The CIPRES Science Gateway: A Community Resource for Phylogenetic Analyses

Mark Miller, Wayne Pfeiffer, and Terri Schwartz

San Diego Supercomputer Center
Systematics is the study of biological diversity and its origins. It focuses on understanding evolutionary relationships among organisms.
Originally, evolutionary relationships were inferred from morphology alone:

Morphological characters are scored “by hand” to create matrices of characters.

Scoring occurs via low volume/low throughput methodologies.

Even though tree inference is NP hard, matrices created using morphological characters alone are typically relatively small, so computations are relatively tractable (with heuristics developed by the community).
Evolutionary relationships are now inferred from DNA sequence comparisons:

Align sequences to determine evolutionary equivalence:

Infer evolutionary relationships based on some set of assumptions:
Inferring Evolutionary relationships from DNA sequence comparisons is powerful:

DNA sequences are determined by fully automated procedures.

Sequence data can be gathered from many species at scales from gene to whole genome.

The high speed and low cost of NexGen Sequencing means new levels of sensitivity and resolution can be obtained.
Inferring Evolutionary relationships from DNA sequence comparisons is powerful, **BUT:**

Current analyses often involve 1000’s of species and 1000’s of characters, creating very large matrices.

Sequence alignment and Tree inference are NP hard, so even with heuristics, computational power often limits the analyses (already).

The length of tree search analysis scales exponentially with number of taxa and with number of characters with codes in current use.

There are at least $10^7$ species, each with 3000 - 30,000 genes, so the need for computational power and new approaches will continue to grow.
Inferring Evolutionary relationships from DNA sequence comparisons is powerful, **BUT:**

Current analyses often involve **1000’s of species and 1000’s of characters**, creating very large matrices.

Sequence alignment and Tree inference are NP hard, so even with heuristics, computational power often limits the analyses (already).

The length of tree search analysis scales exponentially with number of taxa and with number of characters with codes in current use.

There are at least $10^7$ species, each with 3000 - 30,000 genes, so the need for computational power and new approaches will continue to grow.
Inferring Evolutionary relationships from DNA sequence comparisons is powerful, **BUT:**

Current analyses often involve **1000’s of species and 1000’s of characters**, creating very large matrices.

Sequence alignment and Tree inference are NP hard, so even with heuristics, computational power often limits the analyses (already).

The length of tree search analysis scales exponentially with number of taxa and with number of characters with codes in current use.

There are at least $10^7$ species, each with 3000 - 30,000 genes, so the need for computational power and new approaches will continue to grow.
Inferring Evolutionary relationships from DNA sequence comparisons is powerful, **BUT:**

Current analyses often involve **1000’s of species and 1000’s of characters**, creating very large matrices.

Sequence alignment and Tree inference are NP hard, so even with heuristics, **computational power often limits the analyses (already).**

The length of tree search analysis **scales exponentially with number of taxa and with number of characters with codes in current use.**

There are at least **10^7 species**, each with 3000 - 30,000 genes, so the need for computational power and new approaches will continue to grow.
Inferring Evolutionary relationships from DNA sequence comparisons is powerful, **BUT:**

Current analyses often involve **1000’s of species and 1000’s of characters**, creating very large matrices.

Sequence alignment and Tree inference are NP hard, so even with heuristics, **computational power often limits the analyses (already).**

The length of tree search analysis **scales exponentially with number of taxa and with number of characters with codes in current use.**

There are at least **$10^7$ species, each with 3000 - 30,000 genes**, so the need for computational power and new approaches will continue to grow.
The CIPRES Project was created to support this new age of large phylogenetic data sets. The project had as its principal goals:

1. Developing heuristics and tools for analyzing the large DNA data sets that are available.

2. Improving researcher access to computational resources.
The CIPRES Portal was designed to allow users analyze large sequence data sets using community codes on a significant computational resource.

The CIPRES Portal provided:

• Login-protected personal user space for storing results indefinitely.

• Access to most/all native command line options for each code.

• Support for adding new tools and new versions as needed.
Workflow for the CIPRES Portal:

1. Assemble Sequences
2. Upload to Portal
3. Run Alignment
4. Run Tree Inference
5. Store
6. Download
7. Run
8. Post-Tree Analysis

47,500 total jobs in 30 months
Limitations of the original CIPRES Portal

- all jobs were run serially (efficient, but no gain in wall time)
- runs were limited to 72 hours
- demand for job runs was increasing
- the cluster was modest (16 X 8-way dual core nodes)
- the cluster was at the end of its useful lifetime
- funding for the project was ending

This is not a scalable, sustainable solution!
The solution: make community codes available on scalable, sustainable resources via Science Gateway Program.
The solution: make community codes available on scalable, sustainable resources via Science Gateway Program.

- TeraGrid
- Triton
- Workbench Framework

Parallel codes
Serial codes
Greater than 90% of all computational time was used for three tree inference codes: MrBayes, RAxML, and GARLI.

Deploy parallel versions of these codes on TeraGrid Machines; initially using Globus/GRAM.

Work with community developers to improve the speed-up available through the parallel codes offered by CSG.

Add new parallel codes (e.g. MAFFT) as they appear in the community.

Keep other serial codes on local SDSC resources that provide the project with fee-for-service cycles.
CIPRES Science Gateway Code Improvements

- Hybrid MPI/Pthreads version of RAxML was developed
- Hybrid MPI/OpenMP version of MrBayes was developed
- Single-node runs are more efficient than before
- Multi-node runs with more cores are possible
  - Scalability before was limited to about 8 cores for typical analyses
  - Codes now scale well to 10s of cores for typical analyses
Parallel code profiles on Trestles

<table>
<thead>
<tr>
<th>Code</th>
<th>Type</th>
<th>Max cores</th>
<th>Speed-up</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>MrBayes</td>
<td>Hybrid MPI/OpenMP</td>
<td>32</td>
<td>2.4 X (1 nodes)</td>
<td>~60%</td>
</tr>
<tr>
<td>RAxML</td>
<td>Hybrid MPI/Pthreads</td>
<td>60</td>
<td>3.0 X (2 nodes)</td>
<td>~60%</td>
</tr>
<tr>
<td>GARLI</td>
<td>MPI</td>
<td>100</td>
<td>77 X (100 nodes)</td>
<td>77-94%</td>
</tr>
<tr>
<td>MAFFT</td>
<td>Pthreads</td>
<td>32</td>
<td>10 X</td>
<td>25%</td>
</tr>
</tbody>
</table>
What happens if you build it and too many people come???

![Graph showing initial allocation and SUs (thousand) over months with a question mark at the end.](image-url)
What happens if you build it and too many people come???

- make sure resource use is efficient
- make sure resource use delivers impact
- implement policies to maximize access for all
What happens if you build it and too many people come???

• make sure resource use is efficient

• make sure resource use delivers impact

• implement policies to maximize access for all
Job Attrition on the CIPRES Science Gateway

*March – August 2010
## Error Impact analysis

<table>
<thead>
<tr>
<th>Error Type</th>
<th>CPU time</th>
<th>User</th>
<th>Staff</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input error</td>
<td>0</td>
<td>med</td>
<td>low</td>
</tr>
<tr>
<td>Machine error</td>
<td>0</td>
<td>med</td>
<td>low</td>
</tr>
<tr>
<td>Communication error</td>
<td>high</td>
<td>high</td>
<td>high</td>
</tr>
<tr>
<td>Unknown error</td>
<td>?</td>
<td>med</td>
<td>low</td>
</tr>
</tbody>
</table>
CONCLUSION: Time to refactor the job monitoring system!
Normal Operation

User Submission → Command line; files → Globus “gsissh” → TG Machine

Running Task Table

User DB
Normal Operation

User Submission → Command line; files → Running Task Table → Globus “gsissh” → TG Machine

User DB
Normal Operation

User Submission → Command line; files → Globus "gsissh" → TG Machine

Running Task Table

User DB
Normal Operation

User Submission → Command line; files → Globus “gsissh” → TG Machine

Running Task Table ← Mark as “Done”

User DB
Normal Operation

User Submission → Command line; files → Globus “gsissh” → TG Machine

Running Task Table → Mark as “Done”

User DB

LoadResults Daemon fetches via Grid ftp; puts results in the user DB.
Abnormal Operation

User Submission → Command line; files → Globus "gsissh" → TG Machine

Running Task Table

User DB

Communication down, not marked as "Done", results not retrieved
Abnormal Operation

User Submission $\rightarrow$ *checkJobs Daemon* $\rightarrow$ Globus “gsissh” $\rightarrow$ TG Machine

Running Task Table

User DB
Abnormal Operation

User Submission -> checkJobs Daemon -> Running Task Table

Globus "gsissh" -> TG Machine

User DB
Abnormal Operation

User Submission

checkJobs Daemon

Globus "gsissh"

TG Machine

Running Task Table

User DB

Mark as "Done"
Abnormal Operation

User Submission → checkJobs Daemon → Running Task Table → Globus "gsissh" → TG Machine

LoadResults Daemon fetches via Grid ftp; puts results in the user DB.
## Jobs Saved by the GSISSH / Task Table System

<table>
<thead>
<tr>
<th></th>
<th>SEPT</th>
<th>OCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jobs Saved</td>
<td>159*</td>
<td>266*</td>
</tr>
</tbody>
</table>

* 7% of all submitted jobs
What happens if you build it and too many people come???

- make sure resource use is efficient
- make sure resource use delivers impact
- implement policies to maximize access for all
Make sure resource use delivers impact: Usage and Outcomes
CIPRES Science Gateway Usage Dec 2009 – April 2011

4X increase in users/month
CIPRES Science Gateway Usage Dec 2009 – April 2011

- 4X increase in users/month
- 2456 new TG users
CIPRES Science Gateway Usage Dec 2009 – April 2011

4X increase in users/month

2456 new TG users

At least 100 new TG users in each month
CIPRES Science Gateway Usage Dec 2009 – April 2011

4-fold increase in jobs submitted /month

Jobs submitted* / month

SU's / month (in thousands)

Month

Jan, Mar, May, Jul, Sep, Nov, Jan, Mar
CIPRES Science Gateway Usage Dec 2009 – April 2011

- 4-fold increase in jobs submitted/month
- 10+ fold increase in SU/month
CIPRES Science Gateway Usage Dec 2009 – April 2011

- 4-fold increase in jobs submitted/month
- 10+ fold increase in SU/month
- 45,500 total job submissions in 17 months
Intellectual Merit:

Publications enabled by the CIPRES Science Gateway/CIPRES Portal:

<table>
<thead>
<tr>
<th>Year</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011*</td>
<td>38</td>
</tr>
<tr>
<td>2010</td>
<td>74</td>
</tr>
<tr>
<td>2009</td>
<td>56</td>
</tr>
<tr>
<td>2008</td>
<td>4</td>
</tr>
</tbody>
</table>

*As of April 1, 2011

Publications in the pipeline:

<table>
<thead>
<tr>
<th>Status</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>In preparation</td>
<td>103</td>
</tr>
<tr>
<td>In review</td>
<td>24</td>
</tr>
</tbody>
</table>
Intellectual Merit:

“…quick and reliable results often help with brainstorming and pushing ‘things into the right direction.’ The service is invaluable.”
Broad Impact:

• Used for curriculum delivery by at least 57 instructors.

• Jobs run for researchers in 23/29 EPSCOR states.

• Routine submissions from Harvard, Berkeley, Stanford…..

• 76% of users are in the US or have a collaborator in the US

• In Q1, 2011, 28% of all TG users who ran jobs ran them from the CSG
Broad Impact:

“The wall time for these data sets are 30 min (2000 BS reps) [on the CSG], but locally it would take 24 hours for each run. So, instead of 10 days to complete all the analyses with a few local computers, I have publication quality trees overnight.”
Make sure resource use delivers impact: Usage and Outcomes

We know how many jobs are submitted, how many of these jobs produce useful outcomes?
Not All Job Submissions to *Trestles* Appear in the TGDB
Not All Job Submissions to *Trestles* Appear in the TGDB

Trestles jobs less than 2 minutes are not recorded in TGDB
Do jobs run successfully in less than 2 minutes?

Not All Job Submissions to Trestles Appear in the TGDB
Categorization of RAxML and MrBayes jobs by outcome and duration*

<table>
<thead>
<tr>
<th></th>
<th>RAxML</th>
<th>MrBayes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Jobs</td>
<td>Fraction</td>
</tr>
<tr>
<td>Failed</td>
<td>885</td>
<td>0.394</td>
</tr>
<tr>
<td>Succeeded</td>
<td>1363</td>
<td>0.606</td>
</tr>
<tr>
<td>&lt; 2 minutes</td>
<td>521</td>
<td>0.232</td>
</tr>
<tr>
<td>≥ 2 minutes</td>
<td>842</td>
<td>0.375</td>
</tr>
</tbody>
</table>

Categorization of RAxML and MrBayes jobs by outcome and duration*

<table>
<thead>
<tr>
<th></th>
<th>RAxML</th>
<th>MrBayes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Jobs</td>
<td>Fraction</td>
</tr>
<tr>
<td>Failed</td>
<td>885</td>
<td>0.394</td>
</tr>
<tr>
<td>Succeeded</td>
<td>1363</td>
<td>0.606</td>
</tr>
<tr>
<td>&lt; 2 minutes</td>
<td>521</td>
<td>0.232</td>
</tr>
<tr>
<td>≥ 2 minutes</td>
<td>842</td>
<td>0.375</td>
</tr>
</tbody>
</table>

Categorization of RAxML and MrBayes jobs by outcome and duration*

<table>
<thead>
<tr>
<th></th>
<th>RAxML</th>
<th>MrBayes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Jobs</td>
<td>Fraction</td>
</tr>
<tr>
<td>Failed</td>
<td>885</td>
<td>0.394</td>
</tr>
<tr>
<td>Succeeded</td>
<td>1363</td>
<td>0.606</td>
</tr>
<tr>
<td>&lt; 2 minutes</td>
<td>521</td>
<td>0.232</td>
</tr>
<tr>
<td>≥ 2 minutes</td>
<td>842</td>
<td>0.375</td>
</tr>
</tbody>
</table>

Categorization of CSG users by job outcome and duration *

<table>
<thead>
<tr>
<th>Outcome</th>
<th>Users</th>
<th>Fraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only Failed</td>
<td>50</td>
<td>0.13</td>
</tr>
<tr>
<td>Run Succeeded</td>
<td>345</td>
<td>0.87</td>
</tr>
<tr>
<td>Only &lt; 2 minutes</td>
<td>61</td>
<td>0.15</td>
</tr>
<tr>
<td>≥ 2 minutes**</td>
<td>284</td>
<td>0.72</td>
</tr>
</tbody>
</table>


** The user ran at least one job that produced results and recorded an SU charge.
### Categorization of CSG users by job outcome and duration *

<table>
<thead>
<tr>
<th>Outcome</th>
<th>Users</th>
<th>Fraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only Failed</td>
<td>50</td>
<td>0.13</td>
</tr>
<tr>
<td>Run Succeeded</td>
<td>345</td>
<td>0.87</td>
</tr>
<tr>
<td>Only &lt; 2 minutes</td>
<td>61</td>
<td>0.15</td>
</tr>
<tr>
<td>≥ 2 minutes**</td>
<td>284</td>
<td>0.72</td>
</tr>
</tbody>
</table>


**The user ran at least one job that produced results and recorded an SU charge.

15% of all users only submitted short jobs.
Categorization of CSG users by job outcome and duration *

<table>
<thead>
<tr>
<th></th>
<th>Users</th>
<th>Fraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only Failed</td>
<td>50</td>
<td>0.13</td>
</tr>
<tr>
<td>Run Succeeded</td>
<td>345</td>
<td>0.87</td>
</tr>
<tr>
<td>Only &lt; 2 minutes</td>
<td>61</td>
<td>0.15</td>
</tr>
<tr>
<td>$\geq$ 2 minutes**</td>
<td>284</td>
<td>0.72</td>
</tr>
</tbody>
</table>


** The user ran at least one job that produced results and recorded an SU charge.

13% of all users did not submit a successful job
What happens if you build it and too many people come???

• make sure resource use is efficient
• make sure resource use delivers impact
• implement policies to maximize access for all
How much growth can we manage?

- Jobs submitted* / month
- SU's / month (in thousands)

Month:
- Jan
- Mar
- May
- Jul
- Sep
- Nov
- Jan
- Mar
Establish a Fair Use Policy

- How many SUs should each user be permitted?
- How much influence should affiliation with a US institution have on resource allocation?
Establish a Fair Use Policy

Projected SUs /
Cumulative Use

Cur

Projected SUs/year
Fraction of all users
Establish a Fair Use Policy

98% of users consume 58% of all resources
Establish a Fair Use Policy

2% of users consume 42% of all resources
Establish a limit of 50,000 SUs per year, which impacts 42 users.
Establish a Fair Use Policy

- Users are permitted to use 50,000 SUs annually from the community allocation. Anyone from any institution can register.

- Users who wish to use in excess of this amount can apply for a personal allocation. This would include users at US institutions and those who have collaborators at a US institution. User can access their allocation using the CSG.
Tools required to implement the CIPRES SG Fair Use Policy:

- ability to monitor usage by each account automatically (complete)
- ability to halt submissions from a given user account (complete)
- ability to charge to a user’s personal allocation (complete)
- ability for users to track their SU consumption (complete)
- ability to forecast SU cost of a job for users (in progress)
Next Steps:

• Deploy the new user management tools in production.

• Implement the new Fair Usage Policy.

• Add new codes: BEAST, RAxML Light, BEST.

• Partner with the iPlant Project as a service provider.
Next Steps:

- Data Base Access
- NGS Sequencing
- Ancestral Character Estimation
- Tree Reconciliation
- Taxonomic Name Resolution
- Phylogenetic Workflows

- TeraGrid
- CSG
- iPDE
- iPDB

Parallel codes
Acknowledgements:

<table>
<thead>
<tr>
<th>CIPRES Science Gateway</th>
<th>Terri Schwartz</th>
</tr>
</thead>
<tbody>
<tr>
<td>TeraGrid Hybrid Code Development</td>
<td>Wayne Pfeiffer</td>
</tr>
<tr>
<td></td>
<td>Alexandros Stamatakis</td>
</tr>
<tr>
<td>TeraGrid Implementation Support</td>
<td>Nancy Wilkins-Diehr</td>
</tr>
<tr>
<td></td>
<td>Doru Marcusiu</td>
</tr>
<tr>
<td></td>
<td>Leo Carson</td>
</tr>
<tr>
<td>Workbench Framework:</td>
<td>Paul Hoover</td>
</tr>
<tr>
<td></td>
<td>Lucie Chan</td>
</tr>
</tbody>
</table>
Make sure resource use delivers impact: analyze resource consumption

Make sure resource use delivers impact: analyze resource consumption


Who are these users?

Cumulative Use
Identify the TOP 20 Users in each month.

Identify the TOP 20 Users in each month.

Conclusion:
CSG Usage is not driven by 3-4 high end users.

Make sure resource use delivers impact: analyze resource consumption

<table>
<thead>
<tr>
<th>SUs</th>
<th>% of Users</th>
<th>% total SU</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – 30 K</td>
<td>97</td>
<td>45</td>
</tr>
<tr>
<td>30 – 300,000 K</td>
<td>3</td>
<td>55</td>
</tr>
<tr>
<td>&gt; 300,000 K</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Make sure resource use delivers impact: analyze resource consumption

<table>
<thead>
<tr>
<th>SUs</th>
<th>% of Users</th>
<th>% total SU</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – 30 K</td>
<td>97</td>
<td>45</td>
</tr>
<tr>
<td>30 – 300,000 K</td>
<td>3</td>
<td>55</td>
</tr>
<tr>
<td>&gt; 300,000 K</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

50% of users in this bin are from institutions outside the US

Make sure resource use delivers impact: Establish a Fair Use Policy

<table>
<thead>
<tr>
<th>SUs</th>
<th>Number of Users</th>
<th>% total SU</th>
<th>% per user</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 100</td>
<td>936 (58.5%)</td>
<td>0.2</td>
<td>0.0003</td>
</tr>
<tr>
<td>100 - 999</td>
<td>260 (16.3%)</td>
<td>1.6</td>
<td>0.005</td>
</tr>
<tr>
<td>1 – 10 K</td>
<td>264 (16.5%)</td>
<td>16</td>
<td>0.06</td>
</tr>
<tr>
<td>10 – 30 K</td>
<td>93 (5.8%)</td>
<td>27</td>
<td>0.29</td>
</tr>
<tr>
<td>30 – 50 K</td>
<td>20 (1.3%)</td>
<td>14</td>
<td>0.69</td>
</tr>
<tr>
<td>50 – 100 K</td>
<td>15 (0.9%)</td>
<td>16</td>
<td>1.06</td>
</tr>
<tr>
<td>&gt; 100 K</td>
<td>12 (0.8%)</td>
<td>26</td>
<td>2.14</td>
</tr>
</tbody>
</table>
Make sure resource use delivers impact: Establish a Fair Use Policy

<table>
<thead>
<tr>
<th>SUs</th>
<th>Number of Users</th>
<th>% total SU</th>
<th>% per user</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 100</td>
<td>936 (58.5%)</td>
<td>0.2</td>
<td>0.0003</td>
</tr>
<tr>
<td>100 - 999</td>
<td>260 (16.3%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 – 10 K</td>
<td>264 (16.5%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 – 30 K</td>
<td>93 ( 5.8%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30 – 50 K</td>
<td>20 ( 1.3%)</td>
<td>14</td>
<td>0.69</td>
</tr>
<tr>
<td>50 – 100 K</td>
<td>15 ( 0.9%)</td>
<td>16</td>
<td>1.06</td>
</tr>
<tr>
<td>&gt; 100 K</td>
<td>12 ( 0.8%)</td>
<td>26</td>
<td>2.14</td>
</tr>
</tbody>
</table>

Establish 50,000 SUs as the limit for usage from the CSG community account. (Same as Trestles startup account).
Make sure resource use delivers impact: Establish a Fair Use Policy

<table>
<thead>
<tr>
<th>SUs</th>
<th>Number of Users</th>
<th>% total SU</th>
<th>% per user</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 100</td>
<td>936 (58.5%)</td>
<td>0.2</td>
<td>0.0003</td>
</tr>
<tr>
<td>100 - 999</td>
<td>260 (16.3%)</td>
<td>1.6</td>
<td>0.005</td>
</tr>
<tr>
<td>1 – 10 K</td>
<td>264 (16.5%)</td>
<td>16</td>
<td>0.06</td>
</tr>
<tr>
<td>10 – 30 K</td>
<td>93 ( 5.8%)</td>
<td>27</td>
<td>0.29</td>
</tr>
<tr>
<td>30 – 50 K</td>
<td>20 ( 1.3%)</td>
<td>14</td>
<td>0.69</td>
</tr>
<tr>
<td>50 – 100 K</td>
<td>15 ( 0.9%)</td>
<td>16</td>
<td>1.06</td>
</tr>
<tr>
<td>&gt; 100 K</td>
<td>12 ( 0.8%)</td>
<td>26</td>
<td>2.14</td>
</tr>
</tbody>
</table>

Impacts 42 Users only
Make sure resource use delivers impact: resource consumption

Notify users of their usage level