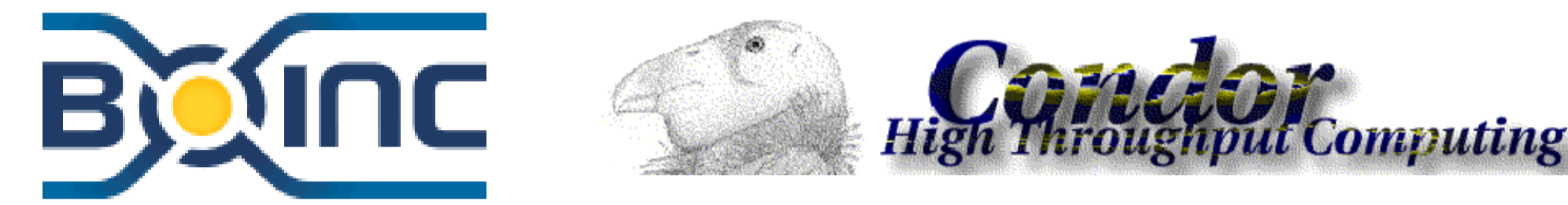


# HTCaaS: A Large-Scale High-Throughput Computing by Leveraging Grids, Supercomputers and Cloud

Seungwoo Rho, Seoyoung Kim, Sangwan Kim, Seokkyoo Kim, Jik-Soo Kim and Soonwook Hwang  
Supercomputing Center at the Korea Institute of Science and Technology Information (KISTI)

## High-Throughput Computing



- Many loosely coupled tasks requiring a large amount of computing power
- Independent, sequential jobs that can be scheduled on many different computing resources
- Growing in the number of jobs and complexity drives HTC area into *Many-Task Computing (MTC)*

## Harnessing as many computing resources as possible is inevitable

- Grids, Supercomputers and Clouds are available to the scientific community

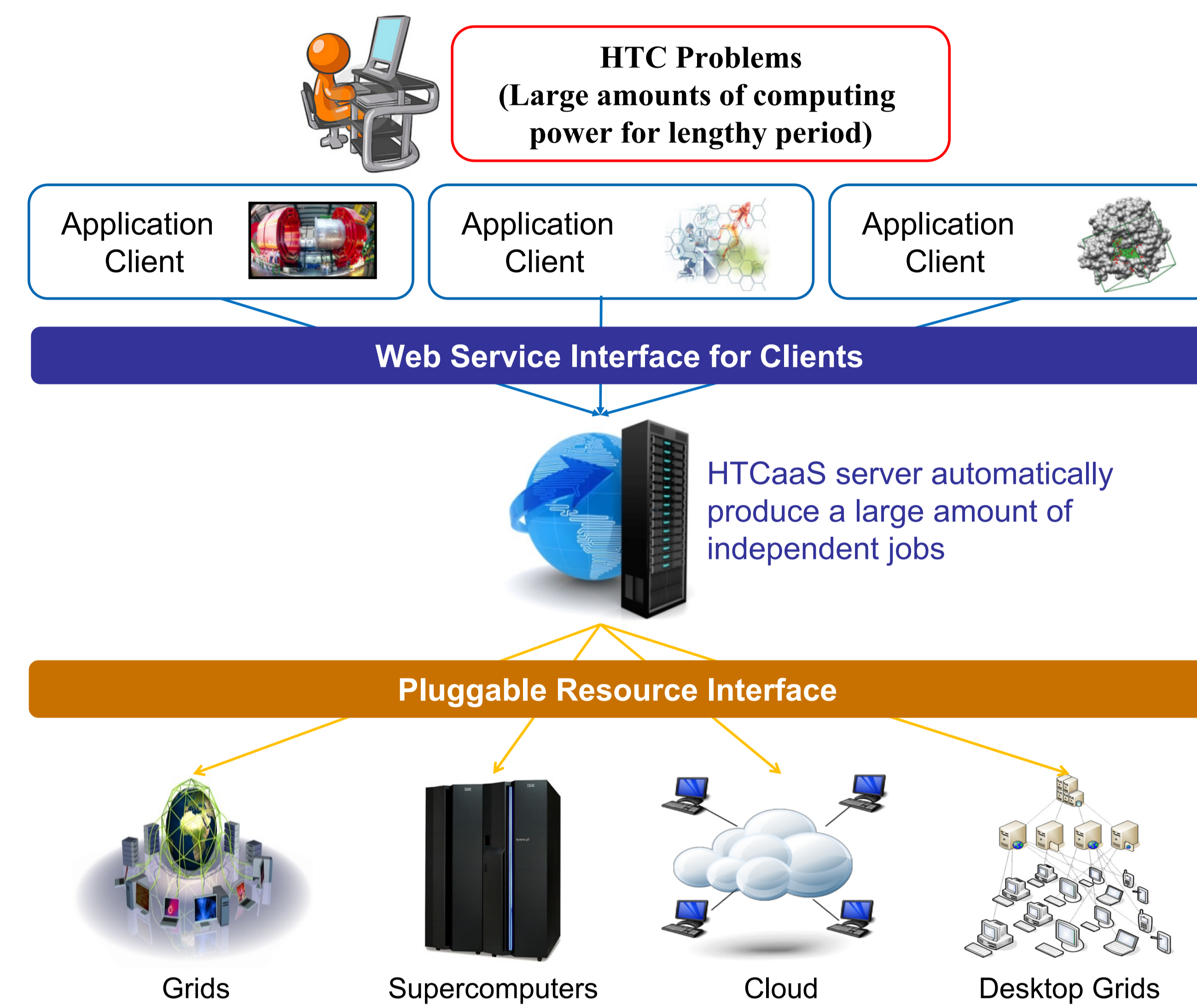


## Hard Problems / Issues

- Hiding heterogeneity and complexity of leveraging different computing resources from users
- Efficiently submitting a large number of jobs at once and managing them
- Effective management and exploitation of all available computing resources

## Our Approach

- *High-Throughput Computing As a Service (HTCaaS)* for scientific computing
- *Meta-Job* based automatic job split & submission (e.g., parameter sweeps)
- User-level job scheduling
- Pluggable interface to heterogeneous computing resources
- Application independent



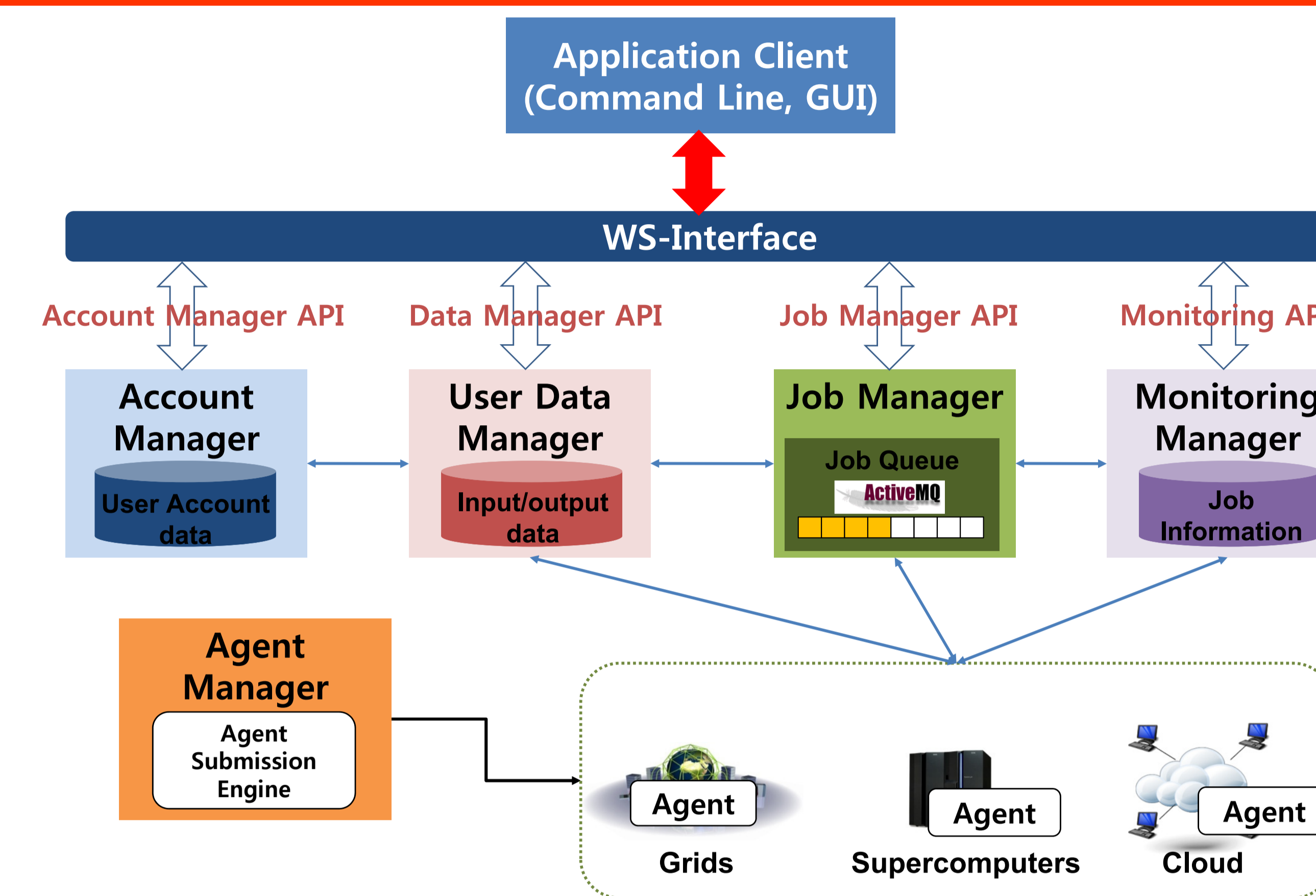
[Figure 1: HTCaaS Concept. Bridging the various HTC applications and heterogeneous computing resources]

## Design Philosophy

- *Ease of Use*: minimize user overhead for handling a large amount of jobs & resources
- *Intelligent Resource Selection*: automatic selection of more responsive and effective resources
- *Pluggable Interface to Resources*: adopt GANGA's plugin mechanism for accessing different resources without hardcoding
- *Support for Many Client Interfaces*: a wide range of client interfaces are supported including a native WS-interface, Java API, CLI and GUI

## System Architecture

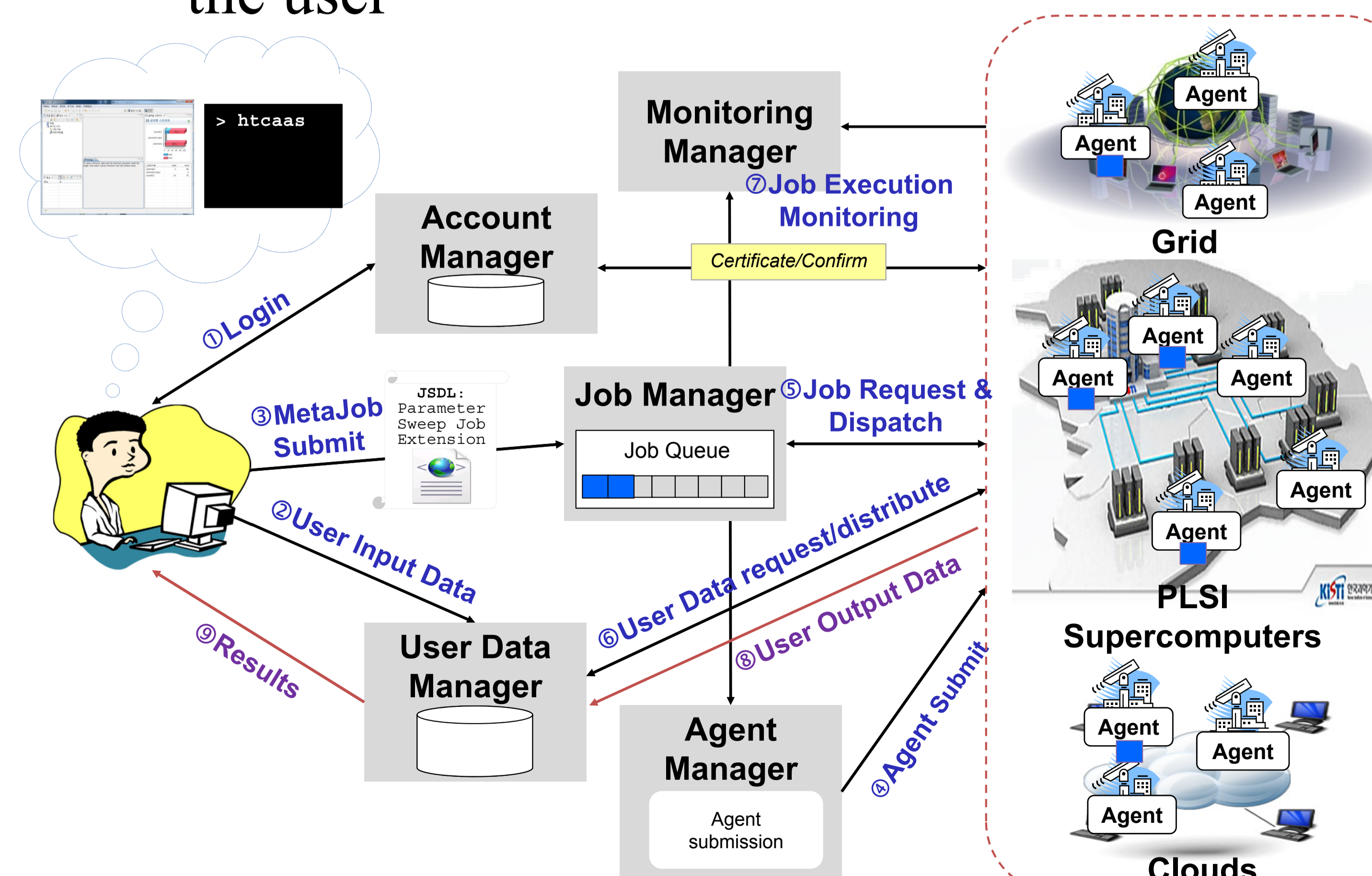
- Powerful Meta-job description based on the OGF JSDL Parameter Sweep Extension Specification
- Agent-based multi-level scheduling & streamlined job dispatching
- Service-Oriented Architecture (SOA) based on WS-Interface



[Figure 2: HTCaaS System Architecture. Jobs & input/output data are managed by Job Manager and User Data Manager. Agents are dispatched from Agent Manager and process jobs in Grids, Supercomputers and Clouds]

## Job Submission & Execution Steps

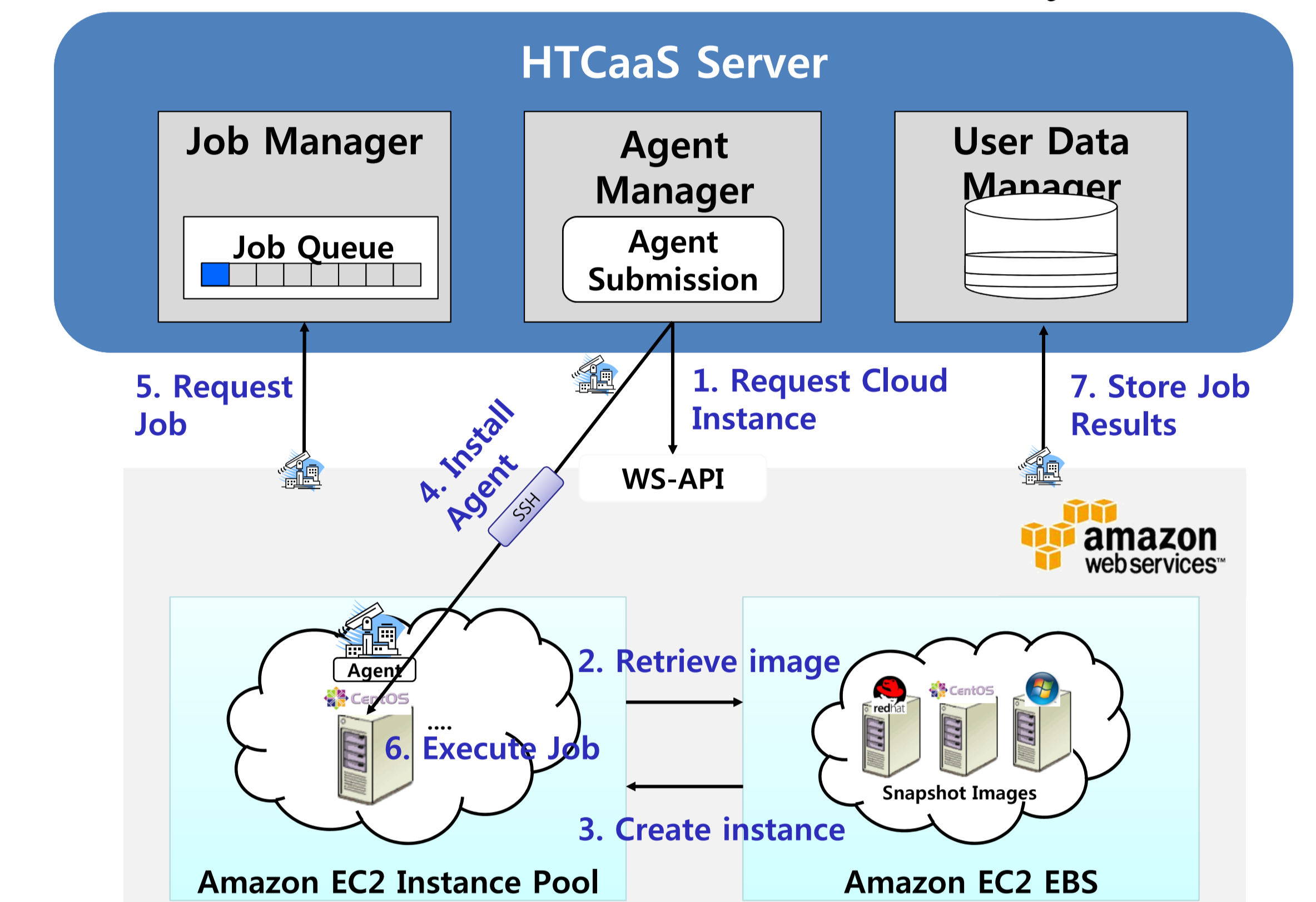
- 1) User logs in HTCaaS and uploads input data through User Data Manager
- 2) User submits a Meta-Job which can be composed of multiple tasks
- 3) HTCaaS *automatically* divides a Meta-Job into multiple tasks based on the specification
- 4) Agent Manager dispatches agents based on job requirements and resource availability
- 5) Agents proactively request tasks and process
- 6) Finished results are stored and notified to the user



[Figure 3: Job Submission & Execution Steps in HTCaaS]

## Utilizing Cloud for Dynamic & On-demand Resource Provisioning

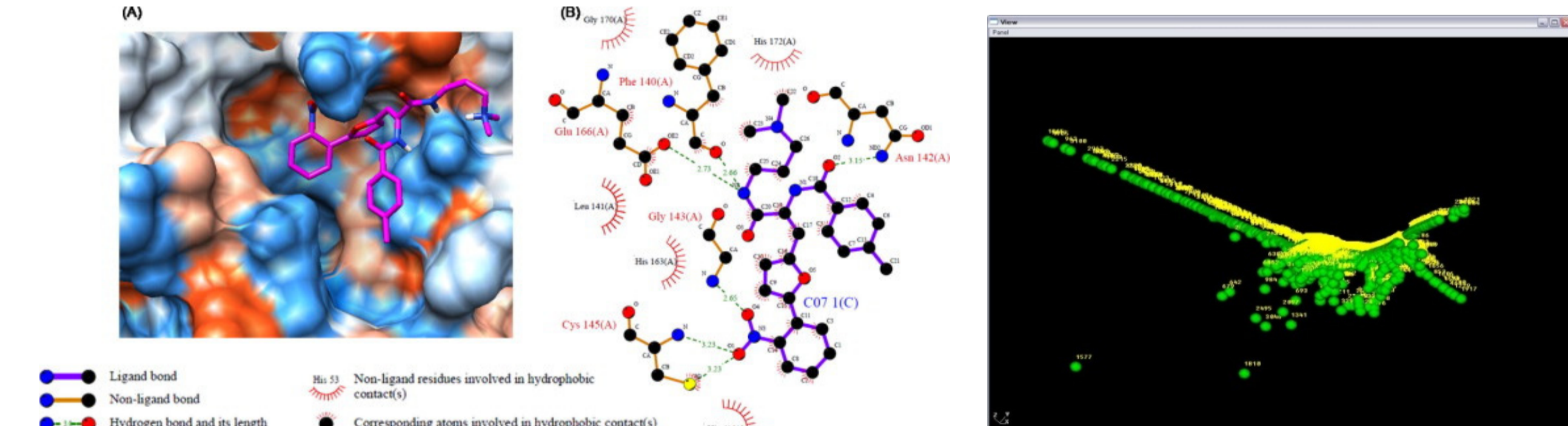
- Constructing a hybrid scientific cloud infrastructure for HTC on the fly



[Figure 4: HTCaaS in the Cloud (Amazon EC2)]

## Applications Support

- Virtual Screening (Docking)
  - A target protein of 3CL-pro of SARS with 1.1 Million chemical compounds
- 3D Visualization of Optimized Design Solution
  - On average 500 CPU utilized, Completed in 2.8 days, totally 2.6 years of computation



[Figure 5: HTCaaS Target Applications]

## Current Status & Future Work

- Completed testing of integration with Grids, and Cloud
- Seamless integration with PLSI (Partnership & Leadership for the nationwide Supercomputing Infrastructure) in Korea
- System Scalability & Fault tolerance
- Improving User-level Job Scheduling