SCEAPI: A Unified Restful Web APIs for High-Performance Computing

Cao Rongqiang, Wang Xiaoning, Xiao Haili, Lu Shasha, Zhao Yining
Computer Network Information Center
Chinese Academy of Science
2016.10.12
Outlines

• Introduction to our team
• China Scientific Computing Environment
• SCEAPI
• Successful examples
• Future work
Introduction to our team

- Build and maintain China Scientific Computing Environment;
- Develop and optimize massively parallel computing applications in various subject areas;
- Provide technical support and consultation on HPC.
Scientific Computing Environment

- CPU: 1300Tflops
  - deepcomp7000; ERA
- National Supercomputer Centers
- GPU: 3000 Tflops
  - 11 institutes from CAS
Scientific Computing Environment

90+ Grid Apps

BioInfo

Chem

Algorithm

Material

CFD

Others

Development
Scientific Computing Environment

Portal / Command Line

Programming Interface

Scientific Computing Cloud

Unified Service, Schedule and Management
Outlines

- Introduction to our team
- China Scientific Computing Environment
- SCEAPI
- Successful examples
- Future work
Why we need RESTful API?

Massive resources:
✓ SCE
✓ More supercomputer ..

Various requirements:
✓ Users
✓ Applications
✓ Data

Many disciplines:
✓ Chemistry, physics, biology, astronomy …
✓ Engineering, economy, environment …

Feature-rich client:
✓ Scripts
✓ Command line
✓ Web portal
✓ Mobile app.

Cross platform and languages
Our vision: computing as infrastructure
SCEAPI

- Architecture

Client, Web Application - HTML/AJAX

HTTP Request

SCEAPI-REST

JSON Response

Authentication
LDAP: User Info

Persistent data: Application ID, session

SCEAPI Library (multi-thread, automatic)

SCE Middleware
the modular and layered architecture
User Management

<table>
<thead>
<tr>
<th>No.</th>
<th>method</th>
<th>path</th>
<th>remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET</td>
<td>/users/login</td>
<td>Login to the SCE environment</td>
</tr>
<tr>
<td></td>
<td>POST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>GET</td>
<td>/user/logout</td>
<td>Exit from the SCE environment</td>
</tr>
<tr>
<td>3</td>
<td>POST</td>
<td>/users/{user}/newpwd</td>
<td>Change password, or reset password if you are an administrator.</td>
</tr>
</tbody>
</table>

Forget your own password?
Redirect url: http://user.scgrid.cn/scgriduser/password/forget
### Job management

<table>
<thead>
<tr>
<th>No.</th>
<th>method</th>
<th>path</th>
<th>remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET</td>
<td>/jobs</td>
<td>Return a list of qualifying jobs.</td>
</tr>
<tr>
<td>2</td>
<td>GET</td>
<td>/jobs/update</td>
<td>Updated jobs for some time.</td>
</tr>
<tr>
<td>3</td>
<td>POST</td>
<td>/jobs</td>
<td>Submit a job depict by JSDL</td>
</tr>
<tr>
<td>4</td>
<td>GET</td>
<td>/jobs/{jobid}</td>
<td>Return information of the specific job</td>
</tr>
<tr>
<td>5</td>
<td>DELETE</td>
<td>/jobs/{jobid}</td>
<td>Kill the specific job.</td>
</tr>
<tr>
<td>6</td>
<td>GET</td>
<td>/jobs/{jobid}/status</td>
<td>Query status of the specific job</td>
</tr>
<tr>
<td>7</td>
<td>PUT</td>
<td>/jobs/{jobid}/status</td>
<td>Set status of the specific job</td>
</tr>
</tbody>
</table>
• File Transfer

<table>
<thead>
<tr>
<th>No.</th>
<th>method</th>
<th>path</th>
<th>remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET</td>
<td>/data/jobs/{jobid}/cs</td>
<td>List job directory in SCE.</td>
</tr>
<tr>
<td>2</td>
<td>GET</td>
<td>/data/jobs/{jobid}/hpc</td>
<td>List job directory in HPC.</td>
</tr>
<tr>
<td>3</td>
<td>GET</td>
<td>/data/jobs/{jobid}/hpc/{fileName}/view</td>
<td>View some lines in the text file.</td>
</tr>
<tr>
<td>4</td>
<td>POST</td>
<td>/jobs/{jobid}</td>
<td>Upload files.</td>
</tr>
<tr>
<td>5</td>
<td>POST</td>
<td>/data/jobs/{jobid}/cs/{fileName}</td>
<td>Upload a file to the specific directory.</td>
</tr>
<tr>
<td>6</td>
<td>GET</td>
<td>/data/jobs/{jobid}/cs/{fileName}</td>
<td>Down the specific file from SCE.</td>
</tr>
</tbody>
</table>
• Advanced feature: data transfer
  – Big data? many small files?

<table>
<thead>
<tr>
<th>No.</th>
<th>method</th>
<th>path</th>
<th>remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET</td>
<td>/data/jobs/{ujid}/fileUpload</td>
<td>Get a url for uploading a file, and upload the file with an independent service.</td>
</tr>
<tr>
<td>2</td>
<td>GET</td>
<td>/data/jobs/{ujid}/files Upload</td>
<td>Get a url for uploading several files, and upload the files with an independent service.</td>
</tr>
<tr>
<td>3</td>
<td>GET</td>
<td>/data/jobs/{ujid}/file Download</td>
<td>Get a url for downloading a file, and upload the file with an independent service.</td>
</tr>
</tbody>
</table>
### Advanced feature: Job Recycle Bin

<table>
<thead>
<tr>
<th>No.</th>
<th>method</th>
<th>path</th>
<th>remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DELETE</td>
<td>/jobs/recycleBin/delete</td>
<td>Delete specified jobs or files immediately or to the recycle bin.</td>
</tr>
<tr>
<td>2</td>
<td>DELETE</td>
<td>/jobs/recycleBin/empty</td>
<td>Empty all things in the recycle bin.</td>
</tr>
<tr>
<td>3</td>
<td>POST</td>
<td>/jobs/recycleBin/restore</td>
<td>Restore specified jobs or files from the recycle bin.</td>
</tr>
</tbody>
</table>
• Typical scenarios
  – Submit a job
    1. Collect parameters and generate the JSDL file
    2. Submit the JSDL file  [POST /jobs]
    3. Upload the input file/files  [ POST /jobs/{jobid}]  
    4. Start the job  [PUT /jobs/{jobid}/status]
  – Submit a batch of jobs
    • Write some scripts
    • Develop web communities
SCEAPI

• Typical scenarios
  – Manage jobs
    • Query job information about jobs
    • View some lines in text files when the job is running
    • Kill some jobs
    • List workspace for a job
    • Download output files

• Advanced
  • Delete jobs
  • Delete workspace for jobs
SCE API

- How to use?

Apply
- Id and Key
- Targets
- Accounts for test

Develop & test
- Dedicated test bed
- SSO & OAuth2
- Programming
- Testing

Deploy
- Application audit
- Deploy to SCE:
  - Replace root urls for API, SSO and OAuth2

Java library & demo
Python library & demo
JavaScript library
Outlines

• Introduction to our team
• China Scientific Computing Environment
• SCEAPI
• Successful examples
• Future work
Successful examples
CNIC-ATLAS Cooperation

- **ATLAS**
  - ATLAS is a particle physics experiment in the Large Hadron Collider at the CERN laboratory
  - Need more computing resources

- **SCE**
  - provide computing resources quickly
  - > 400+ users from CAS & universities;
  - > 100,000,000+ CPU hours
CNIC-ATLAS Cooperation

• Quick steps:
  – 1, confirmed interfaces, PBS-like Commands by Python
  – 2, design and implemented Python library and PBS commands based on SCEAPI;
  – 3, ARC-CE integrated SCE based on python examples and library;
  – 4, submitted jobs to SCE.

Examples:
- job management: sqsub, qstatus, qkill
- data transfer: put, get
- authorization: login

Python library:
- login/logout
- resource query
- encode/decode json
- job management
- md5 generation

Dependent libraries:
- SCEAPI
- requests & toolbelt
- md5
- json
Future work

- Add more services, e.g. resource reservation
- Improve performance, e.g. data transfer, log

More resources
- Job schedule
- Security

More users
- Easy-to-use
- Stability

More areas
- Simple & flexible
- Feature-rich

Stable  Easy-to-use  Professional
Thank you!