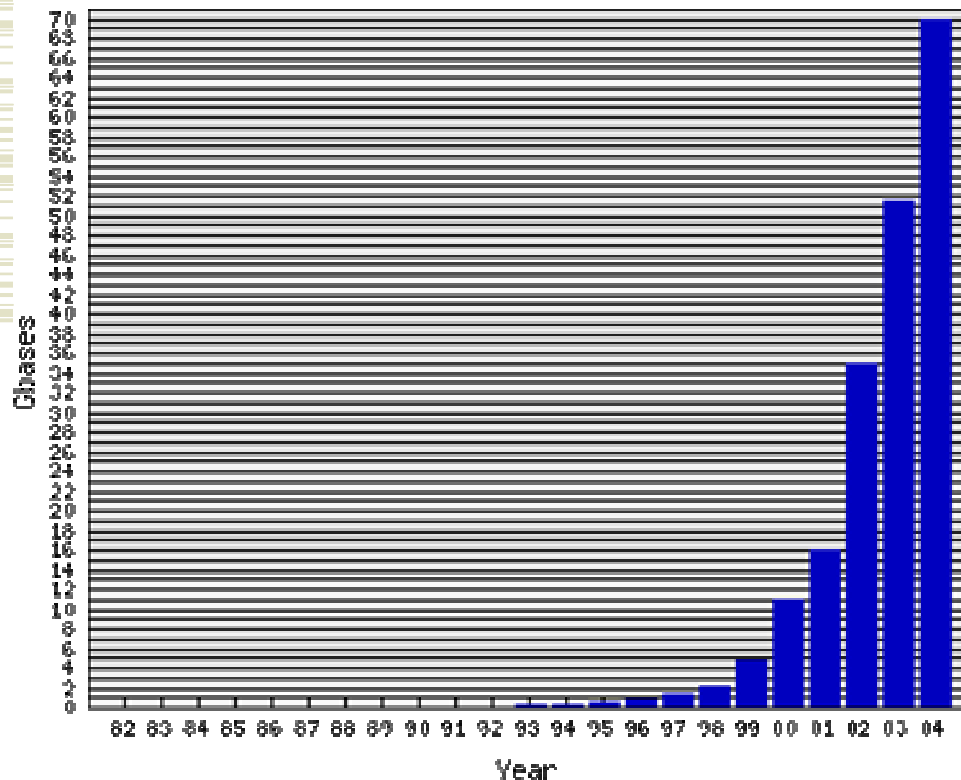
A small representative part of globin-protein database (Database-Sequence)

Question: Are there sequences in the Database-sequence which are similar (identical) to globin-protein of Query-sequence?

Sequence Similarity Search Problem is *embarrassingly* parallel!

NCBI BLAST and sequence comparison: Issues (1/2)

NCBI (**N**ational **C**entre for **B**iotechnology **I**nformation) **BLAST** is one of the most popular software used for rapid biological sequence-similarity search.



- ◆ Sequence DB are growing exponentially (roughly doubling every year)

- ◆ Hardware growth usually follows Moore's Law

Fig: Year-wise growth of nucleotide database at EMBL

NCBI BLAST and sequence comparison: Issues (2/2)

- ◆ quite compute-intensive
- frequently one may wish to look for more than one query sequences
- the database of sequences can be (very-very) big!

Important Issue: If not enough physical memory to hold the entire database

→ paging

→ significantly downgrades BLAST performance

**So, we propose to develop a parallel, distributed
Grid-enabled version of NCBI BLAST (GeB)**

Parallel BLAST: Various Approaches

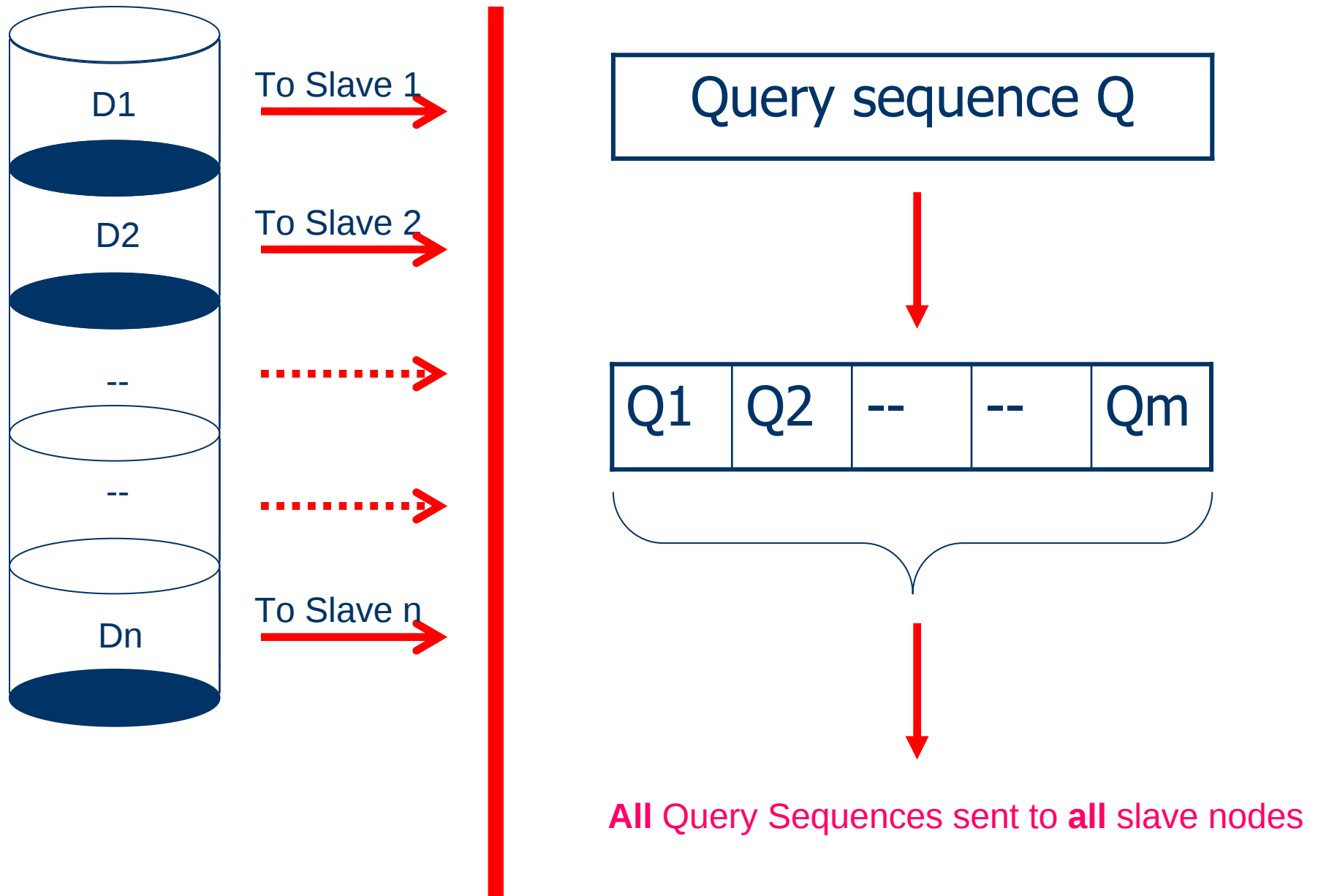
- **Hardware Parallelization:** Requires custom hardware
- **Database Segmentation:** Split the database in roughly equal parts as there are number of computing nodes.
Advantage: can eliminate the high overhead of disk I/O
can => **super-liner speedups**
- **Query Segmentation:** Split the query-sequence file
can => **liner-speedups**
- **A Hybrid Approach:** very good load-balancing!
can => **super-linear speedups**

GeB: Parallelism Strategy

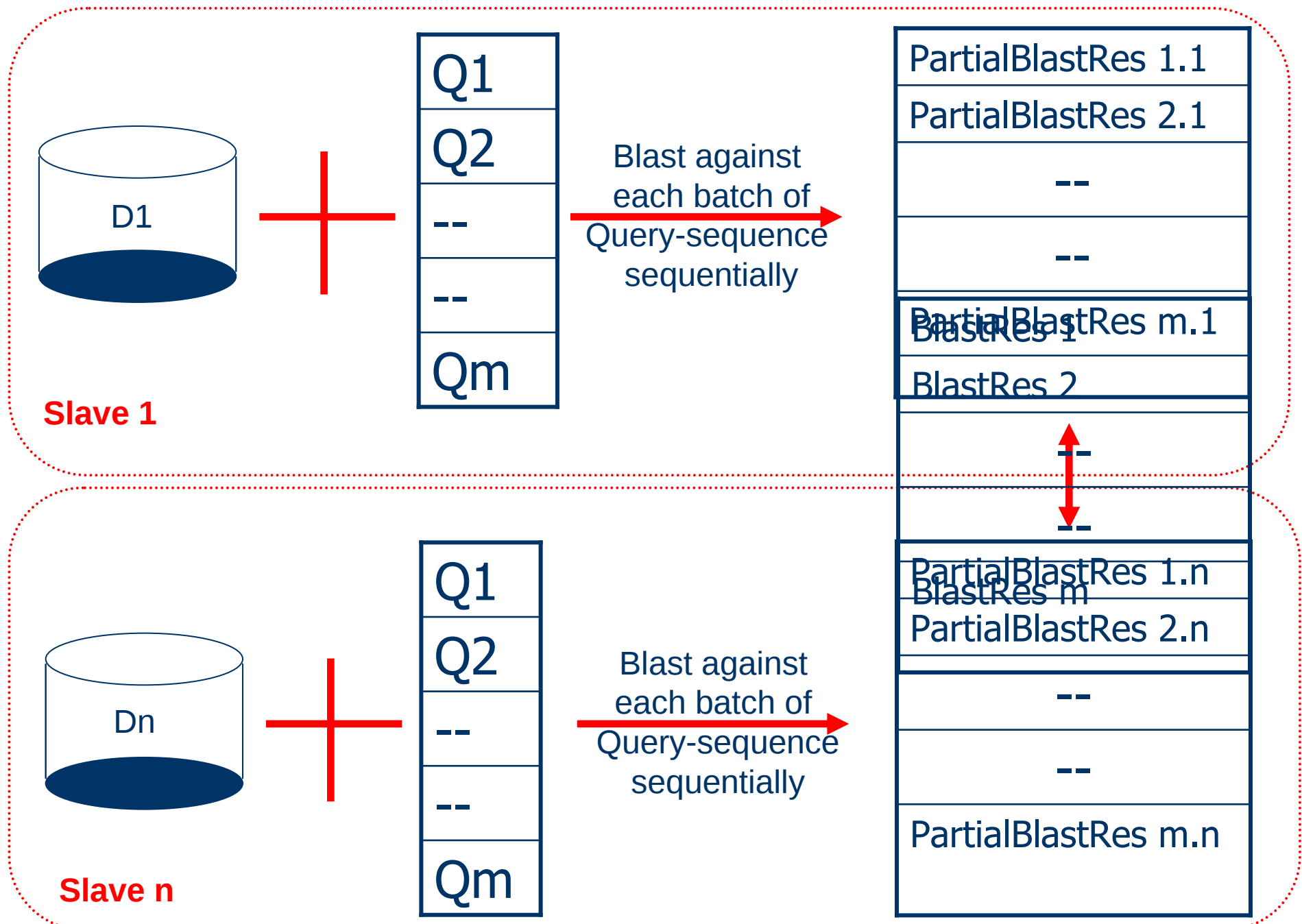
	Query	Database	Parallelism Strategy
<i>Fine-grained</i>	1 sequence	1 sequence	Single query on single database sequence
<i>Medium-grained</i>	1 sequence	m sequences	Single query on partioned database
<i>Coarse-grained</i>	n sequences	m sequences	batch of queries on partioned database

- ♦ **Finest grained:** Not very much suitable due to the high overhead of launching BLAST program each time.
- ♦ **Medium or Coarse grained?**
In GeB, the design is kept flexible so that the user can determine how much fineness (s)he requires

GeB: Architecture and Scenario (1/2)



GeB: Architecture and Scenario (2/2)



GeB Implementation

ProActive - The platform for GeB

- ◆ **Slaves Nodes - Virtual Nodes:** defined through an XML–Deployment Descriptor file.
- ◆ **ProActive Group:** A group of slave-nodes where actual BLASTing is done.

Additional Open Source Libraries Used

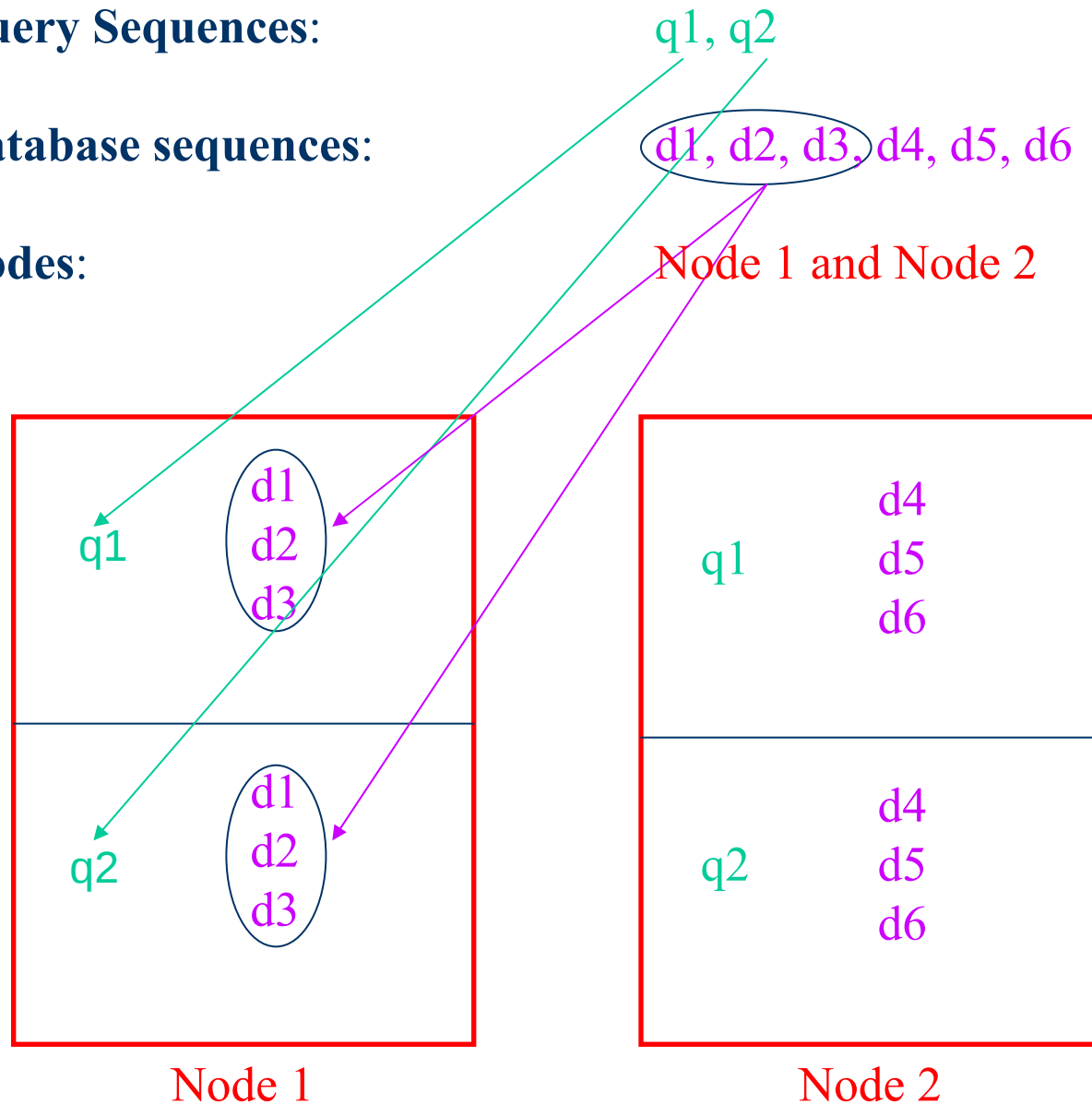
- ◆ **DBSR JBlast/JLaunch Package:** For launching the NCBI BLAST program on each nodes.
- ◆ **BioJava BLAST Parser:** For parsing the BLAST output got from each node so as to merge the partial results easily to get the final result

GeB: Building of Result (1/3)

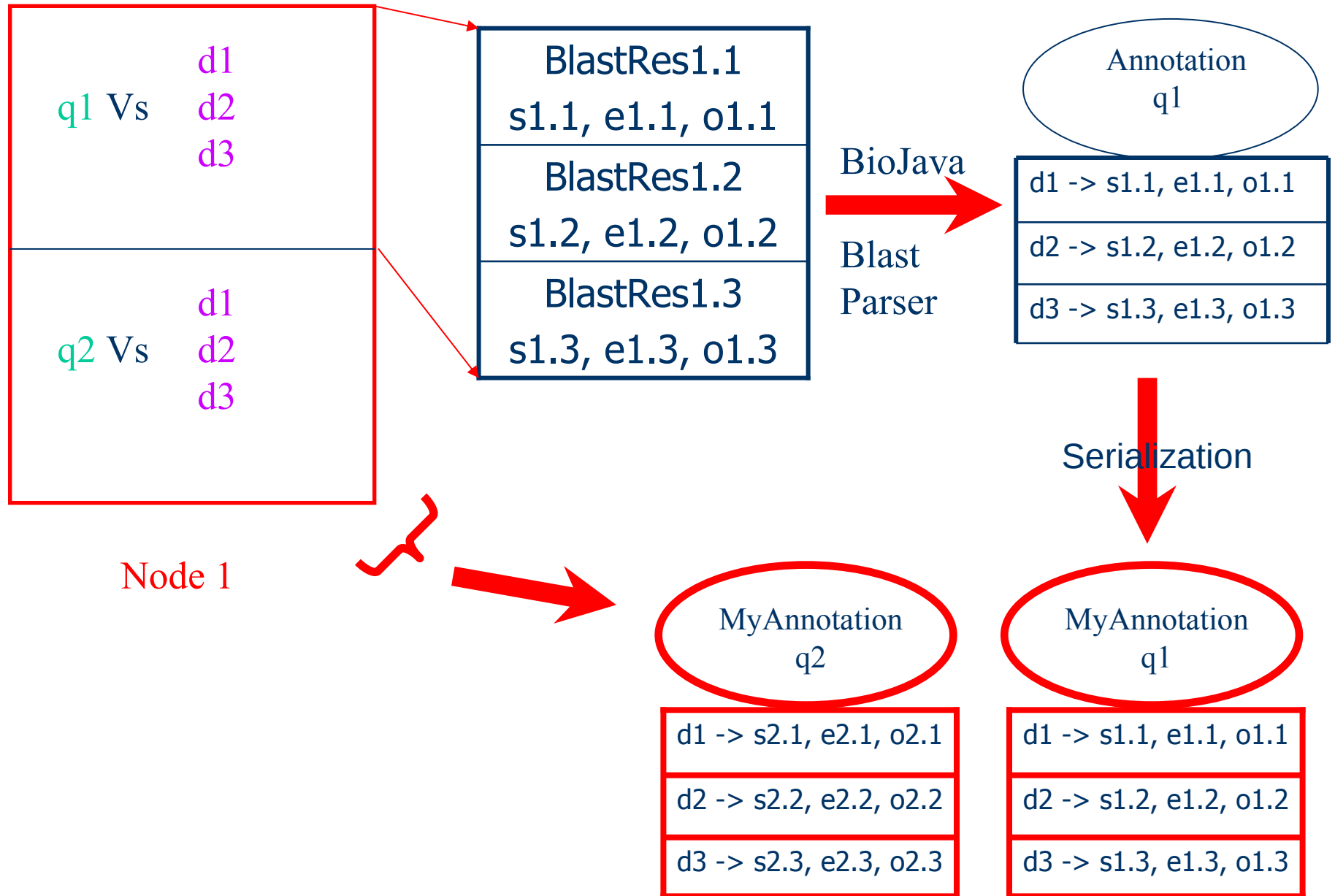
Query Sequences:

Database sequences:

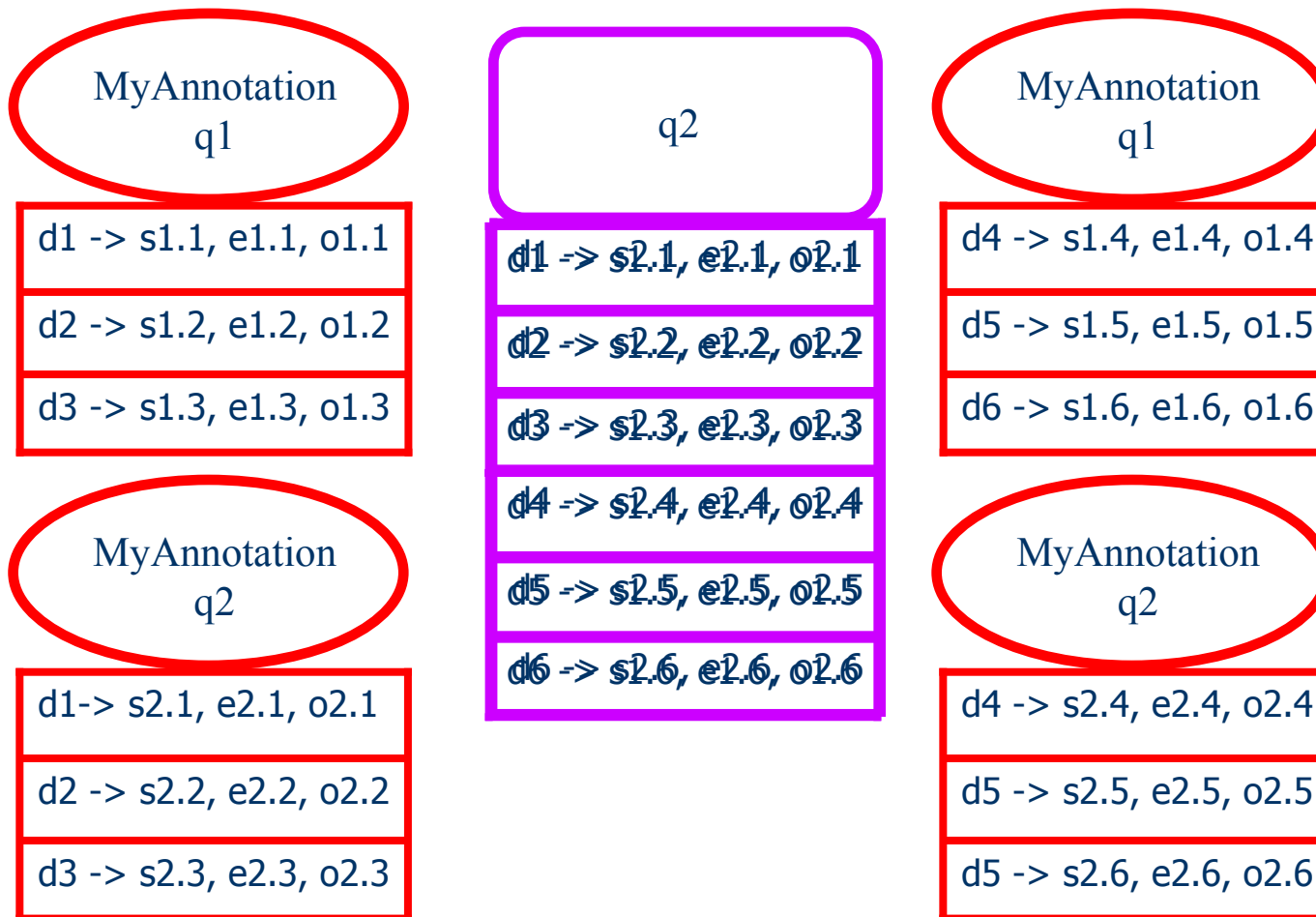
Nodes:



GeB: Building of Result (2/3)



GeB: Building of Result (3/3)

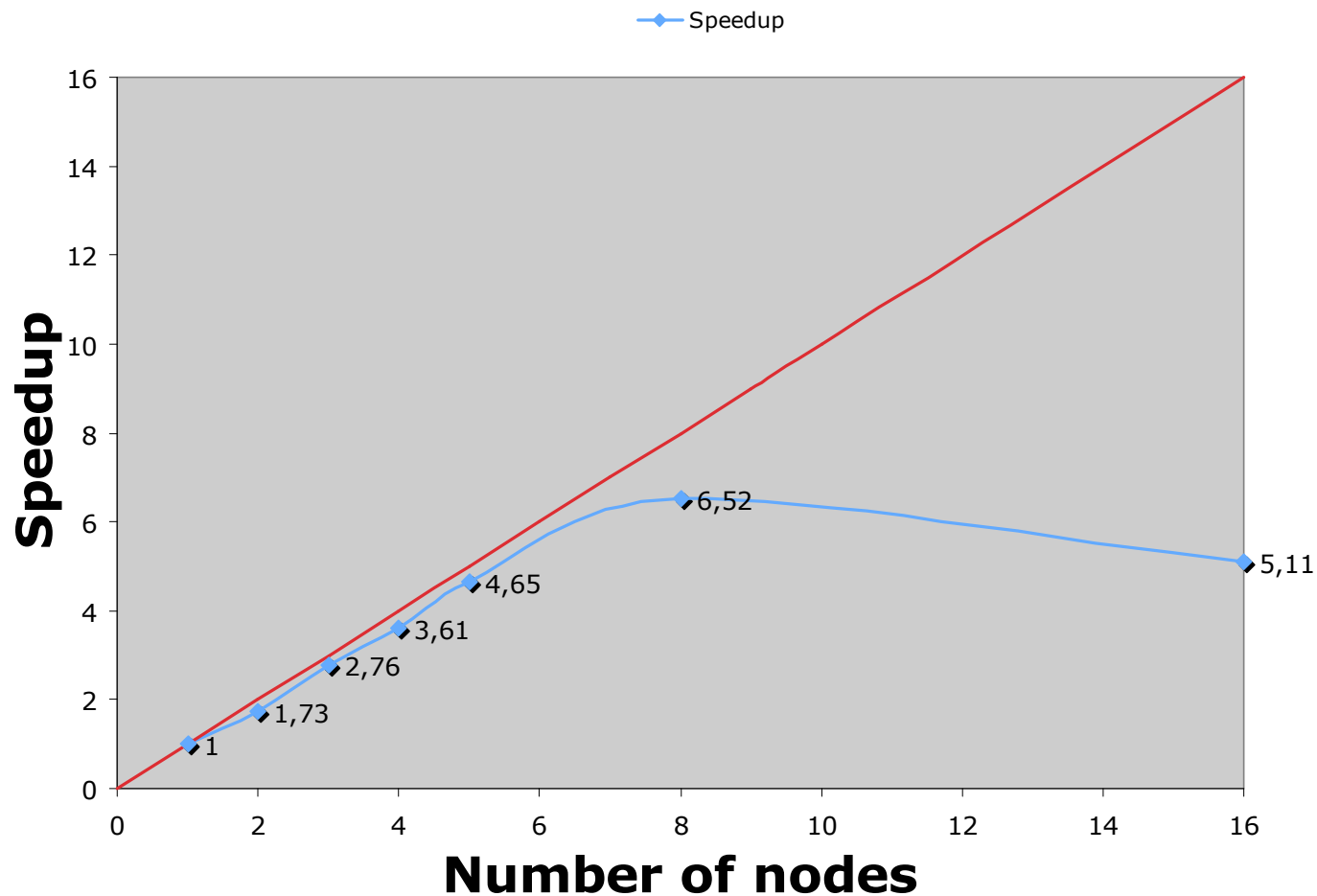


Result for query sequence q1
Partial Result
From Node 1

Result for query sequence q2
Partial Result
From Node 2

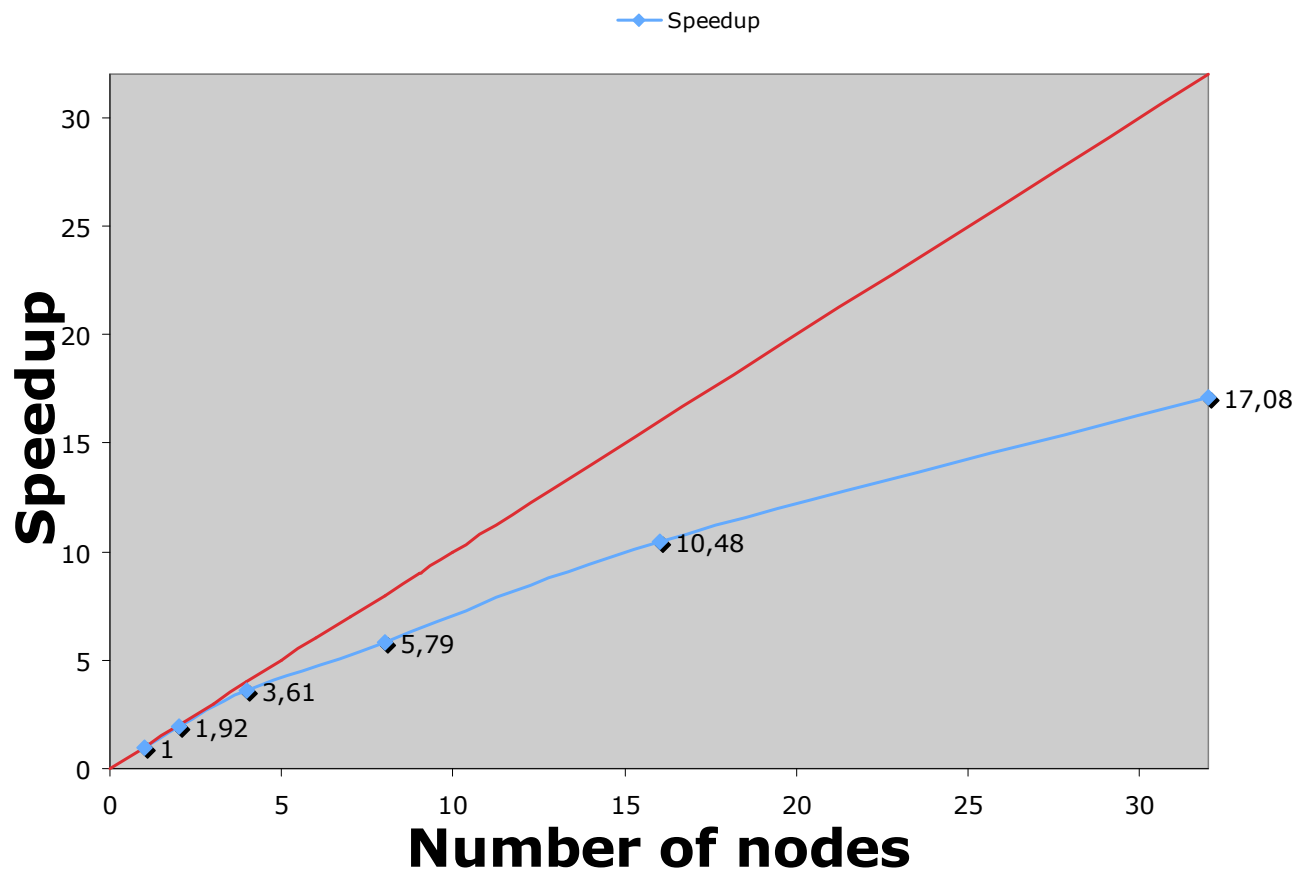
Benchmark Results: Desktop Computers

**The benchmark result on desktop
computers**



Benchmark Results Cluster

The benchmark result on cluster linux-pf



Summary and Future Roadmap

- Initial results encouraging
 - GeB is *scalable* (checked on 39 processors)
 - can run in both *cluster* and *desktop* environment
 - good speedup for small number of processors **BUT** the performance degrades for large number of processors
 - **NEED FOR LOAD BALANCING**
- **Future Roadmap**
 - To work on the proper load balancing to gain better-speedups
 - Final packaged release



What else?

Thank you!