Protein Multiverse on University HPC Grid

Azhar Ali Shah

Protein Multiverse Meeting, Sep 26, 2007

The University of Nottingham

world-changing research
from The University of Nottingham

Outline

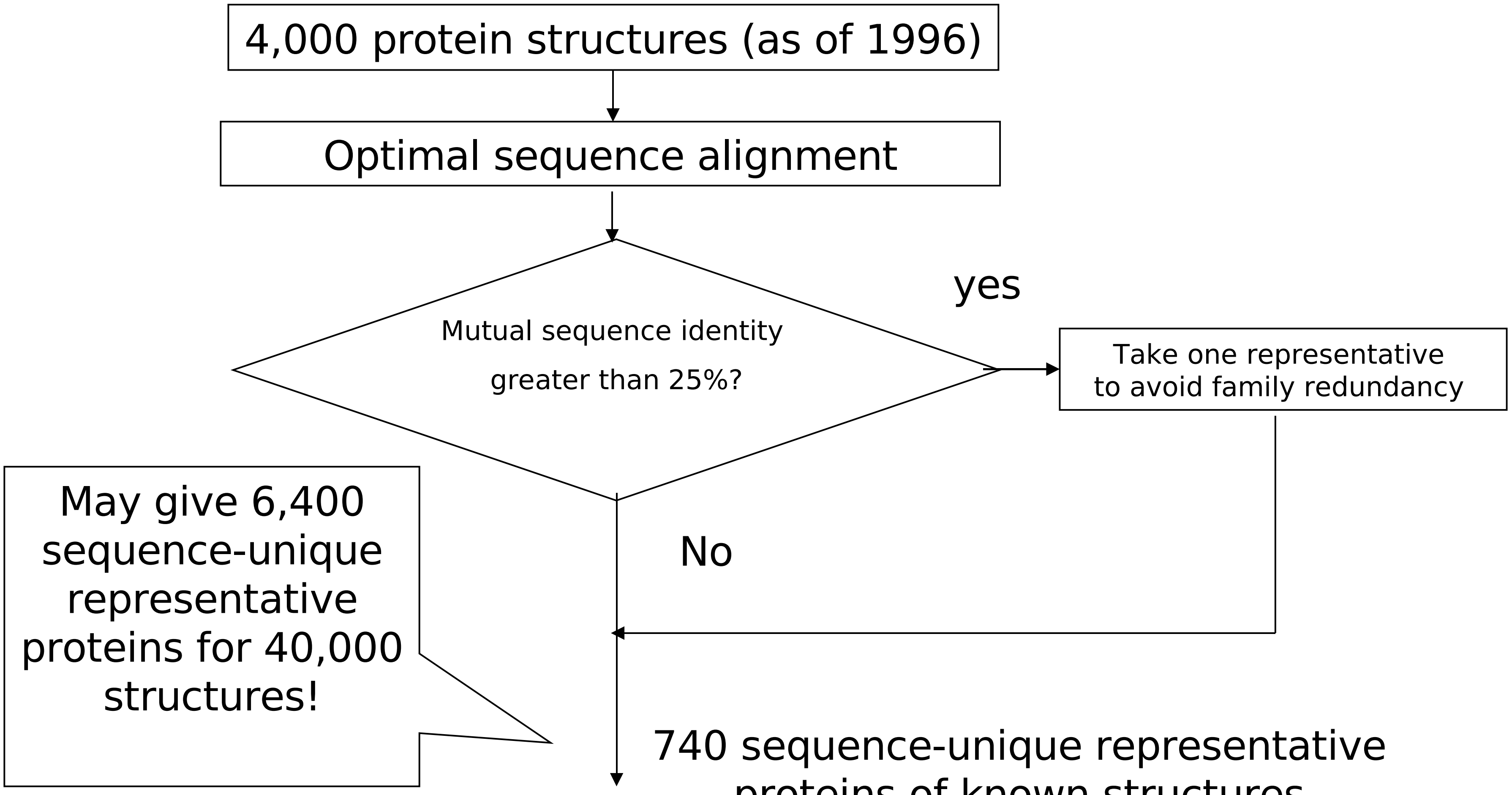
- **Related Work**
- **Complexity of the Problem**
- **Architectural Design**
- **Program Workflow Design (PWD)**
- **Infrastructure Details**
- **Discussion**

Related Work1

- Mapping the Protein Universe (Holm, Sander 1996)

  - **Motivations for all-on-all comparison:**
    - Distribution of known structures in shape space
    - Grand view of the architecture of all proteins
    - A map of physical attractor regions in the abstract shape space of proteins
      - Help to understand protein folding and evolution

Database Preparation



4,000 protein structures (as of 1996)

Optimal sequence alignment

Mutual sequence identity greater than 25%?

yes

Take one representative to avoid family redundancy

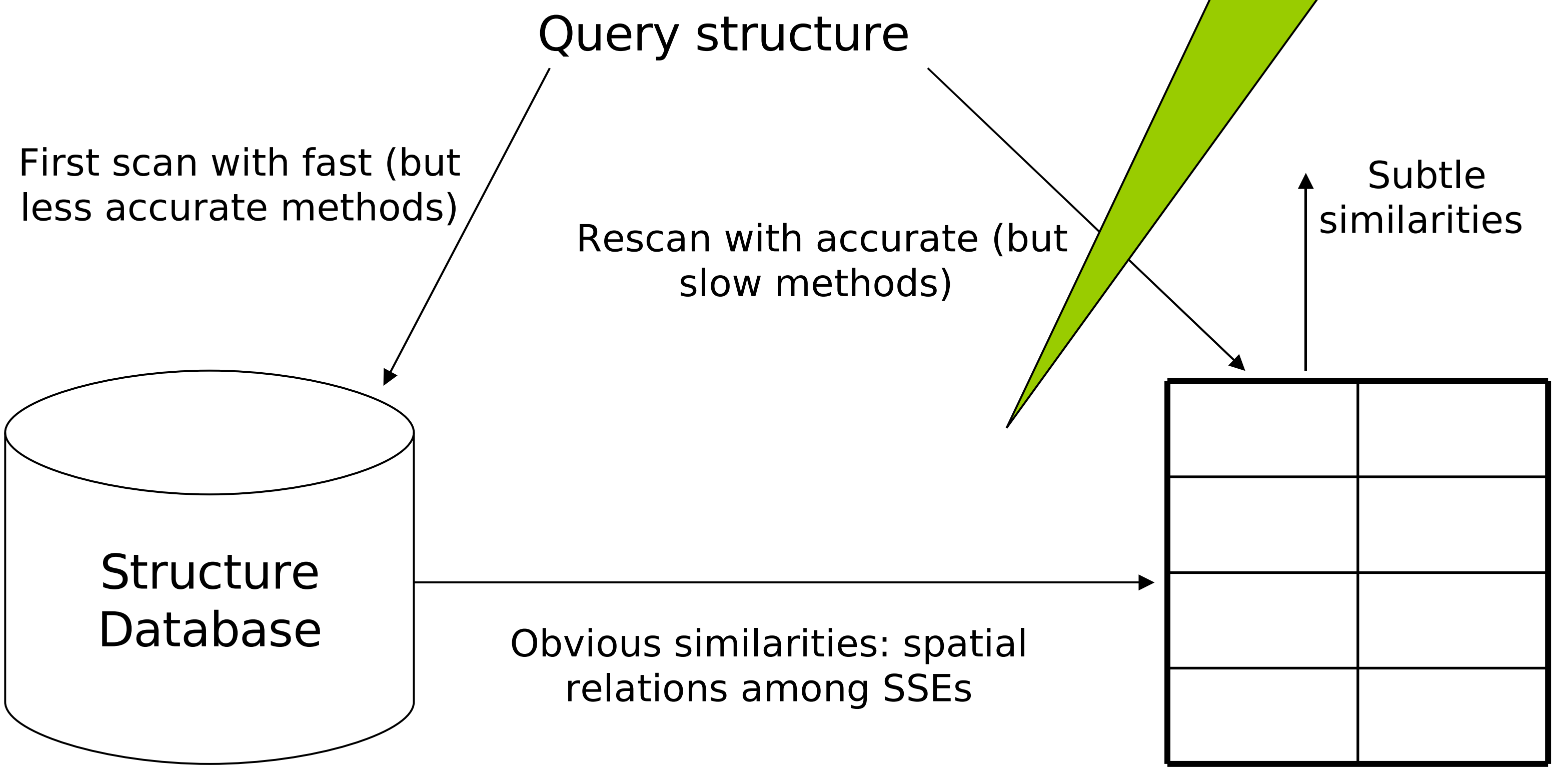
No

May give 6,400 sequence-unique representative proteins for 40,000 structures!

740 sequence-unique representative proteins of known structures

Searching 3D Datab

No parallelization


Query structure
First scan with fast (but less accurate methods)
Rescan with accurate (but slow methods)
Subtle similarities
Structure Database
Obvious similarities: spatial relations among SSEs

One structure against several thousand structures takes 5 minutes on a normal workstation

Classification

On a high-dimensional fold space:

Families: Close range clusters
Folds: Intermediate range clusters
Attractors: Long range clusters

- Domains:
  - Structures having same recurring substructures are grouped into *Domains*
    - 1048 domains for 740 structures
- Fold classes:
  - Similar domains are grouped into *fold* classes
    - 287 folds for 1048 domains
  - Fold class is based on structural similarity and is analogous to ***Family*** which is based on sequence similarity.
- Attractors:
  - Five long regions in an abstract high-dimensional ***fold space***

Related Work2

- Global mapping of protein ***structure space*** and application in structure-based inference of protein function (Hou et al. 2005), PNAS.
  - Problem:
    - Simple structure comparison does not provide function inference for a protein with new fold
  - Solution:
    - A method based on map distance of protein structure space

Kim says. "This map provides us with a conceptual framework to organize ***all protein structures*** and functions and have that information readily available in one place", Berkley Lab Research News, Feb 2003.

Database preparation

- PDB_SELECT 25 DATASET (Rel. Dec 2002)
  - A representative subset of the PDB containing ***1,949*** chains having <25% sequence identity
    - ***51*** chains further removed because of low resolution or length requirement of DaliLite
  - The resultant dataset consisted ***1,898*** protein chains

Mapping the pro re space 1/3

IBM SP RS/6000

- The pairwise structural similarity of 1,898 chains was measured with DaliLite (**25,000 cpu hours)**
- The 1898x1898 similarity matrix [s_{ij}] was converted to dissimilarity matrix [d_{ij}] using:

$$dij = \begin{cases} s_{99.95} - s_{ij}\left(s_{99.95} > s_{ij}, i \neq j\right) \\ 0,\left(i = j\right) \\ s_{99.95,\left(otherwise\right)} \end{cases}$$

This matrix was used for structure space map (SSM)

Where $s_{99.95}$ is the 99.95th percentile of the distribution of all off-diagonal values of s_{ij}

Mapping the protein structure space 2/3

- Four scores:
  - Structure Space Map (SSM) distance score
  - DaliLite similarity score
  - DaliLite Z-score
  - BLAST-E values of pairwise sequence alignment

- ROC plots for evaluation
- Comparison of function inference among different scores based on GO function families
  - SSM outperforms other scores!

Mapping the protein structure space 3/3

- Based on present results it is predicted that the conceptual map of ***all protein structures*** would have same essential features.

Let us test this hypothesis!

Multi-method 3-D Map of Protein Structure Universe

- Related work
- **Complexity of the Problem**
- Architectural Design
- Program Workflow Design (PWD)
- Infrastructure Details
- Discussion

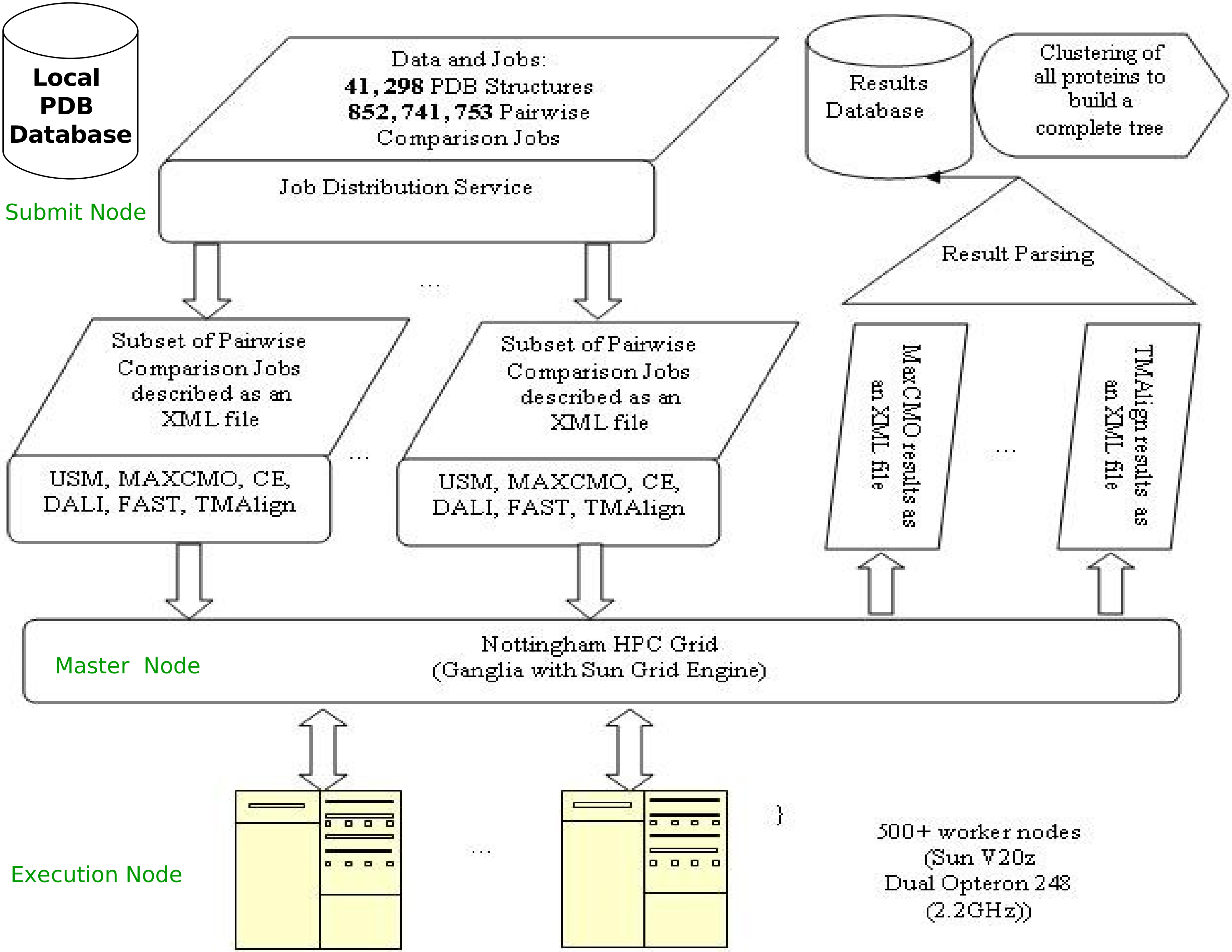
Complexity of the problem

- Job complexity

$$N_j = \frac{n(n-1)}{2} = \frac{41298x41297}{2} = 852741753$$

- Computational time
  - 4088 hours => 170 days
- Storage complexity
  - It takes 21 hours to download the PDB database with 41,298 structures which requires the space of 35 GB
  - RAM would be the main obstacle for XML based input/output files

- Related work
- Complexity of the Problem
- **Architectural Design**
- Program Workflow Design (PWD)
- Infrastructure Details
- Discussion


Local PDB Database
Data and Jobs:
41,298 PDB Structures
852,741,753 Pairwise Comparison Jobs
Job Distribution Service
Results Database
Clustering of all proteins to build a complete tree
Result Parsing
...
Subset of Pairwise Comparison Jobs described as an XML file
USM, MAXCMO, CE, DALI, FAST, TMAlign
...
Subset of Pairwise Comparison Jobs described as an XML file
USM, MAXCMO, CE, DALI, FAST, TMAlign
MaxCMO results as an XML file
...
TMAlign results as an XML file
Nottingham HPC Grid
(Ganglia with Sun Grid Engine)
...
}
500+ worker nodes
(Sun V20z
Dual Opteron 248
(2.2GHz))

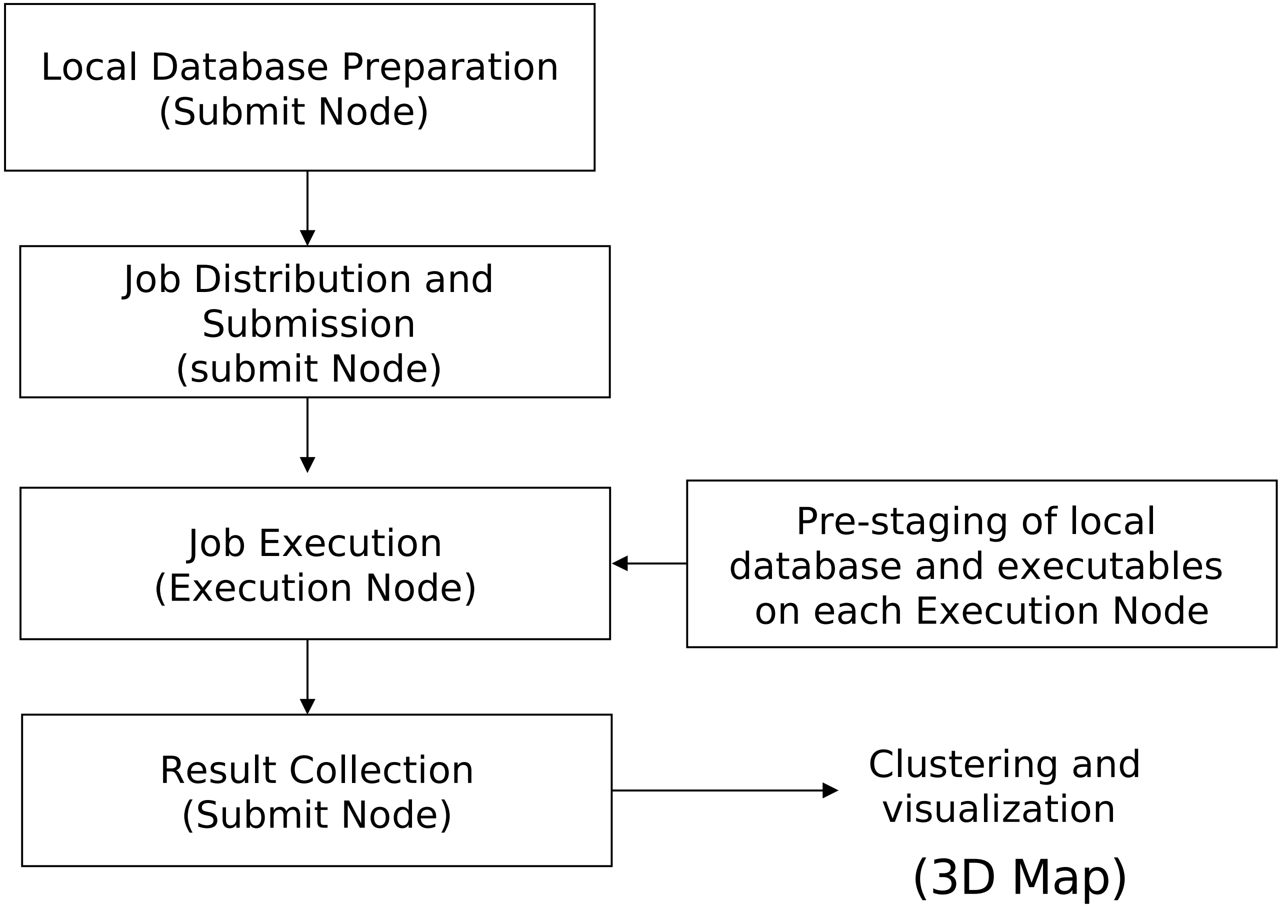
Submit Node

Master Node

Execution Node

- Related work
- Complexity of the Problem
- Architectural Design
- **Program Workflow Design (PWD)**
- Infrastructure Details
- Discussion

PWD: Main Tasks



Local Database Preparation
(Submit Node)

Job Distribution and
Submission
(submit Node)

Job Execution
(Execution Node)

Pre-staging of local
database and executables
on each Execution Node

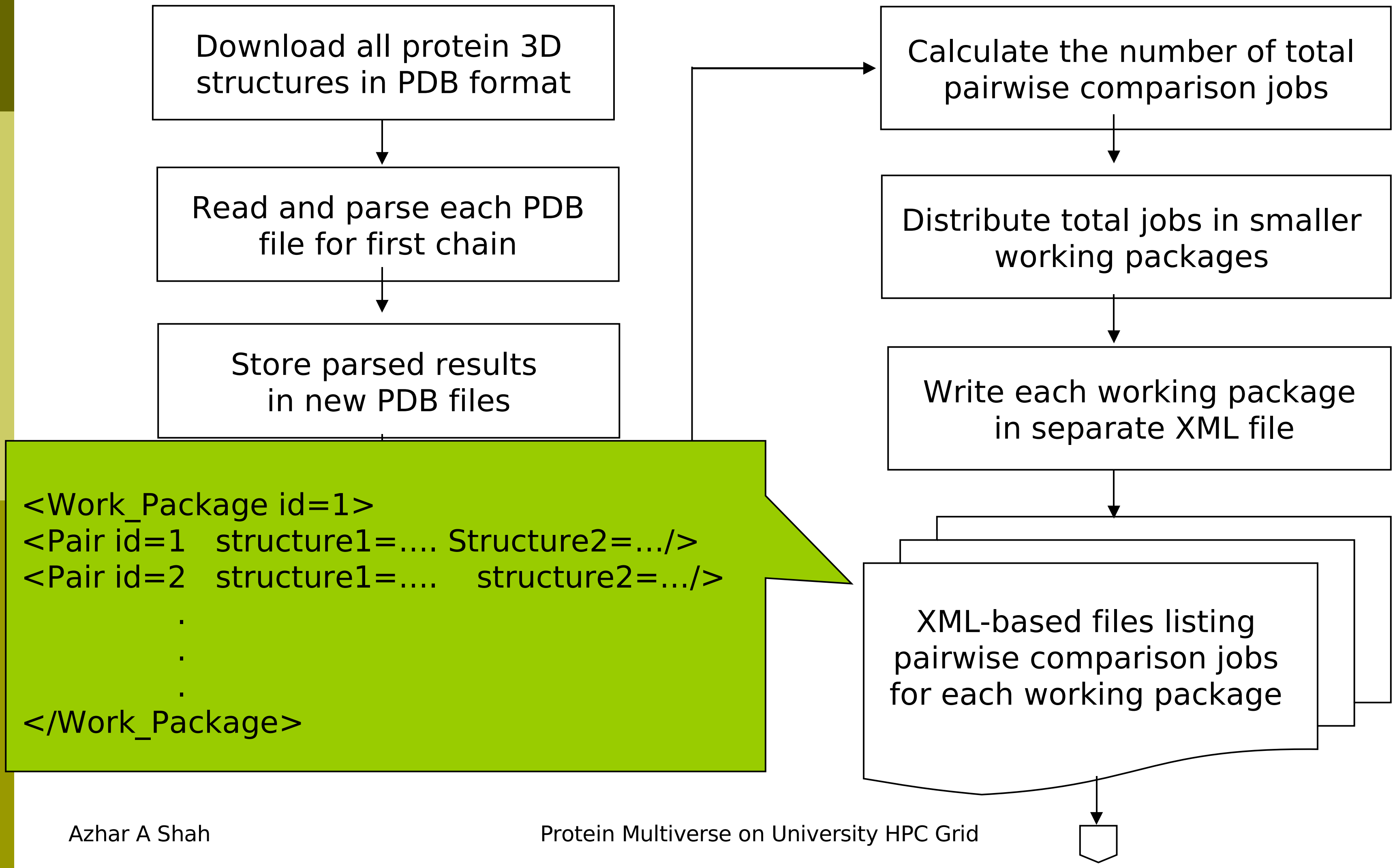
Result Collection
(Submit Node)

Clustering and
visualization

(3D Map)

Local Database Preparation

Job Distribution



Download all protein 3D structures in PDB format

Read and parse each PDB file for first chain

Store parsed results in new PDB files

Calculate the number of total pairwise comparison jobs

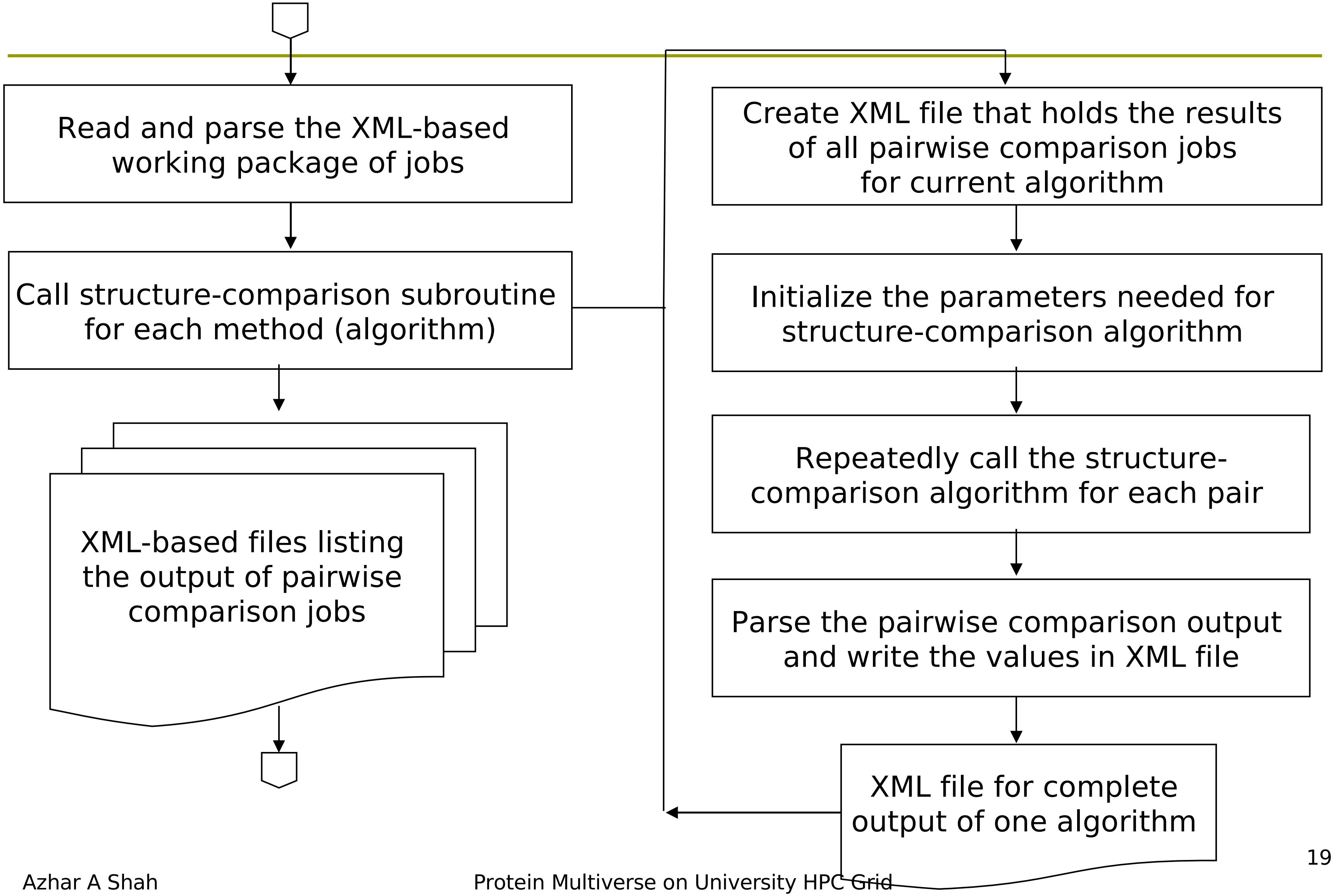
Distribute total jobs in smaller working packages

Write each working package in separate XML file

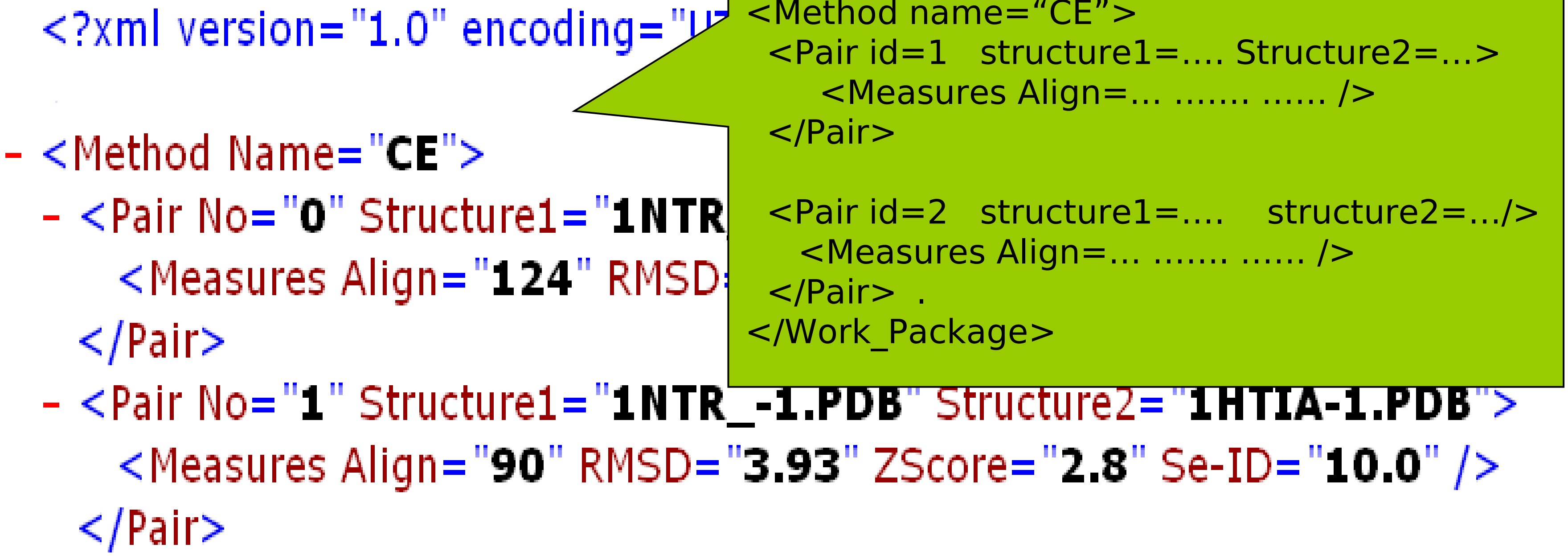
XML-based files listing pairwise comparison jobs for each working package

```
<Work_Package id=1>
<Pair id=1   structure1=.... Structure2=.../>
<Pair id=2   structure1=....    structure2=.../>
             .
             .
             .
</Work_Package>
```

Job Execution


Read and parse the XML-based working package of jobs
Call structure-comparison subroutine for each method (algorithm)
XML-based files listing the output of pairwise comparison jobs
Create XML file that holds the results of all pairwise comparison jobs for current algorithm
Initialize the parameters needed for structure-comparison algorithm
Repeatedly call the structure-comparison algorithm for each pair
Parse the pairwise comparison output and write the values in XML file
XML file for complete output of one algorithm

Typical output XML file



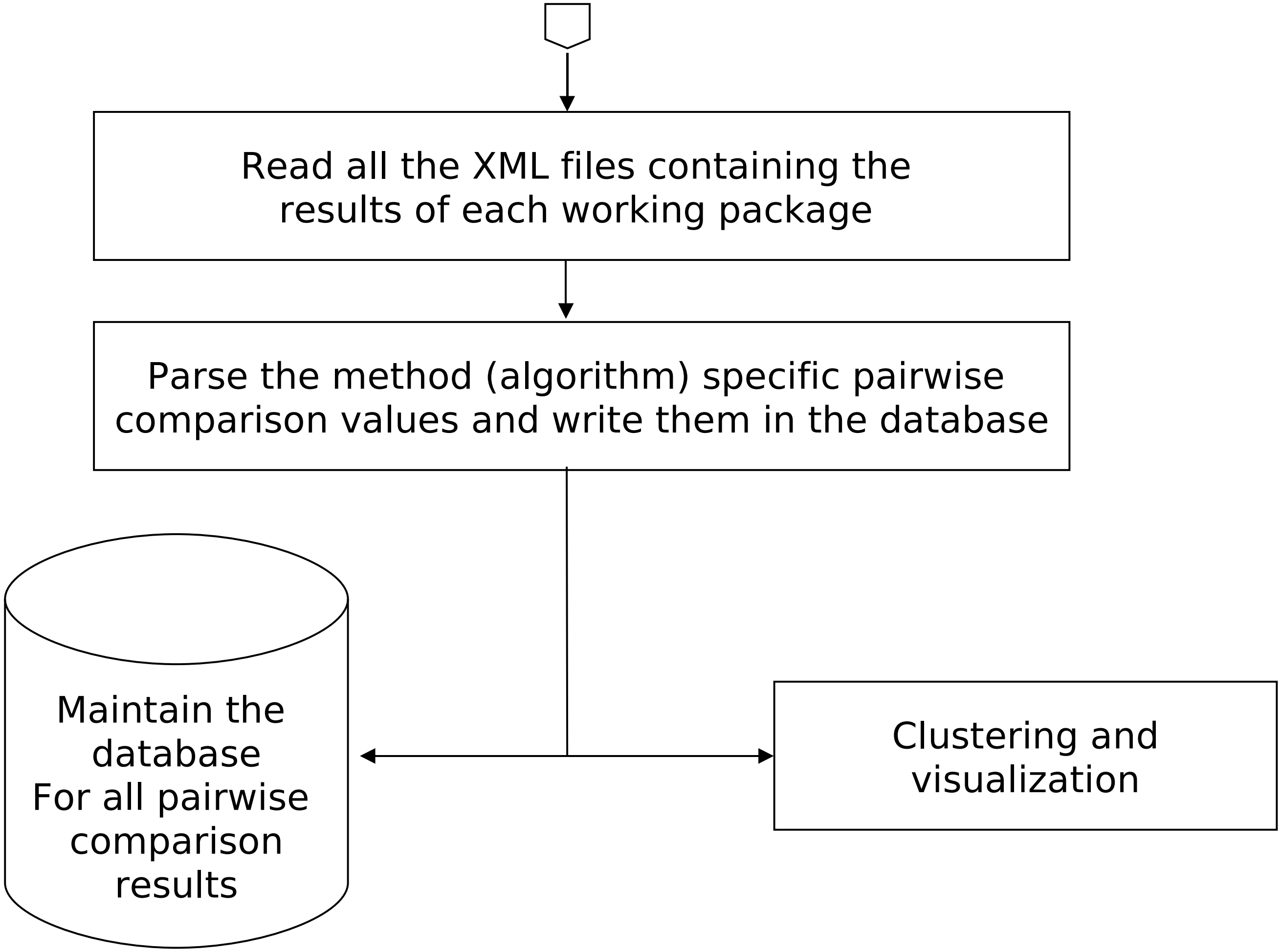
```
<?xml version="1.0" encoding="U

- <Method Name="CE">
  - <Pair No="0" Structure1="1NTR
      <Measures Align="124" RMSD
    </Pair>
  - <Pair No="1" Structure1="1NTR_-1.PDB" Structure2="1HTIA-1.PDB">
      <Measures Align="90" RMSD="3.93" ZScore="2.8" Se-ID="10.0" />
    </Pair>
```

```
<Work_Package id=1>
<Method name="CE">
  <Pair id=1   structure1=…. Structure2=…>
       <Measures Align=… ……. …… />
  </Pair>

  <Pair id=2   structure1=….    structure2=…/>
     <Measures Align=… ……. …… />
  </Pair>  .
</Work_Package>
```

Result Collection



Read all the XML files containing the results of each working package

Parse the method (algorithm) specific pairwise comparison values and write them in the database

Maintain the database For all pairwise comparison results

Clustering and visualization

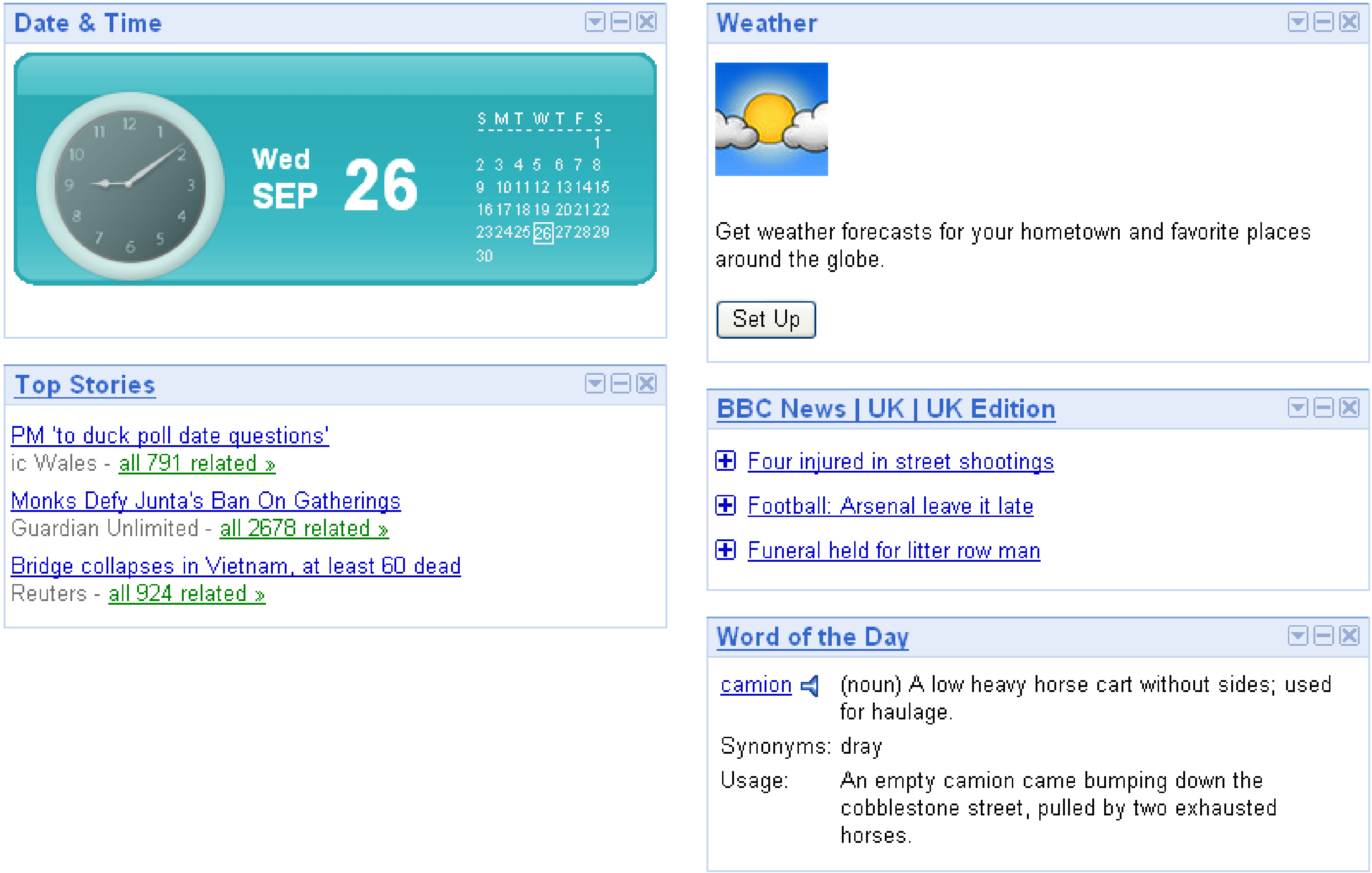
Results Database Schema

CE			
Pair_Lablel	RMSD	Z-Score	…
Str1:Str2	123	123	…
…	…	…	…

- Related work
- Complexity of the Problem
- Architectural Design
- Program Workflow Design (PWD)
- **Infrastructure Details**
- Discussion

Consistent look and feel for multiple applications/services

User Interface: Example



Top Stories

PM 'to duck poll date questions'
ic Wales - all 791 related »

Monks Defy Junta's Ban On Gatherings
Guardian Unlimited - all 2678 related »

Bridge collapses in Vietnam, at least 60 dead
Reuters - all 924 related »

Weather

Get weather forecasts for your hometown and favorite places around the globe.

BBC News | UK | UK Edition

- Four injured in street shootings
- Football: Arsenal leave it late
- Funeral held for litter row man

Word of the Day

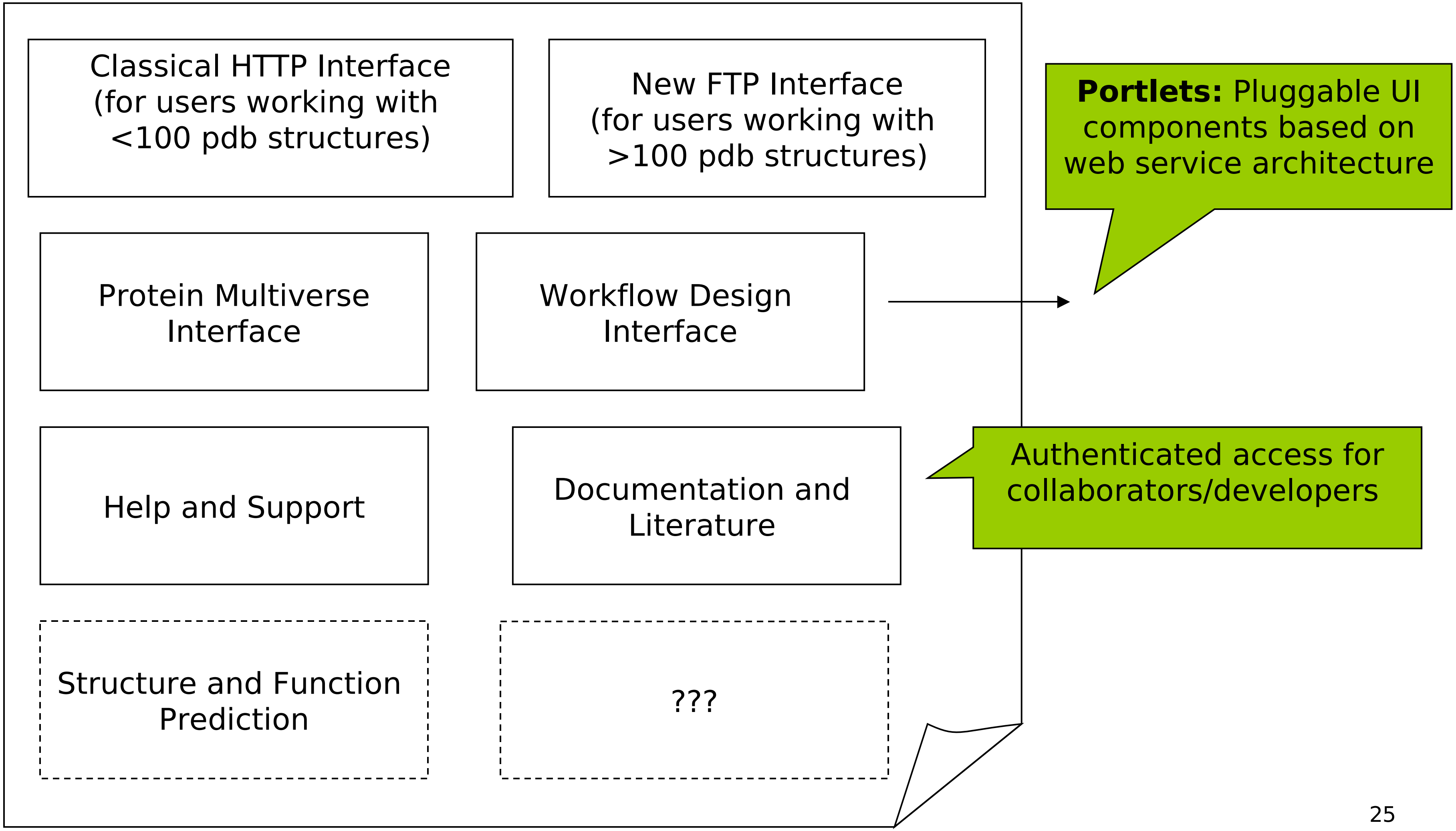
camion (noun) A low heavy horse cart without sides; used for haulage.

Synonyms: dray

Usage: An empty camion came bumping down the cobblestone street, pulled by two exhausted horses.

www.google.com

ProCKSI User Interface: Grid-based Portal Environment

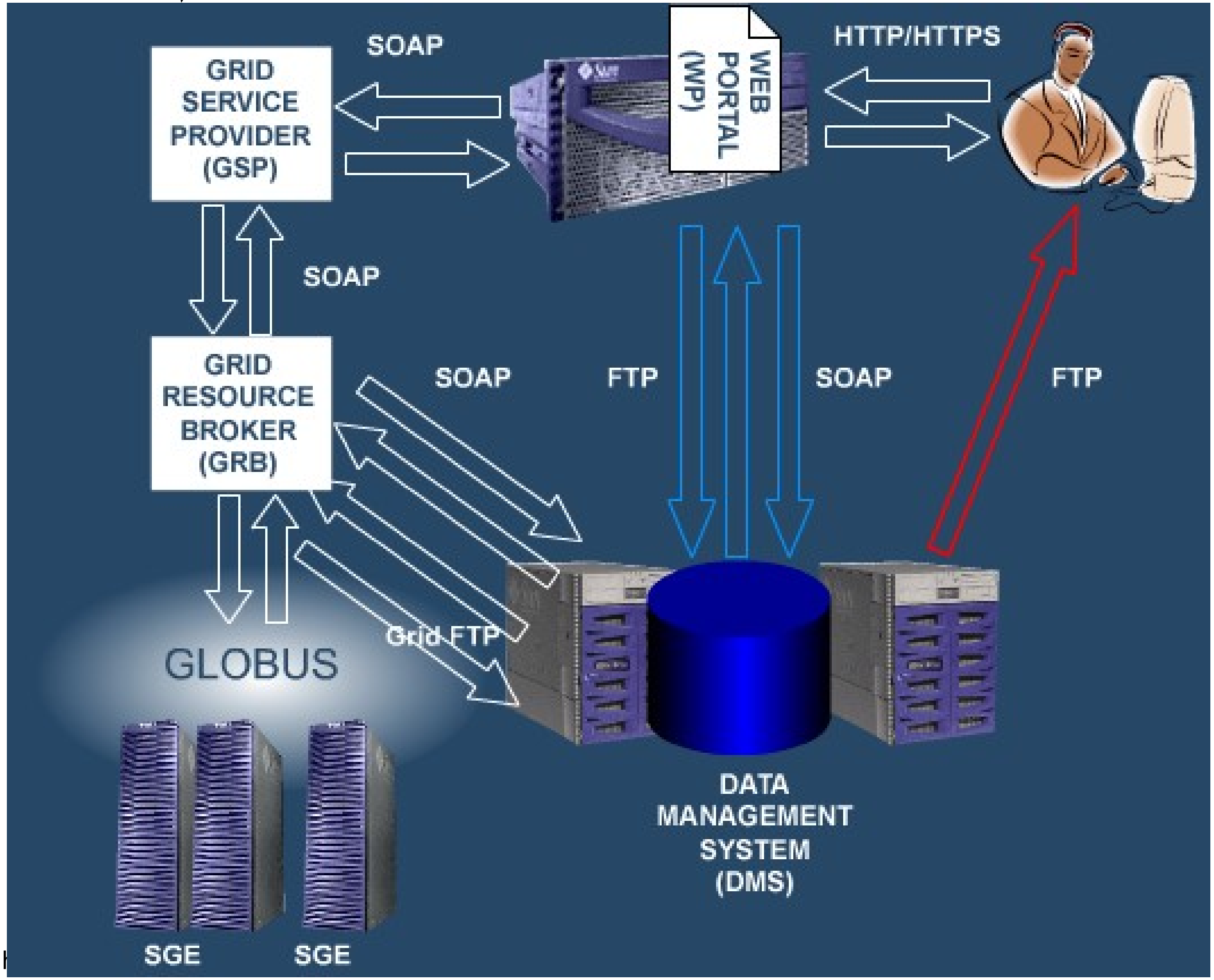

Classical HTTP Interface (for users working with <100 pdb structures)
New FTP Interface (for users working with >100 pdb structures)
Protein Multiverse Interface
Workflow Design Interface
Help and Support
Documentation and Literature
Structure and Function Prediction
???

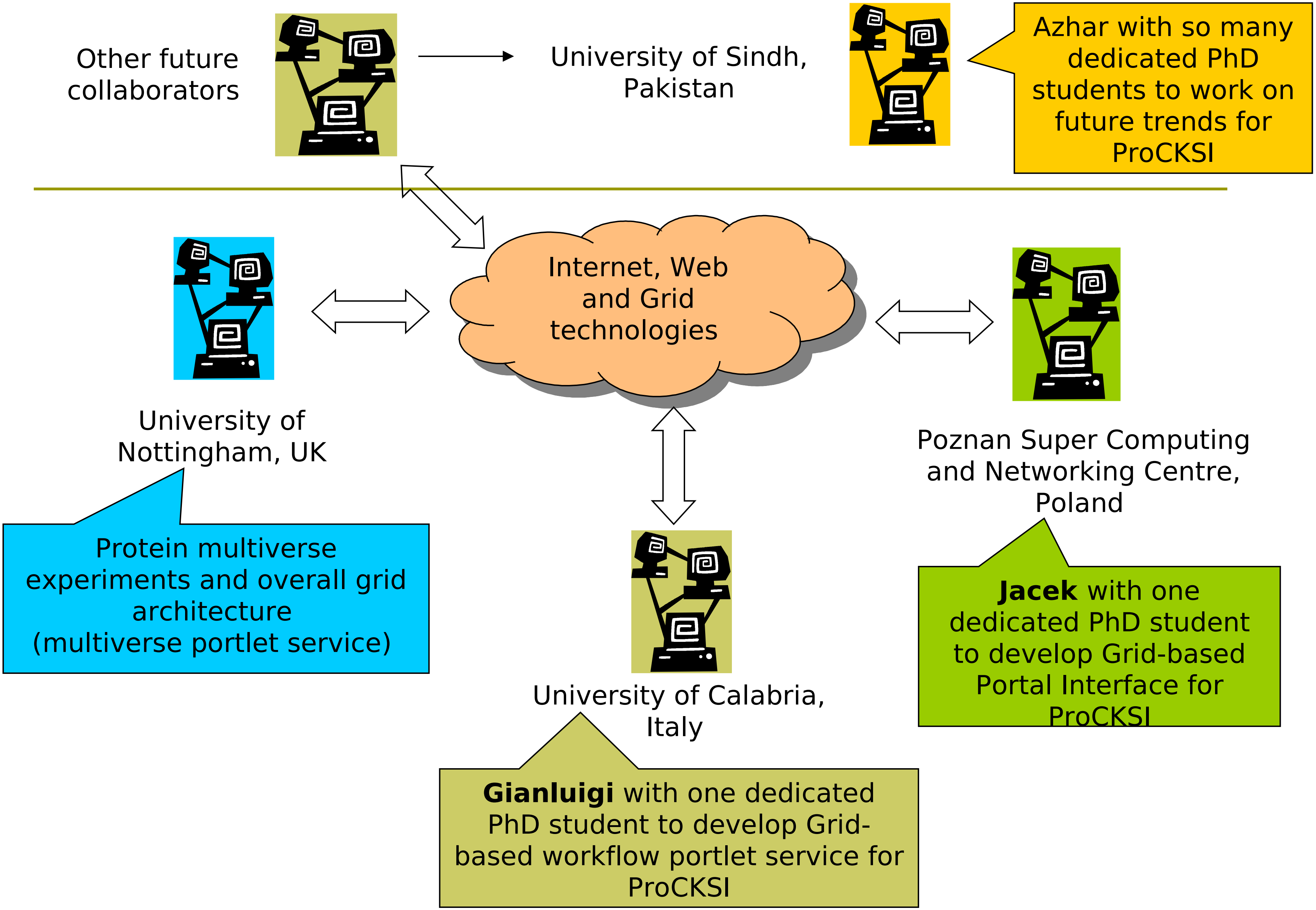
Portlets: Pluggable UI components based on web service architecture

Authenticated access for collaborators/developers

Web service factory:
Portals as web services

Example: PROGRESS Portal Access:
(Bogdanski Maciej et al. 2004)


GRID SERVICE PROVIDER (GSP)
SOAP
WEB PORTAL (WP)
HTTP/HTTPS
SOAP
GRID RESOURCE BROKER (GRB)
SOAP
FTP
SOAP
FTP
Grid FTP
GLOBUS
DATA MANAGEMENT SYSTEM (DMS)
SGE
SGE



Other future collaborators

University of Sindh, Pakistan

Azhar with so many dedicated PhD students to work on future trends for ProCKSI

Internet, Web and Grid technologies

University of Nottingham, UK

Protein multiverse experiments and overall grid architecture (multiverse portlet service)

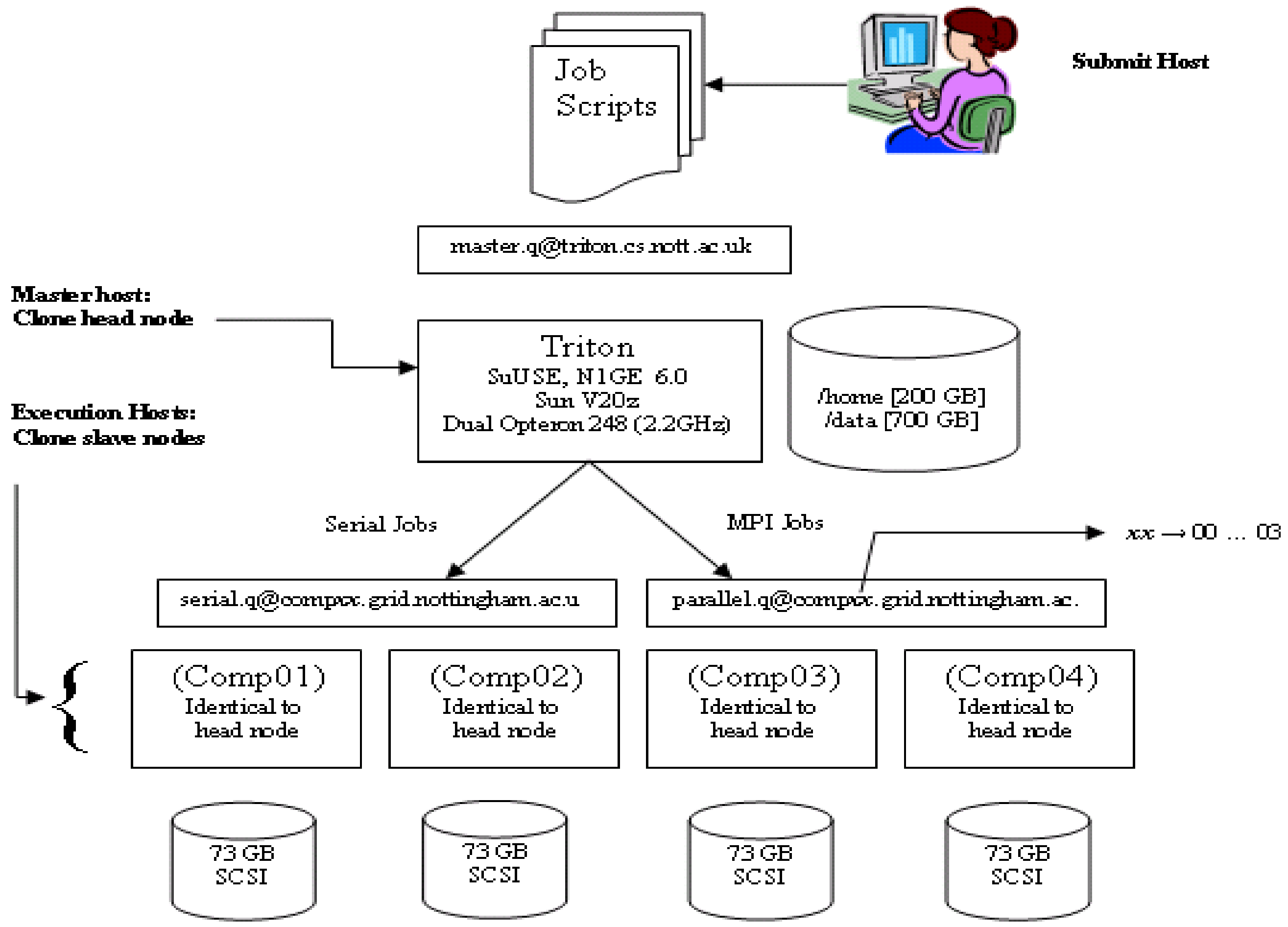
Poznan Super Computing and Networking Centre, Poland

Jacek with one dedicated PhD student to develop Grid-based Portal Interface for ProCKSI

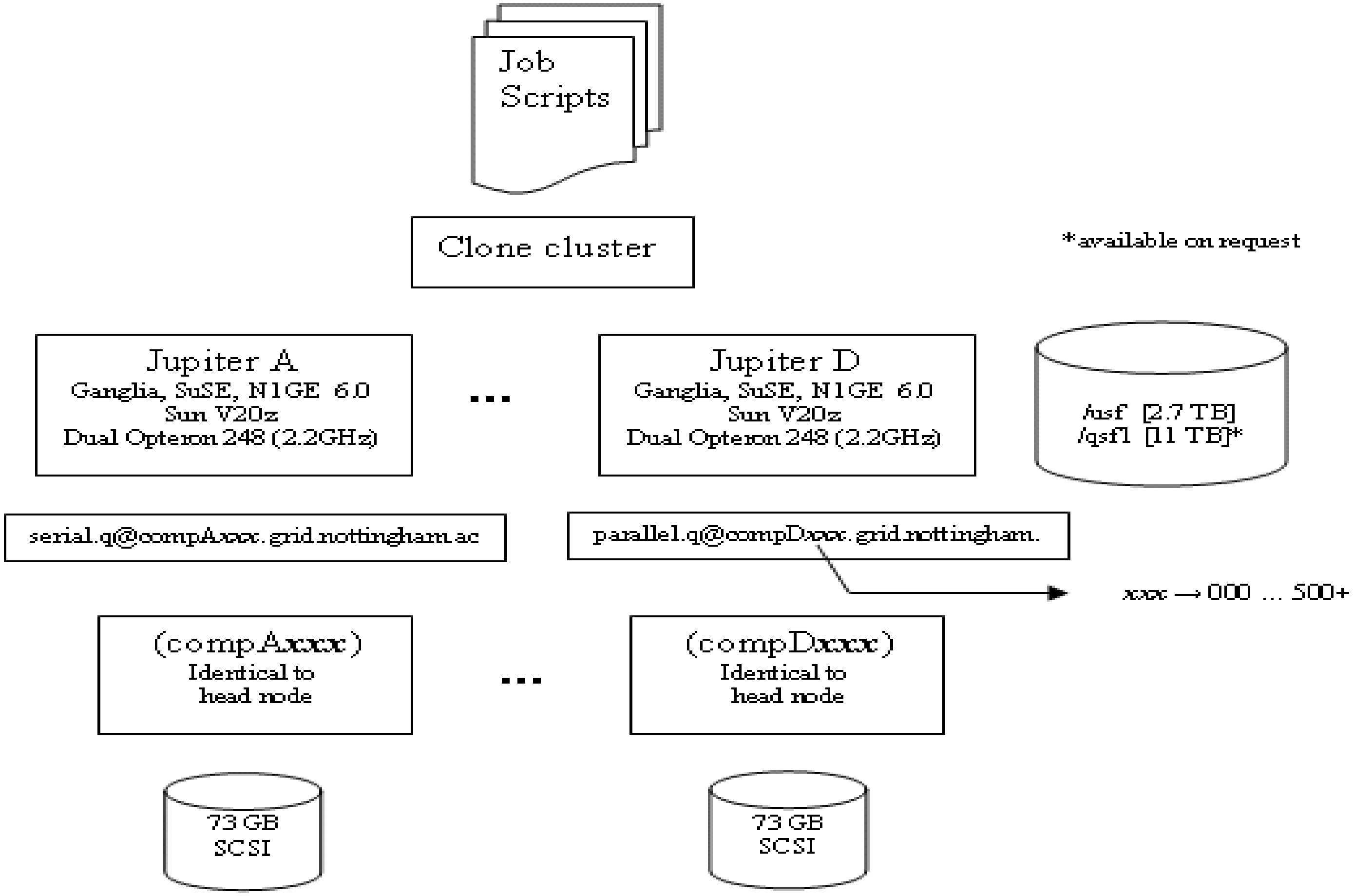
University of Calabria, Italy

Gianluigi with one dedicated PhD student to develop Grid-based workflow portlet service for ProCKSI

University of Nottingham: Triton



University of Nottingham: Jupiter



- Related work
- Complexity of the Problem
- Architectural Design
- Program Workflow Design (PWD)
- Infrastructure Details
- **Discussion**

Discussion

- Is it OK?