ISBN 978-89-6211-607-6

다중클라이언트를 지원하는 GLOVE 시스템 설계 및 구현

(GLOVE system design and implementation for multiple client)

김 민 아 (petimina@kisti.re.kr)

한 국 과 학 기 술 정 보 연 구 원
Korea Institute of Science & Technology Information

목차

1. 개 요 ····· 1

2. GLOVE의 시스템 구조 ····· 2

가. GLOVE Overall Architecture ····· 2

나. 데이터 관리를 위한 계층 구조 ····· 3

3. GIP (GLOVE Interface Protocol) ····· 4

4. GIVI (GLOVE Integrated Visualization Interface) ····· 6

5. GLORE ····· 7

6. GDM (GLOVE Data Manager) ····· 7

7. 로터 시뮬레이션 데이터 가시화 ····· 9

가. 대용량 데이터를 위한 로딩 기능 ····· 10

나. 블레이드 데이터 가시화 기능 ····· 10

다. 필드 데이터 가시화 기능 ····· 11

라. 로터 시뮬레이션 데이터 가시화 결과 ····· 12

8. 결론 ····· 14

9. 참고문헌 ····· 15

그림 차례

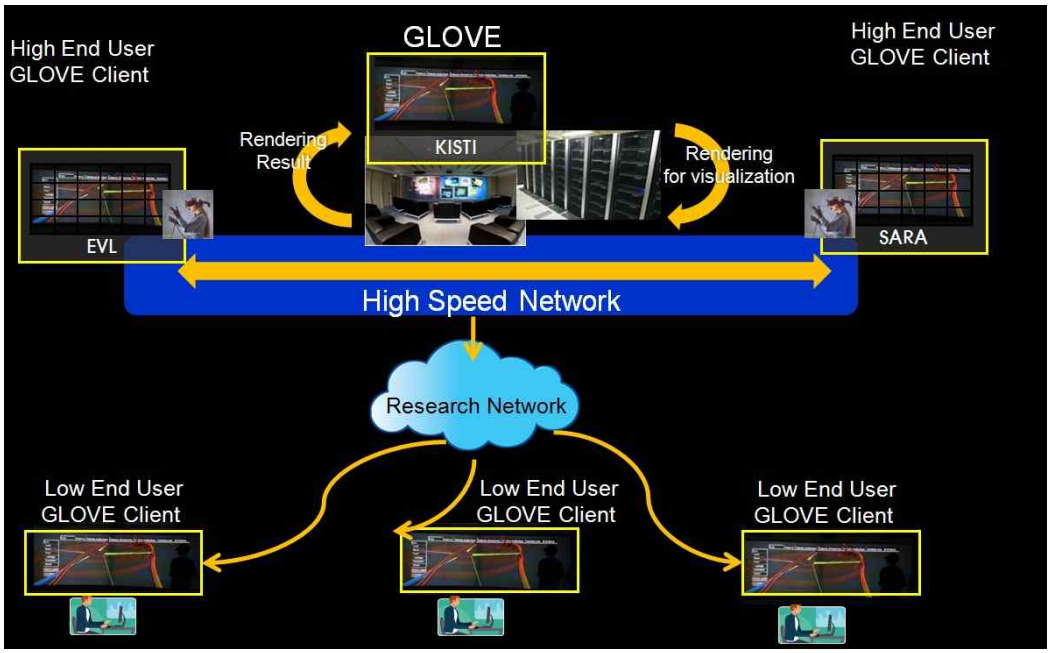
[그림 1] GLOVE 시스템 ····· 1
[그림 2] GLOVE Architecture ····· 2
[그림 3] GLOVE Data Management Hierarchy ····· 3
[그림 4] GIP Architecture ····· 4
[그림 5] 다중 클라이언트를 지원하는 Remote Rendering Protocol GIP ·· 5
[그림 6] GIP Session negotiation ····· 5
[그림 7] VR 환경에서의 GIVI ····· 7
[그림 8] GDM 데이터 처리 동작 ····· 9
[그림 9] 동체 데이터 가시화 결과 ····· 12
[그림 10] 1.21 테라바이트 로터 시뮬레이션 데이터 가시화 결과 ····· 13

표 차례

[표 1] Loading time ····· 8
[표 2] 6GB, 30 time step 로터 데이터 성능 측정 ····· 13
[표 3] 1.2TB, 90 time step 로터 시뮬레이션 데이터 성능 측정 ····· 14

1. 개 요

GLOVE(Global Virtual Reality Visualization Environment for scientific simulation)는 컴퓨팅 자원의 성능향상으로 데이터양이 급속히 증가한 응용과학과 전산 시뮬레이션 분야에 대용량 과학데이터를 효율적으로 가시화하여 분석하기 위한 가시화 도구이다. GLOVE는 응용과학자의 연구실에서 원격으로 병렬 렌더링 서버로 가시화를 요청할 수 있을 뿐만 아니라, 대용량 데이터를 고해상도로 볼 수 있는 tiled display, 다양한 데이터의 특성을 파악할 수 있는 VR환경의 사용자 인터페이스를 제공한다. 또한, 대용량 데이터 병렬처리를 위해 분산 공유 메모리를 제공하는 Global Array를 도입하여 테라바이트 단위의 데이터를 실시간으로 처리할 수 있도록 한다. 본 문서에서는 고성능 컴퓨팅 환경에서 대용량 시뮬레이션 데이터를 효과적으로 분석하고 가시화하여 공유할 수 있는 프레임워크인 GLOVE에 대해 소개한다.



[그림 1] GLOVE 시스템

GLOVE는 다음과 같은 특징을 갖는다.

(1) VR 기반 통합 환경 내에서 데이터의 정성적, 정량적 분석.

(2) 병렬 처리를 통한 대용량 데이터의 효율적 분석 및 가시화

(3) 단일 모니터에서 가상현실 장비에 이르기까지 다양한 형태의 가시화 장비 지원

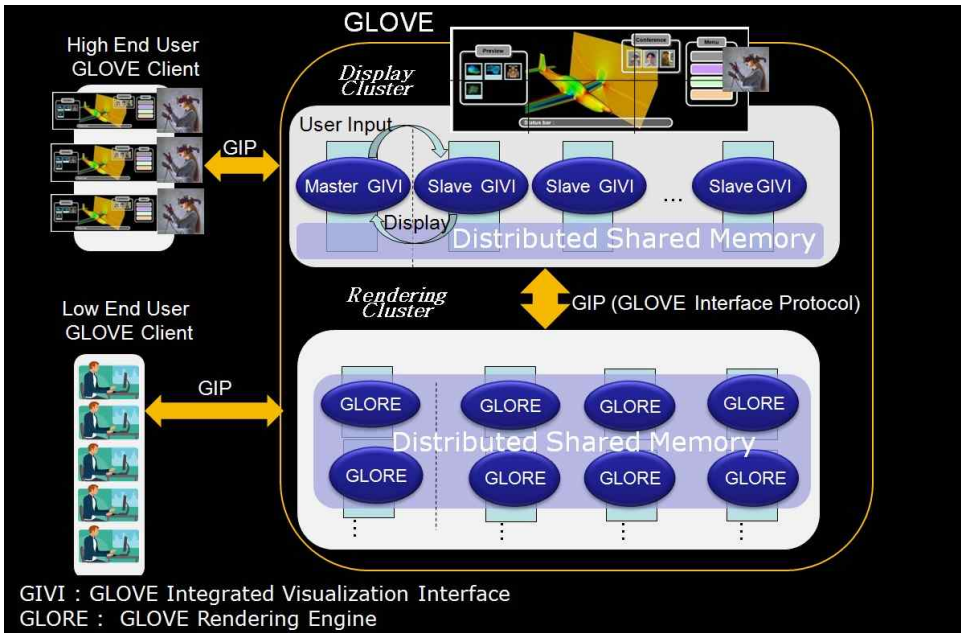
(4) 대용량 데이터 가시화 서비스 제공

GLOVE는 이러한 기본 프레임워크 위에 사용자 인터페이스를 각 응용분야에 특화시켜서 해당분야 연구자들이 접근하기 용이한 통합 환경을 제공한다. 본 문서에서는 다양한 응용 및 다중 클라이언트를 지원하는 GLOVE 의 프레임워크를 설계하고, 개발한다.

2. GLOVE의 시스템 구조

가. GLOVE Overall Architecture

그림 2는 GLOVE의 기본 구조를 보여 준다. GLOVE는 사용자 인터페이스인 GIVI(GLOVE Integrated Visualization Interface)와 데이터 가공 및 렌더링을 담당하는 GLORE(GLOVE Rendering Engine)의 두 부분으로 나뉜다. GIVI는 가상현실 입출력 장치를 총괄하며, 사용자로부터 입력을 받아서 GLORE에 전달하고 GLORE의 실행 결과를 전달 받아서 화면에 출력한다. GLORE는 대용량 데이터의 가공 및 렌더링을 위한 GIVI의 서브시스템이다.



[그림 2] GLOVE Architecture

GLOVE에서 GLORE와 GIVI는 서로 독립적으로 운영되는 클러스터에서 실행된다. 이러한 구조는 GLOVE의 인터페이스(GIVI)와 핵심 렌더링 기능(GLORE)을 물리적으로 분리할 수 있기 때문에, 충분한 그래픽 처리능력을 갖추지 못한 시스템을 가지고 있는 사용자가 자신이 보유하고 있는 시스템에서 GIVI를 실행하고,

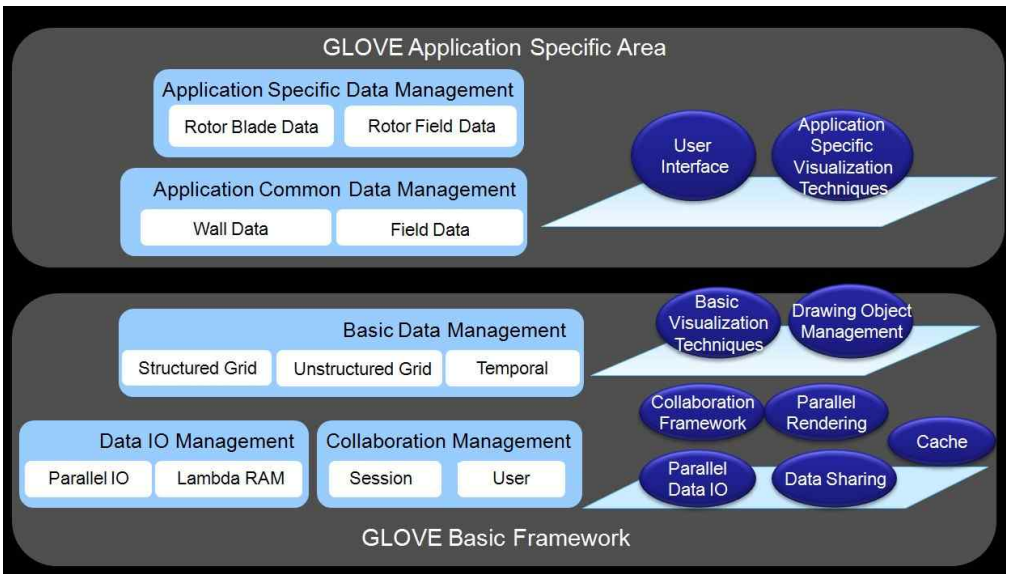
원격지의 고성능 렌더링 엔진인 GLORE를 이용해 대용량 데이터를 렌더링 할 수 있다.

나. 데이터 관리를 위한 계층 구조

GLOVE는 범용적인 가시화를 위한 프레임워크 위에 각 어플리케이션의 특성을 반영하는 맞춤형 사용자 인터페이스를 제공한다. 이를 구현하기 위해 GLOVE에서는 데이터를 계층적 구조로 관리한다. 그림 2는 GLOVE의 데이터 계층구조를 보여준다.

데이터 계층 구조의 가장 아랫부분에 위치한 Data I/O management 계층에서는 데이터의 병렬 입출력 및 공유 캐시(cache) 등의 메커니즘을 관리한다.

Basic data management 계층에서는 가시화를 위한 structured grid, unstructured grid 등 기본적인 메시 구조와 다중 블록(multi-block) 데이터 및 시변환(time-variant) 데이터를 위한 자료구조를 관리한다.

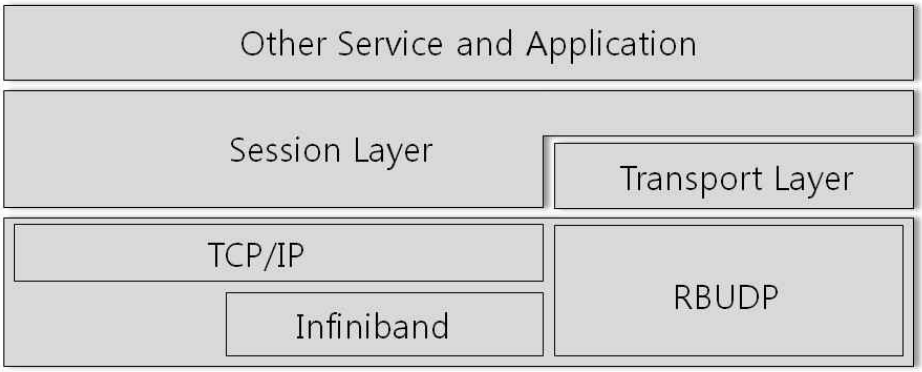


[그림 3] GLOVE Data Management Hierarchy

Application common data management 계층에서는 GLOVE가 지원하는 응용분야에서 공통으로 활용하는 자료구조를 다룬다. CFD에 있어서는 헬기의 표면을 나타내는 벽면데이터와 기류의 흐름을 나타내는 유동 데이터를 이 계층에서 정의할 수 있다. 의료 영상과 같은 다른 분야의 응용을 추가할 경우에는 이 계층부터 새로운 자료구조를 정의한다. Application specific data management 계층에서는 특정 응용분야에서 사용하는 자료구조를 관리한다. 로터 시뮬레이션 데이터 해석을 위한 가시화의 경우 로터 블레이드 데이터, 로터 필드 데이터 등을 이 계

층에서 정의한다.

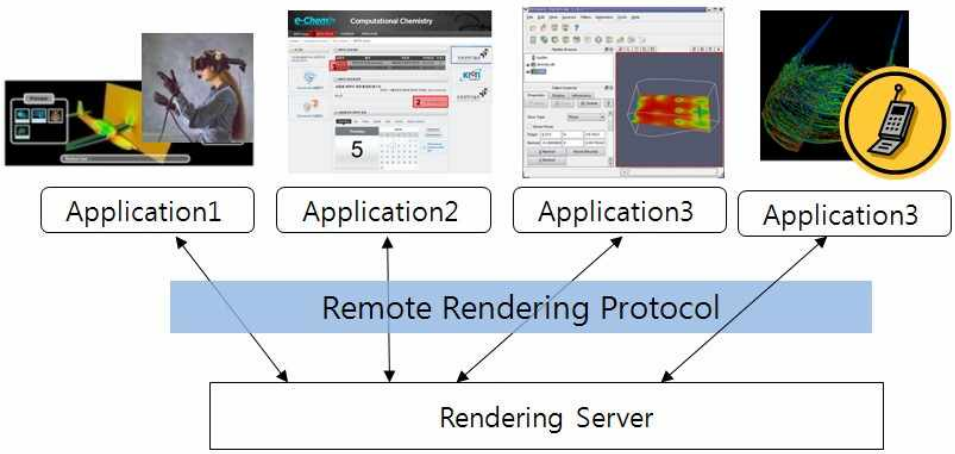
3. GIP (GLOVE Interface Protocol)



[그림 4] GIP Architecture

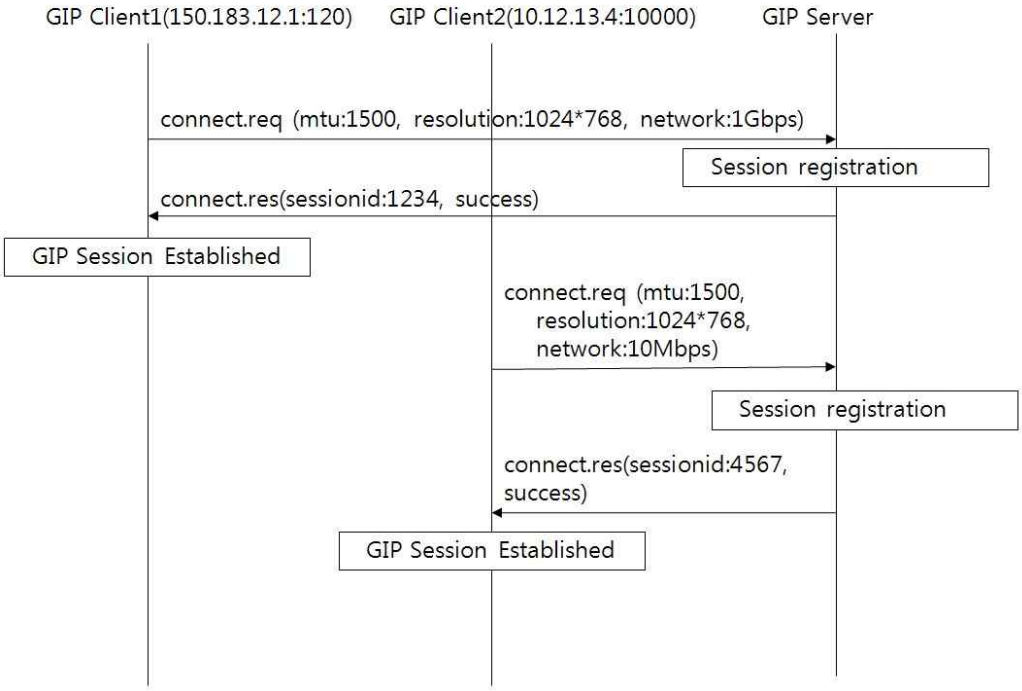
GIP은 원거리에 있는 GIVI와 GLORE 간의 통신을 위한 TCP/IP 기반의 프로토콜이다. 프로토콜은 request와 response로 나누어지며, request는 응용 맞춤형 메뉴에 따르는 것이 아니라 streamline, iso-surface, pathline 등 가시화의 형태에 근거한다. 따라서, 가시화할 응용 데이터가 바뀌고 메뉴가 바뀐다할지라도, GIP 메시지는 변하지 않는다. GIP을 사용하는 응용 프로그램이 응용에 맞게 요청을 구성하고, 결과를 해석한다. GIP은 GIVI와 GLORE 간의 통신 외에도 다수의 디스플레이 노드를 가진 tiled display 환경에서는 master GIVI와 slave GIVI 간의 통신에도 사용된다. 이 때문에, 고성능 내부 네트워크인 Infiniband 와 같이 인프라 네트워크로 지원한다.

GIP은 WAN 환경을 지원함으로써, 원거리에 있는 여러 GIVI 클라이언트를 수용하며, 응용 독립적으로 설계되어 다양한 응용을 수용할 수 있다. 렌더링 서버인 GLORE는 GIP 프로토콜로만 접근한다면, 그림 5와 같이 어떠한 응용, 어떠한 device 라도 수용 가능하다. 이를 위해 GIP은 세션을 관리한다. GIP의 세션 연결은 클라이언트와 서버 사이의 negotiation을 수행한다. GIP 서버로 동작하는 GLORE는 세션 등록 시 클라이언트 디바이스와 네트워크 상황을 저장하여 클라이언트에 맞는 데이터를 전송할 수 있다.



[그림 5] 다중 클라이언트를 지원하는 Remote Rendering Protocol GIP

GIP 세션이 맺어진 후, GIP 메시지를 전송할 때 마다 보내는 GIP 세션 아이디는 클라이언트를 유일하게 구분할 수 있는 아이디이다. GIP 서버는 이 세션 아이디를 관리함으로써 여러 클라이언트에 대한 요청을 처리할 수 있다. 세션 아이디는 클라이언트 주소와 접속 포트 서버 주소와 접속 포트의 쌍으로 생성하여 관리한다.



[그림 6] GIP Session negotiation

그림 6은 세션 생성 시 GIP서버의 동작을 보여 준다. connect.req에 클라이언트

는 자신의 mtu 크기와 화면 resolution, 네트워크 속도 등의 정보를 전달하면, 서버는 이 정보를 세션 테이블에 저장하여 결과 전송 시 참조한다. 또한 클라이언트 주소와 서버 주소의 쌍으로 만들어진 session id를 connect.res에 전송하여, 클라이언트가 이를 다음 GIP 메시지마다 첨부하여, 응답 시 세션 테이블을 참조할 수 있도록 한다.

현재, KISTI의 VR 환경은 16개의 display node를 가지며, 이들 노드에 동일한 가시화 결과가 전달되어야 한다. 이러한 경우, GLORE가 WAN으로 연결된 16개의 노드에 모두에 데이터를 보내는 것은 비효율적이다. 따라서, GLORE는 master GIVI와 통신하고, master GIVI가 나머지 16대의 노드에 데이터를 전달한다. 현재 KISTI는 Infiniband로 16대의 노드가 연결되어 있으며, GIP은 Infiniband를 지원한다.

4. GIVI (GLOVE Integrated Visualization Interface)

고해상도의 디스플레이는 가시화 데이터가 커질 경우 시각적으로 판별할 수 있는 직관적 분석의 정밀도를 높여 줄 수 있다. 또한 VR환경은 3D로 가시화 결과를 보여줌으로써, 또 다른 분석의 가능성을 제공한다. 예를 들어 로터 데이터 분석의 경우, 유동 기류가 로터를 치고 빠져 나가는지에 대한 분석은 3차원일 경우 확인이 더욱 용이해 진다.

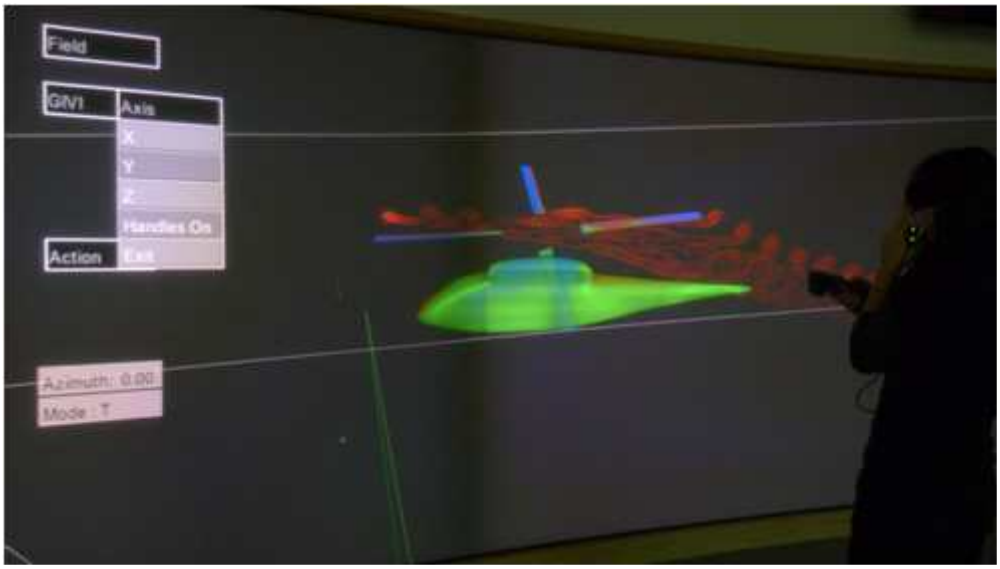
GIVI는 VR 환경을 제공하는 프레임워크인 VRJuggler로 구현하여 키보드와 마우스뿐만 아니라 IS-900, CAVE, tiled display와 같이 특수한 형태의 가상현실 입출력 장치를 지원한다. 따라서 사용자는 책상 위의 PC부터 CAVE나 고해상도의 대형 tiled display에 이르는 다양한 가시화 시스템에서 GLOVE를 동일한 형태로 실행할 수 있다.

GIVI는 VRJuggler 어플리케이션의 동작 매커니즘에 따라 master 프로세스와, master와 동일하게 동작하는 슬레이브 프로세스로 구성된다. 여러 대의 컴퓨터와 디스플레이로 구성된 대형 가시화 시스템에서 동작할 경우, 서로 다른 GIVI 슬레이브 프로세스가 각 디스플레이 노드의 렌더링을 담당한다.

또한, GIVI의 메뉴는 응용 맞춤형으로 해당 응용분야의 사용자의 요구를 반영하여 수정가능하다. 현재 GIVI는 로터 데이터 시뮬레이션을 위한 사용자 인터페이스만을 지원하지만, xml로 메뉴 구성을 지원하여, 다른 응용으로의 확장이 용이하다.

그림 6은 Head tracker 와 wand 등 VR 입력장치와 함께, 8개로 분할된 tiled di

splay에 하나의 가시화 결과를 보여 주는 GLOVE 시스템이다. 전체 스크린의 해상도는 7308*2116 이며, 각 tiled display 는 좌측, 우측 view를 담당하는 2개의 노드와 연결되어 모두 16개의 노드에 동일한 가시화 데이터가 존재한다.



[그림 7] VR 환경에서의 GIVI

5. GLORE

GLORE는 GIVI의 사용자 요청을 받아 렌더링을 수행하고, 그 결과를 GIVI에 전달하는 렌더링 엔진이다. GLORE는 master 노드가 사용자 요청을 받아 여러 대의 slave 노드에서 실행하여 그 결과를 GIVI로 돌려준다.

또한, 대용량 데이터의 처리를 위해 다수의 노드를 스케줄링하여 병렬처리를 수행한다. GLORE는 응용 데이터에 독립적이며, GIVI에서의 사용자 요청을 전달하는 GLOVE Interface Protocol인 GIP으로 요청을 받고 GIP으로 결과를 GIVI에 전달하는 GIP 서버의 역할을 수행한다.. GLORE는 데이터 사이즈에 따라 한 노드 내에서도 병렬처리를 수행한다. 효율적인 병렬 렌더링 스케줄링과 대용량 데이터를 위해 GLOVE Data Manager (GDM)을 사용하여 데이터를 관리한다.

6. GDM (GLOVE Data Manager)

대용량 데이터의 경우 가시화 도구의 성능은 매우 중요한 요소이다. 이 때문에 사용자들은 될 수 있는 한 많은 양의 데이터를 메모리에 올려놓고 작업을 수행하기를 원한다. 그러나, 하나의 노드가 사용할 수 있는 메모리의 양은 정해져 있고 이 때문에 사용자들은 데이터가 클 경우 데이터의 로딩 시간과 처리 시간을

참아 낸다. 데이터에 따라 분석 자체를 포기할 수밖에 없는 경우도 발생한다. 따라서, 대용량 데이터를 처리하는 가시화 도구에 있어 가장 큰 쟁점은 데이터의 효율적 관리와 처리이다. 가시화에 있어 가장 많은 시간을 요하는 것이 데이터를 로드하는 데 걸리는 시간이다. 표 1은 GDM을 이용하여 노드들이 병렬 입출력으로 데이터를 로딩할 때와 1노드가 데이터를 로드할 때 걸리는 시간을 보여준다. GDM은 현재 가시화 도구에서 처리 불가능한 데이터를 처리할 수 있게 해준다.

[표 1] Loading time

Data Size	1 node (sec)	GDM(sec)
1.09GB	4.69	0.56
6.13GB	31.31	2.14
30.22GB	처리불가	25.914
1.21TB	처리불가	158.408

GDM은 GLOVE의 가시화를 위해 분산 공유 메모리를 제공한다. 테라바이트 단위의 데이터는 하나의 노드에서 처리 불가능하다. 따라서, 일반적으로 이러한 데이터를 가시화할 때 데이터를 미리 각 노드에 나누어 두고, 데이터가 있는 노드가 처리할 수 있도록 병렬처리를 스케줄링한다. 그러나, 이러한 방식의 static data decomposition의 경우 컴퓨팅 자원이 부족해도, 데이터를 가지지 않는 노드에는 작업을 분배할 수 없다.

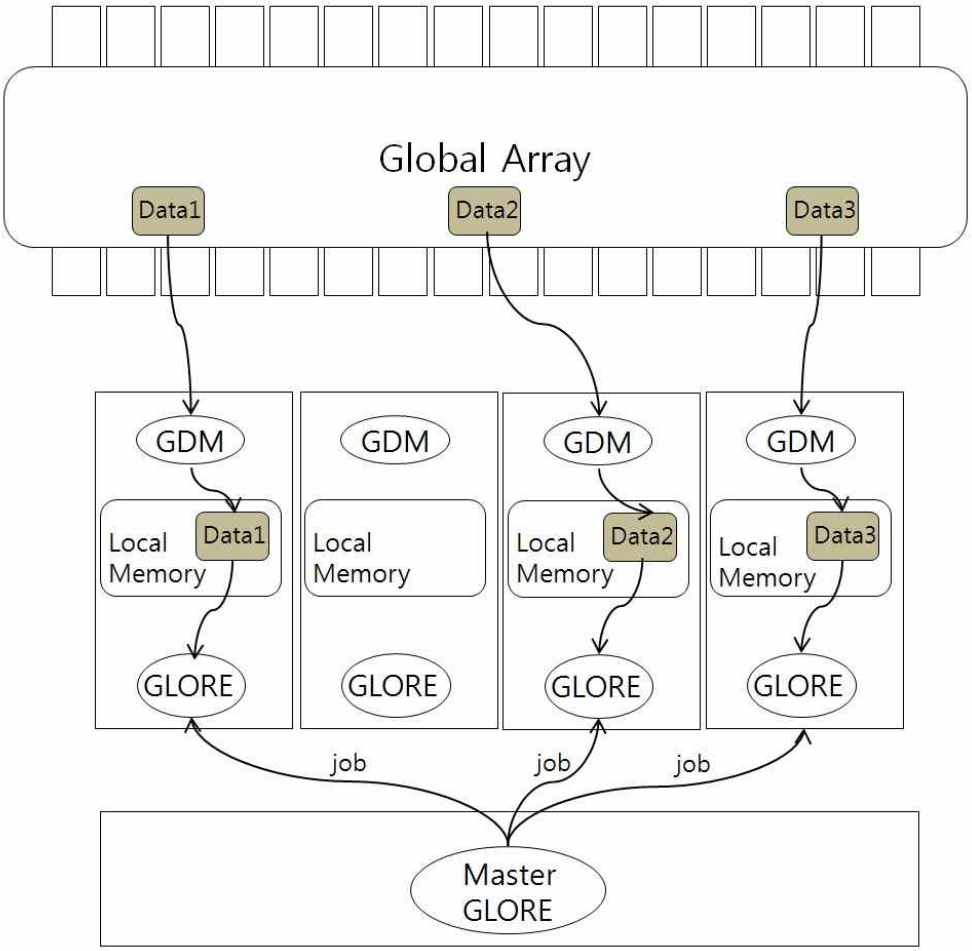
GDM은 GLORE가 이러한 data의 decomposition을 고려하지 않고, 스케줄링을 수행할 수 있도록 해 준다. GDM은 분산 공유 메모리를 제공하는 Global Array를 이용해 이를 구현한다.

Global Array(GA)는 여러 노드에 존재하는 메모리를 하나의 공유 메모리처럼 사용할 수 있게 하는 분산 공유 메모리 툴킷이다. 배열형태의 데이터를 GA영역에 저장하면, GA가 제공하는 Get()함수를 사용하여 로컬 메모리 영역으로 가져올 수 있다[4].

GDM은 GA의 상위에서 렌더링 엔진을 위한 데이터를 가공한다. 그러나, GA는 오직 배열 형태의 데이터만 지원하기 때문에, 데이터를 적절히 구조화한 배열 형태로 변환할 필요가 있다. GDM은 이들 데이터의 변환과 변환에 따른 메타 데이터 정보를 유지 관리하며, 필요한 때 적절한 데이터를 GA로 요청하는 기능을 수행한다. 또한, 응용의 데이터 요청 패턴에 맞는 heuristic cache 알고리즘을 수행한다.

GDM에 하나의 dataset을 올리면, GDM은 각 노드에 데이터를 분산한다. GLOR

E가 각 노드에 병렬 처리 작업을 스케줄링하여 요청하면, 각 노드에 존재하는 GDM이 그 데이터가 자신의 노드에 있지 않더라도 GA 영역에서 찾아 작업을 요청한 노드의 로컬 메모리에 복사한다. 따라서, GLORE의 스케줄러는 데이터의 존재 유무에 관계없이 작업을 수행하지 않고 있는 유휴 노드에 작업을 분배할 수 있어 공정한 load balancing을 수행할 수 있다. 그림 7은 GLORE와 GDM이 이를 어떻게 처리하는지 보여준다.



[그림 8] GDM 데이터 처리 동작

7. 로터 시뮬레이션 데이터 가시화

GLOVE는 로터시뮬레이션 데이터를 가시화함으로써, 1차적인 프레임 워크의 전반적인 과정을 수행 검증하였다. 로터 데이터 시뮬레이션 데이터는 헬기의 이착륙 시 발생하는 기류의 흐름을 미리 시뮬레이션 함으로써, 소음, 진동, 로터의 변형 등 새로운 헬기 로터 설계 시 발생할 수 있는 상황을 예측할 수 있다. 로터

시뮬레이션 데이터의 주요 현상은 주로 헬기 로터 주변의 유동 데이터에서 관찰할 수 있다. 가시화 도구는 이러한 현상들을 인지할 수 있게 보여 주어야 한다. 다음 각 절에서는 로터 시뮬레이션 데이터를 위한 GLOVE의 기능들을 기술한다.

가. 대용량 데이터를 위한 로딩 기능

대용량 데이터를 다루는 가시화 도구에 있어 데이터의 로딩을 효율적으로 빠르게 수행하는 일은 매우 중요하다. 효율적인 데이터 로딩을 위해 GLOVE는 병렬로 데이터를 읽어 들여 이들을 각 노드들이 공유하는 형태를 목표로 한다. 병렬 IO와 함께 LambdaRAM과 같은Distributed Cache Memory 을 추가적으로 사용할 경우 Computation 속도는 20배로 향상될 것이 기대된다[1]. GLOVE 는 2.1에서 기술한 분산된 캐쉬 구조로 데이터를 읽고 공유하는 것을 목표로 2단계로 접근한다. 1단계는 master 노드 VRJuggler 응용이 데이터를 읽어 각 디스플레이 노드들에 네트워크를 통해 공유하는 방법으로 효율을 향상시킨다. 2단계로 분산 캐쉬 메모리인 LambdaRAM을 도입한다. 현재 GLOVE는 1단계로 구현되었다.

나. 블레이드 데이터 가시화 기능

로터 시뮬레이션의 결과로 얻어지는 블레이드 데이터는 블레이드 상에 압력 데이터와 후처리로 얻어진 분석 데이터들이 있다. GLOVE는이들 블레이드 압력 데이터의 가시화와 후처리 데이터를 위한 그래프 기능을 제공한다.

(1) Pressure Distribution on blades

GLOVE는 블레이드 데이터의 압력에 대한 데이터를 읽어 날개 표면에 압력 분포를 보여준다.

(2) Graph

- Pressure distribution graph

입력 받은 span position에서 회전각에 따른 pressure의 변화를 커브 형태의 그래프로 보여준다.

- Pitch angle variation graph

입력 받은 span position 과 0~360도까지 회전각에 대한 pitch angle 의 변화값을 커브와 디스크 형태의 그래프로 제공한다.

- Sectional Force

입력 받은 span position 과 0~360도 까지 회전각에 대한sectional normal force, sectional chord force, sectional span force 그래프를 커브와 디스크 형태로 제공한다.

- Sectional Moment

입력 받은 span position 과 0~360도 까지 회전각에 대한 sectional x-moment, sectional y-moment, sectional z-moment 그래프를 커브와 디스크 형태로 제공한다.

(3) Animation

시간경과에 따른 블레이드 상의 압력 분포 변화를 그래프의 time step과 동기화하여 animation으로 보여주는 기능을 제공한다.

다. 필드 데이터 가시화 기능

필드 데이터에는 유동에 관한 정보들이 존재한다. 따라서, 로터 데이터의 가시화에 있어 필드 데이터는 중요한 의미를 갖는다. GLOVE는 VR 환경의 3차원 가시화를 지원하므로, 이들 데이터를 가시화 함으로써, 블레이드와 필드 데이터들 사이의 상호작용을 가시적으로 좀더 명확하게 확인할 수 있다. 이를 위해, GLOVE는 다음의 기능들을 제공한다.

(1) Vorticity Iso-surface

동일한 vorticity 크기 값을 가지는 부분에서 iso-surface를 그려 준다. 그리고자 하는 vorticity의 크기 값을 입력 받아 그릴 수 있다.

(3) Q-Criteria

Votex 영역을 Q-criteria 방법을 사용하여 보여준다.

(4) Velocity Streamline

관심 있는 영역을 지정하거나 vorticity 값이 특정 값 이상인 부분에서 velocity streamline 을 생성하여 보여 줄 수 있다.

(5) Velocity Iso-surface

동일한 velocity 크기 값을 가지는 부분에서 iso-surface를 그려준다. 사용자가 그리고자 하는 velocity 크기 값을 지정할 수 있다.

(6) Animation

블레이드 데이터와 vorticity, velocity 에 대한 데이터를 가시화하여 블레이드 회

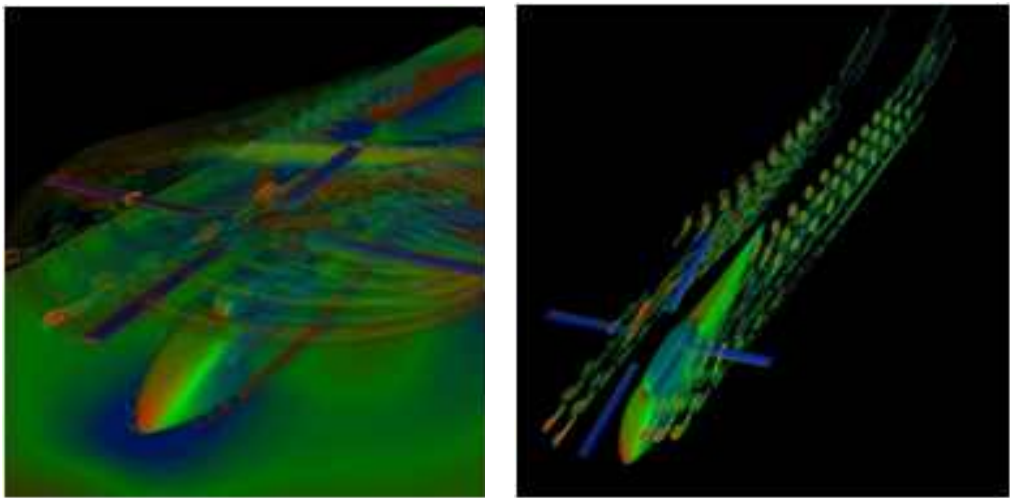
전 시 vorticity와 velocity 변화를 관찰할 수 있다.

(7) Contour line

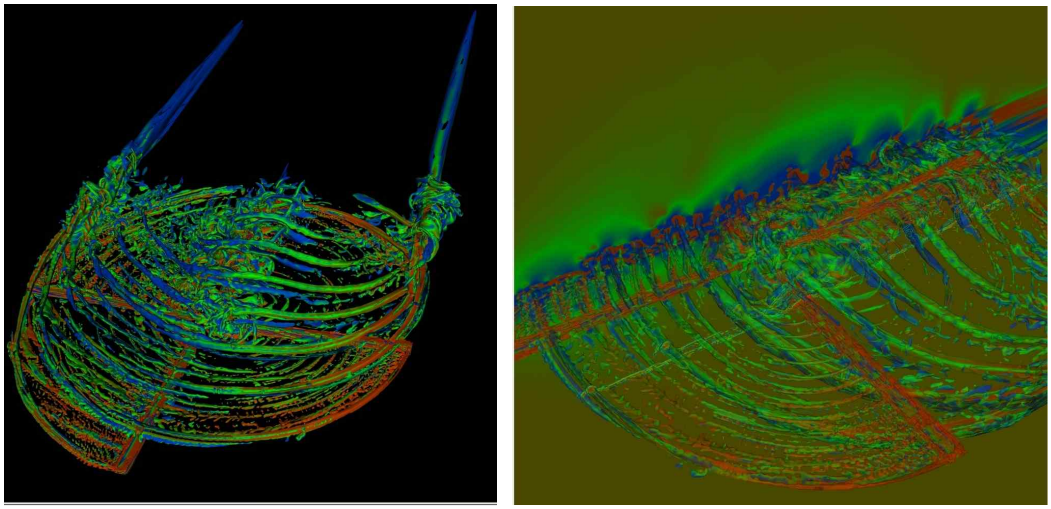
동일한 값을 가지는 영역을 선을 연결하여 보여 준다. vorticity, velocity 각각에 대해 가능하다.

라. 로터 시뮬레이션 데이터 가시화 결과

그림 5는 동체를 포함한 로터 시뮬레이션 데이터를 가시화한 결과이다. 왼쪽 그림은 동체와 로터 블레이드의 데이터의 압력 분포를 가시화 한 다음 vorticity iso-surface를 그리고, probe by plane으로 하나의 평면에 필드 데이터의 압력 분포를 그린 결과이다. 오른쪽 그림은 동체와 블레이드의 표면 압력분포를 그린 다음 probe by plane 으로 세 개의 평면을 만든 다음, 각각에 대해 contour line을 그린 결과이다.



[그림 9] 동체 데이터 가시화 결과



[그림 10] 1.21 테라바이트 로터 시뮬레이션 데이터 가시화 결과

그림 9는 1.21TB의 데이터를 가시화한 결과이다. 모두 90개의 노드에 데이터를 로드하고, 병렬처리를 수행하였다. 렌더링 노드 당 메인 메모리는 64GB 이며, CPU Clock은 3.0GHz이다. 왼쪽 그림은 유동 데이터의 압력 데이터를 iso-surface 로 가시화 하였으며, 22개의 특정 값에 해당하는 데이터를 추출해 내는데 걸린 시간은 1.55초(하나의 노드가 수행할 경우 20.24초)이다.

표 2와 표 3은 로터 시뮬레이션 데이터의 성능 측정 결과를 보여 준다. 6GB 보다는 1.2TB 즉 대용량 데이터일수록 GLOVE의 성능이 COVISE나 Paraview를 앞선다. 특히 COVISE는 병렬렌더링을 지원하지 않으므로, 사용자의 응답시간은 1.2TB iso-surface 의 경우, 병렬 렌더링을 수행하는 GLOVE에 비해 20배이상 느리다. Paraview는 싱글노드 처리에 있어서는 성능이 우수하나 병렬처리를 수행할 경우, 1.2TB iso-surface를 처리하는 데 있어 5배 가량 GLOVE이 성능이 우수한 것으로 측정 되었다.

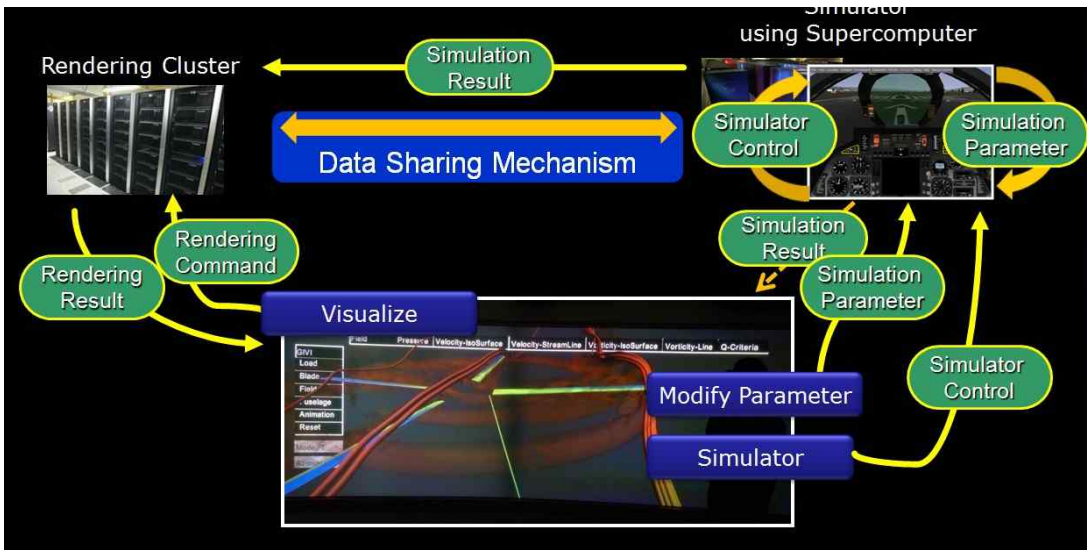
[표 2] 6GB, 30 time step 로터 데이터 성능 측정

Tool	Parallel	COVISE	Paraview	GLOVE	COVISE:Paraview:GLOVE
Probe by Plane (sec)	single	11	15.22	1.23	8.8 : 12.37 : 1
	multi	N/A	0.09	0.15	∞ : 0.6 : 1
iso-surface (sec)	single	13	23.47	1.68	7.7 : 13.97 : 1
	multi	N/A	0.09	0.30	∞ : 0.3 : 1
iso-surface animation(sec)	single	N/A	219.00	1.68	∞ : 129.6 : 1
	multi	N/A	1.27	0.26	∞ : 4.88 : 1

[표 3] 1.2TB, 90 time step 로터 시뮬레이션 데이터 성능 측정

Tool	Parallel	COVISE	Paraview	GLOVE	COVISE:Paraview:GLOVE
Probe by Plane (sec)	single	36	11.93	20.04	1.8 : 0.59 : 1
	multi	N/A	1.79	1.80	∞ : 0.99 : 1
iso-surface (sec)	single	47	14.24	18.21	2.58: 0.78 : 1
	multi	N/A	10.11	2.12	∞ : 4.76 : 1
iso-surface animation(sec)	multi	N/A	274	2.12	∞ :129.2 : 1

8. 결론



GLOVE는 고성능 컴퓨팅 환경에서 대용량 시뮬레이션 데이터를 효과적으로 분석하고 가시화하여 공유할 수 있는 가시화 도구이다. 고해상도의 tiled display와 VR환경은 응용 연구자에게 데이터 분석에 있어 직관적으로 다른 시각의 관점을 제공한다. 또한 다계층의 데이터 구조는 응용이 달라지더라도 최소한의 수정으로 응용에 맞는 하나의 새로운 인터페이스를 제공할 수 있다. 무엇보다 기존의 가시화에 비해 5배 가량의 성능이 향상되었음을 보여 주었다.

그러나, 현재의 가시화 기능은 VTK라는 가시화 툴킷의 기능에 종속적이다. 따라서, 대용량 데이터를 위한 보다 다양한 가시화 기능을 제공하기 위해, 렌더링 엔진인 GLORE의 추가 구현이 필요하다.

궁극적으로 GLOVE는 슈퍼컴퓨터와의 연계를 통해 시뮬레이션 과정에서 생성되는 데이터를 실시간으로 가시화함으로써, 시뮬레이션의 결과를 즉각적으로 판단해 시뮬레이션을 제어할 수 있는 통합 가시화 시뮬레이션 제어 도구를 목표로

확장해 갈 것이다.

9. 참고문헌

[1] Andy Cedilnik, Berk Geveci, "Remote Large Data Visualization in ParaVie w Framework", Eurographics Symposium on Parallel Graphics and Visualizatio n, 2006

[2] COVISE, http://www.hlrs.de/covise

[3] S.Byron, "Virtual Reality in Scientific Visualization", Communications of t he ACM, 39(5):62-71, 1996.

[4] A.van Dam, A.S.Forsberg, D.H.Laidlaw, J.J.Laviola, R.M.Simpson, "Immer sive VR for Scientific Visualization: A Progress Report", IEEE Computer Gra phics and Applications, 2000.

[5] ParaView, http://paraview.org/New/index.html

[6] VTK, http://www.vtk.org

[5] COVISE, http://www.hlrs.de/organization/av/vis /covise

[7] A.Bierbaum, C.Just, P.Hartling, K.Meinert, A.Baker, C.Cruz-Neira, "VRJu ggler: A Virtual Platform for Virtual Reality Application Development", Proce edings of IEEE VIrtual Reality, 2001.

[8] VRJuggler, http://www.vrjuggler.org

[9] CaveLib, http://www.mechdyne.com/integrated Solutions/software/products /CAVELib/CAVELib.html

[10] M.Schirki, A.Germdt, T.van Reimersdahl, T.Kuhlen, P.Adorneit, O.Lang, S.Pischinger, C.Bischof, "Vista FlowLib - A Framework for Interactive Visua lization and Exploration of Unsteady Flows in Virtual Environments", Eurogra phics Workshop on Virtual Environments, 2003.

[11] Andrew Forsberg, Prabhat, Graff Haley, Andrew Bragdon, Joseph Levy, Caleb I.Fassett, David Shean, James W.Head III, Sarah Milkovich, Mark Duch aineau, "Adviser: Immersive Field Work for Planetary Geoscientists", IEEE C omputer Graphics and Applications, July 2006.