



Enabling Grids for E-science

Performance Analysis and Optimization of AMGA for the WISDOM environment

Sunil Ahn

KISTI e-Science Department

www.eu-egEE.org

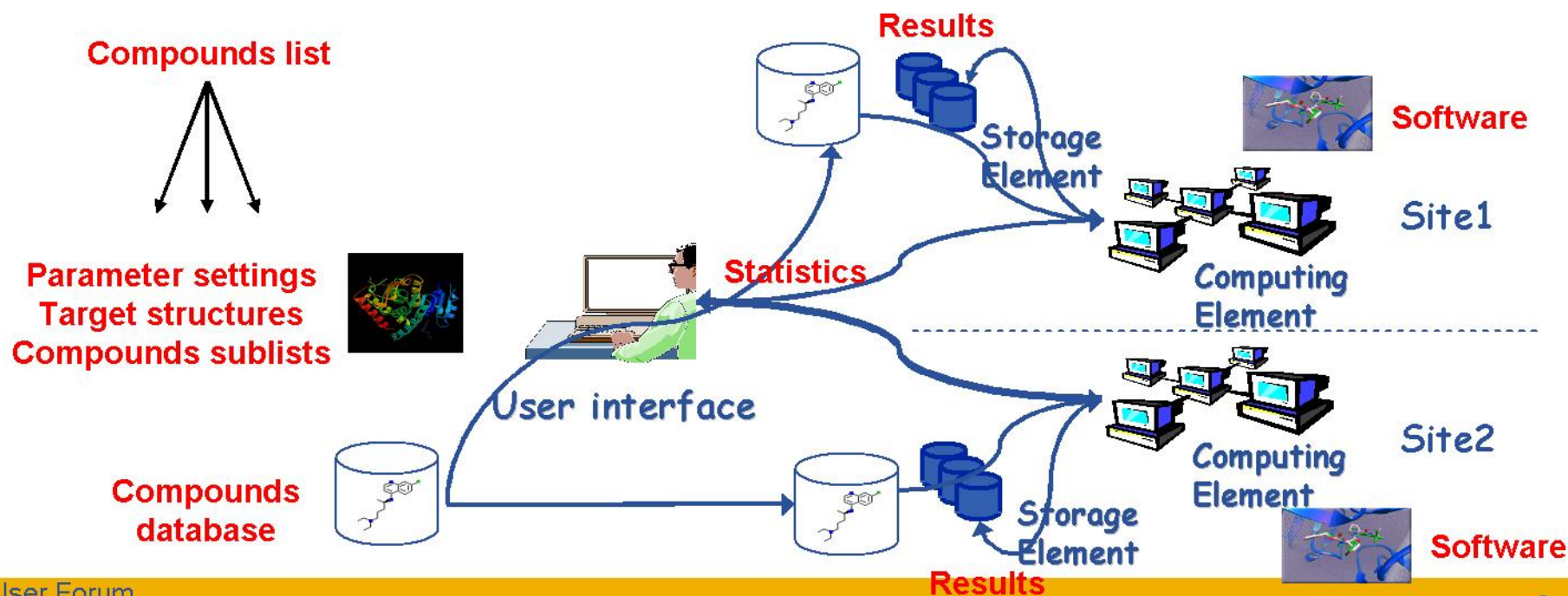


Information Society
and Media



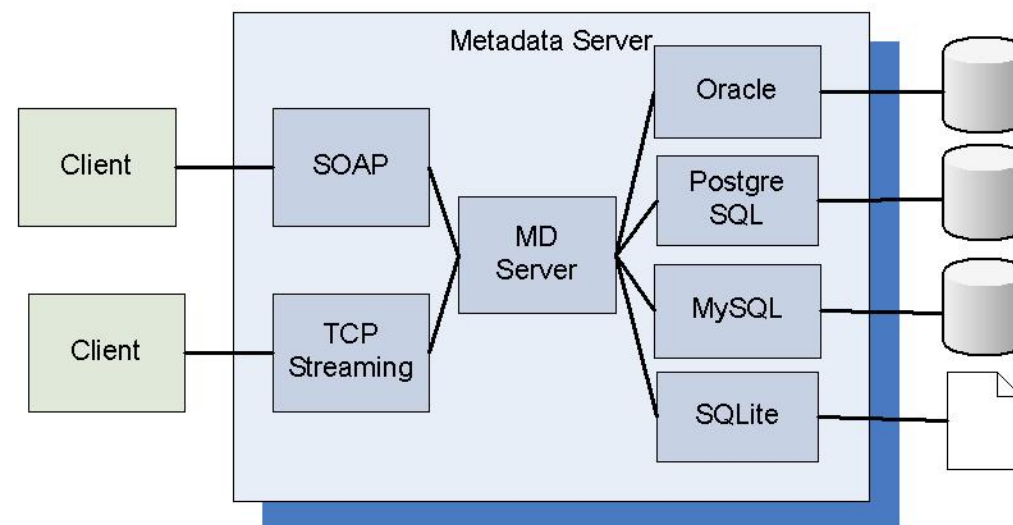
- **I. Introduction**
- **II. AMGA Performance Analysis for WISDOM**
- **III. TASK Distribution Issue with the use of AMGA**
- **IV. Conclusion**

- **What is WISDOM ?**
 - Grid-enabled virtual screening initiative
 - Search for new drug using grid infrastructure
- **WISDOM environment**
 - Producing a large amount of data in a limited time with a minimal human cost during the data challenge.

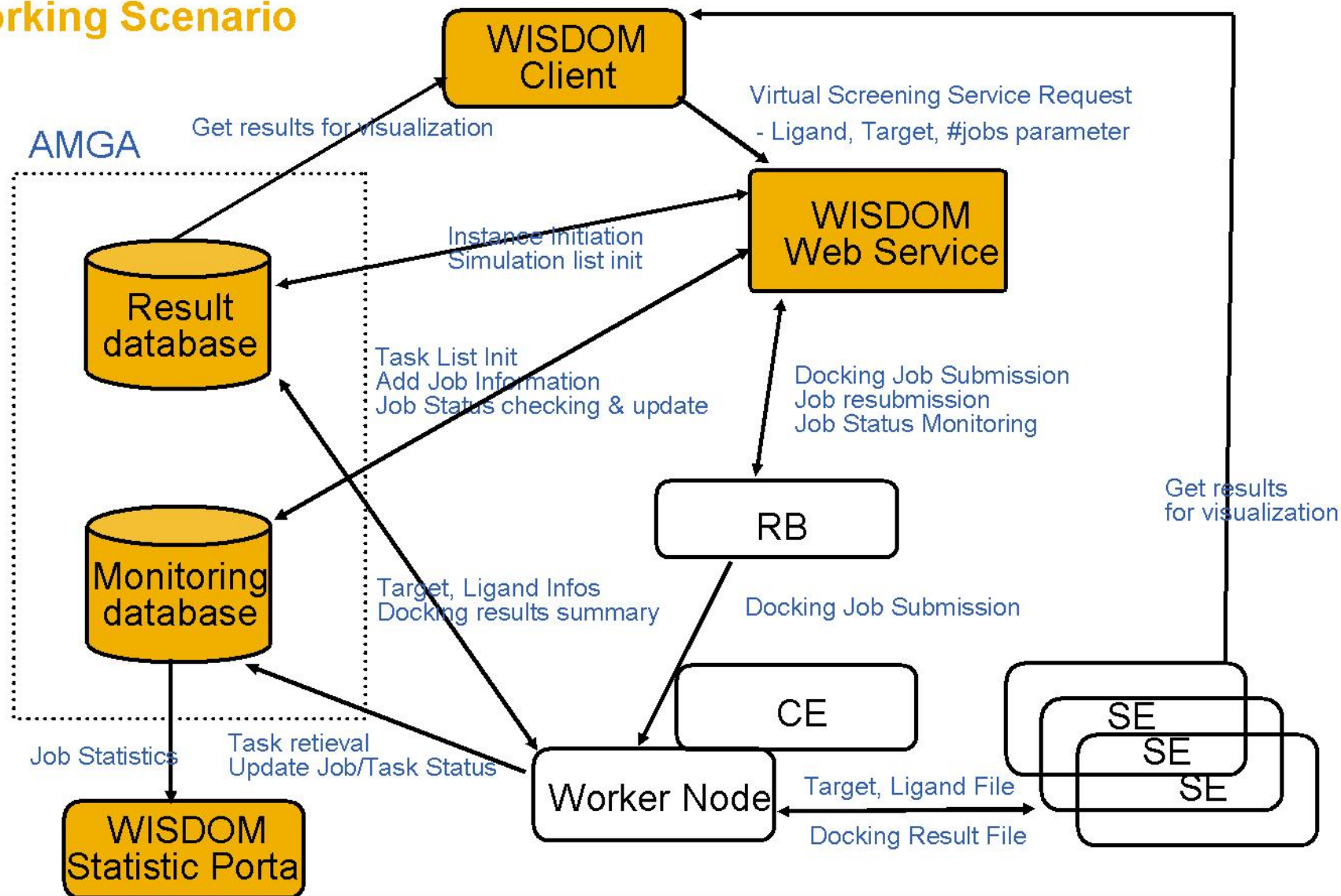


What is AMGA (ARDA Metadata Grid Application) ?

- AMGA is the Metadata Catalogue for gLite:
- AMGA is included in gLite release 3.1
- AMGA in preproduction within several projects:
 - LHCb and ATLAS: GANGA
 - EGEE BioMed applications
 - medical images metadata
- Master/Slave replication model is supported

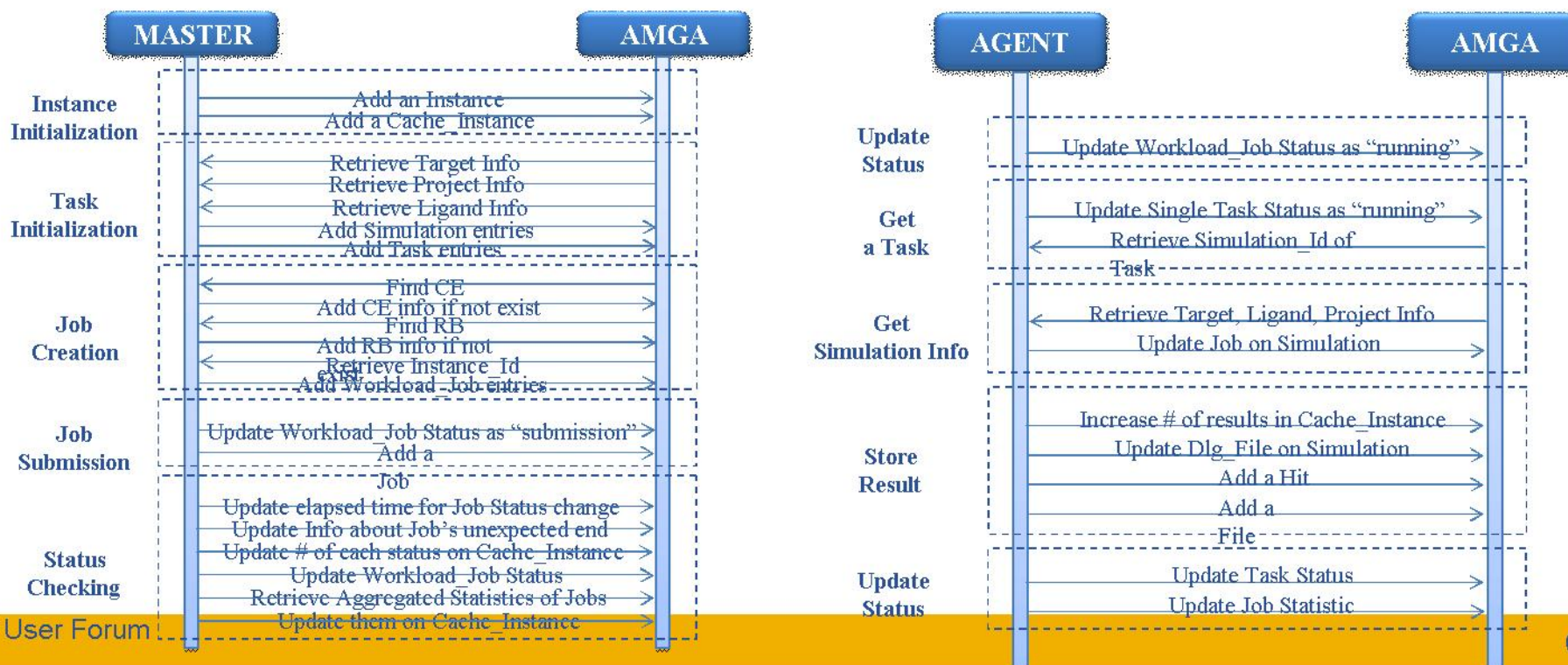


Working Scenario



Throughput Requirement on AMGA

- Required Throughput = (# agents * # AMGA access per docking) / Time taken per docking
- 133 requests per second = 4000 * 10 / 300 (4000 WNs)
- 665 requests per second = 20000 * 10 / 300 (20000 WNs)



Performance Measurement in LAN

- **Objective**
 - Evaluate AMGA provides enough throughput for WISDOM
 - analyze where and how much overhead is located by within AMGA
- **5 Test Cases**

	Test Case	Description
Case1	Access DB through AMGA Connection closed after querying one entry With GSI/SSL Connection (without session caching)	Multiple client threads access AMGA concurrently through the use of GSI/SSL authentication
Case2	Access DB through AMGA Connection closed after querying one entry With GSI/SSL Connection (with session caching)	To evaluate the performance improvement of session caching mechanism in AMGA
Case3	Access DB through AMGA Connection closed after querying one entry Without GSI/SSL Connection	To figure out overhead of handling GSI/SSL authentication in AMGA
Case4	Access DB through AMGA Not Closing Connection after querying one entry Without GSI/SSL Connection	To figure out overhead of handling network connections in AMGA
Case5	Direct Access to DB, Without AMGA Not Closing Connection after querying one entry Without GSI/SSL Connection	To compare the performance of AMGA with the performance of direct DB access

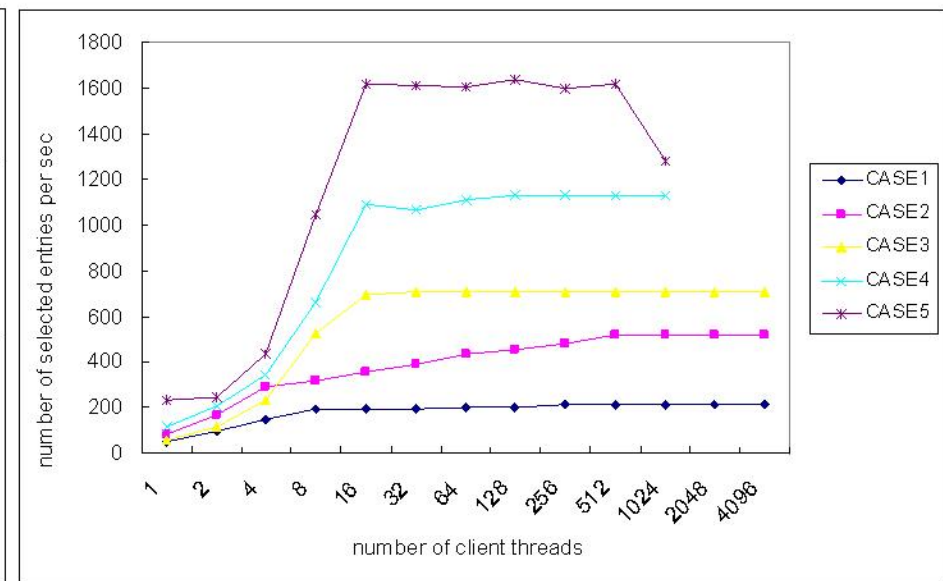
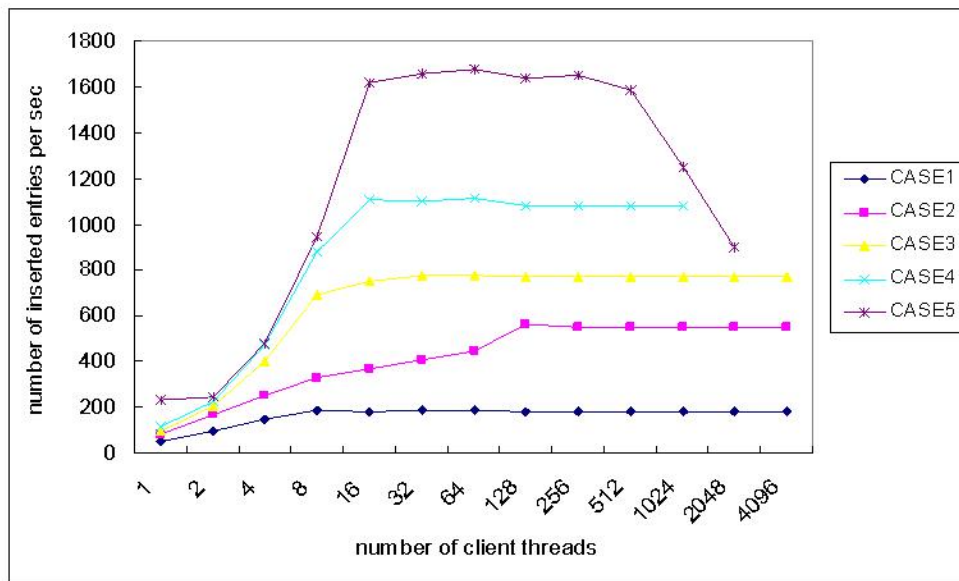
Performance Measurement in LAN (continued)

- **Testing Environment**

- 1 PostgreSQL(v8.2.5) DB server, 1 AMGA(v 1.3) server, 4 client nodes

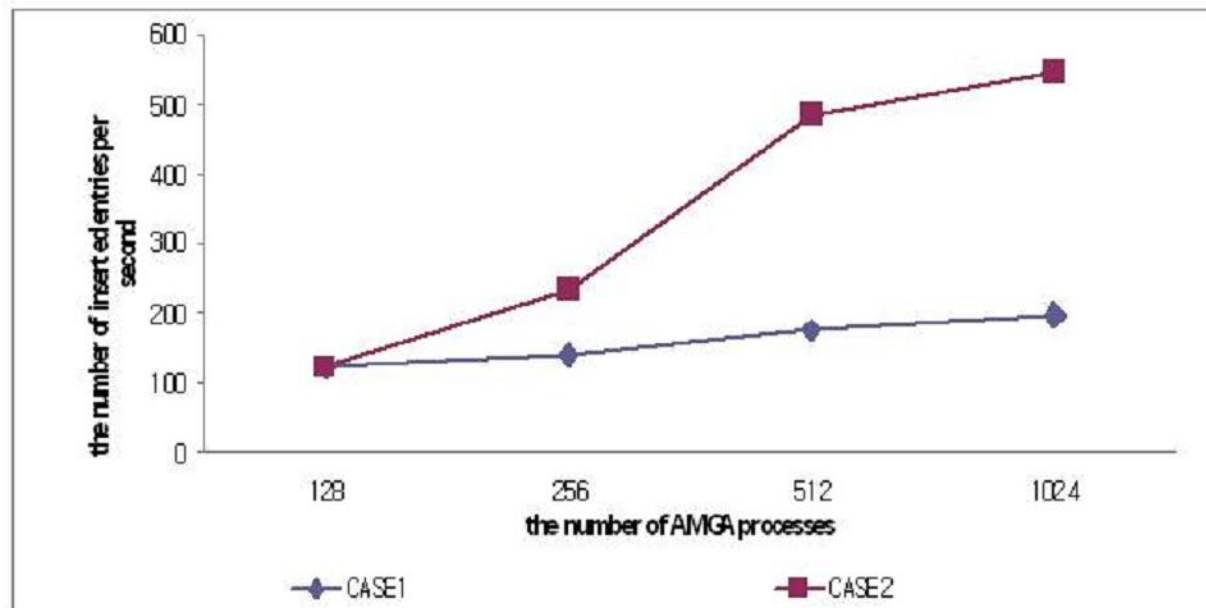
- **Results**

- About 600 requests per second could be handled
- Enough throughput for 18,000 agents
- 2.67 times slower than the direct DB access



Performance Measurement in WAN

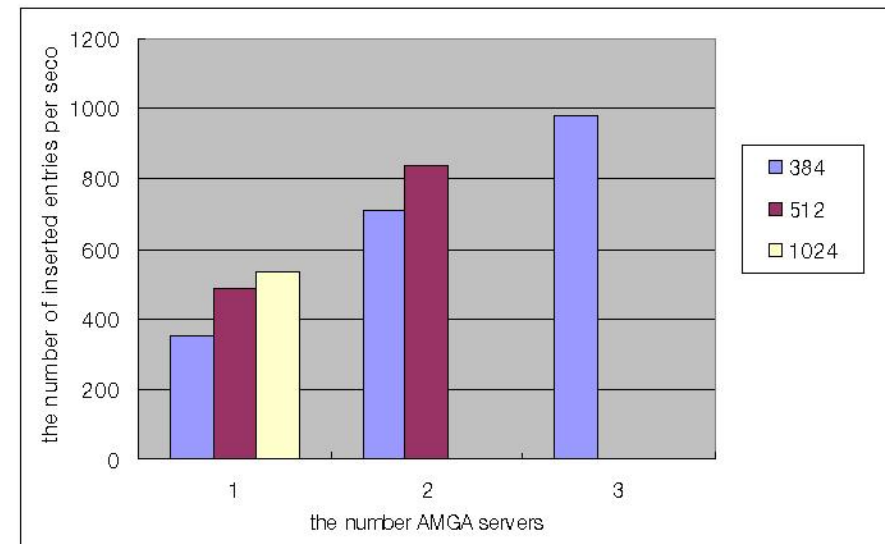
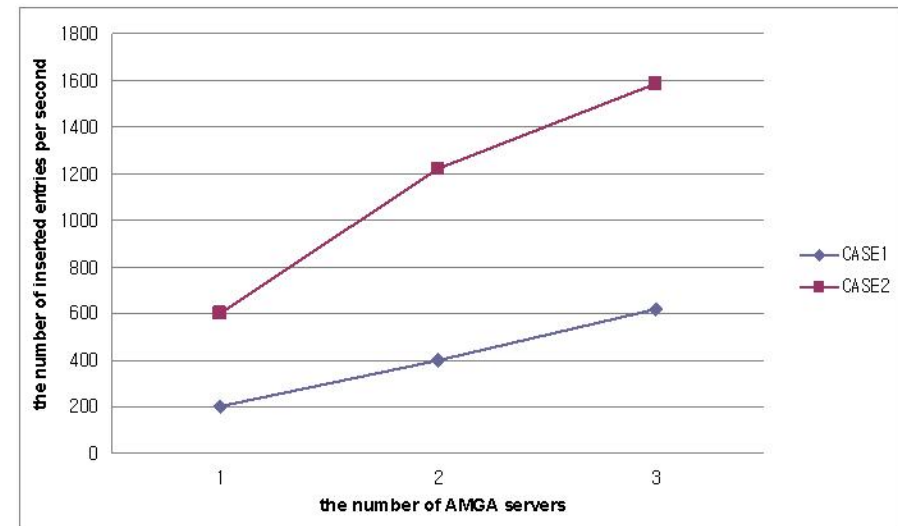
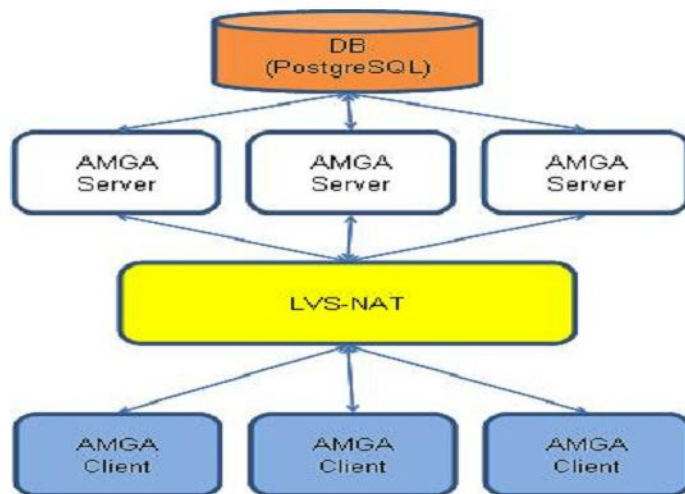
- **Testing environment**
 - DB & AMGA servers at KISTI site & 4 client nodes at CERN site
- **Results**
 - The number of AMGA processes is the major influential factor in WAN
 - Enough throughput for about 15000 WISDOM agents in WAN too
 - However, if the number of agents is increased further ?



II. AMGA Performance Analysis

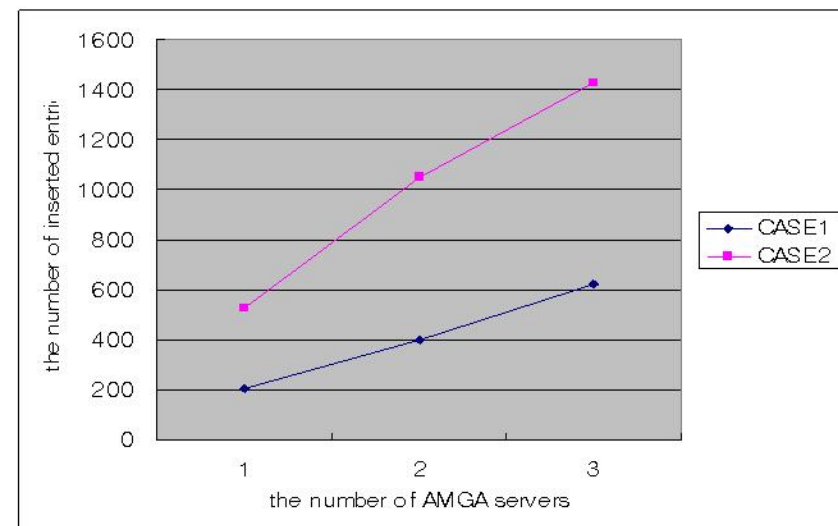
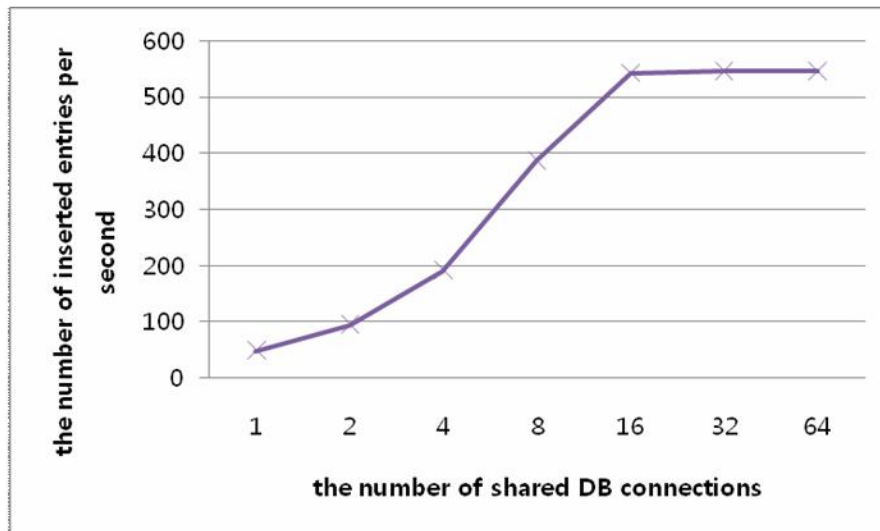
Optimization (Load balancing)

- linear scale-up almost up to the limit of direct DB access in LAN
- 2.6 ~ 3 times with 3 AMGA servers
- 1.8 times in WAN



Optimization (DB Connection Pool)

- **Throughput limitation in WAN**
 - Load balanced AMGA Throughput is limited by the number of DB connections in WAN
- **DB connection pool integrated into AMGA**
- **Results**
 - linear scale-up almost up to the limit of direct DB access in WAN too, improving AMGA throughput 2.6 ~ 3 times with 3 AMGAs



Requirement on AMGA

- **Scalability**
 - If the number of agents are increased further ?

What is the problem ?

- **AMGA is used as a placeholder for a task distribution table**
 - Task Pull model
- **Task retrieval process from a job is as follows**
 - Lock the task table
 - Select a task which state is waiting
 - Update the state of task to working
 - Release the lock
- **Serious performance degrade due to the overhead by the need to lock the whole table**
 - About 1 task per second could be retrieved when tested in WAN

Optimization

- **Addition of a new AMGA API for task retrieval**
 - updateattr_single
- **Performance Measurement**
 - 1 AMGA server at KISTI site, 3 client nodes at CERN sites
 - 80 tasks per second retrieved
 - Throughput was independent on the number of task entries with the use of index

Requirement on AMGA

(1) providing a generic server side programming interface

- **performance issues addressed on the use of AMGA in the WISDOM environment**
 - some techniques presented to drastically improve the throughput of AMGA
- **The techniques are expected to be integrated in the new release of WISDOM environment,**
 - being deployed in the EGEE biomed VO infrastructure for the next WISDOM data challenge
- **Future Work**
 - Co-Develop with AMGA team for those issues addressed in this talk

- Questions ?

Role of KISTI for WISDOM

- **KISTI**

- the national computing resource center for research in Korea
- Promoting e-Science infrastructure

- **Role of KISTI for WISDOM**

- Development of the WISDOM environment related to AMGA
 - Task distribution, AMGA Fault Tolerance, AMGA Optimization for WISOM
 - Cooperation with AMGA team
- AMGA Deployment and Service for WISDOM
- Development of WISDOM UI