

# Pathways into Large Parameter Search Spaces: Experiences with Molecular Hyperdynamics



Justin M. Wozniak, Santanu Chatterjee, Paul Brenner,  
Douglas Thain, Aaron Striegel and Jesus A. Izaguirre

11th September 2008

Current location: Argonne National Laboratory, USA  
[wozniak@mcs.anl.gov](mailto:wozniak@mcs.anl.gov)

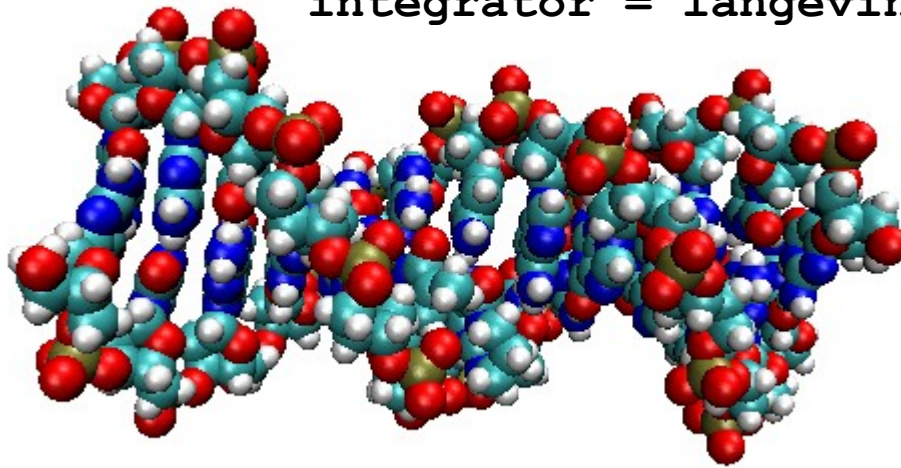


# Scientific Repositories

---

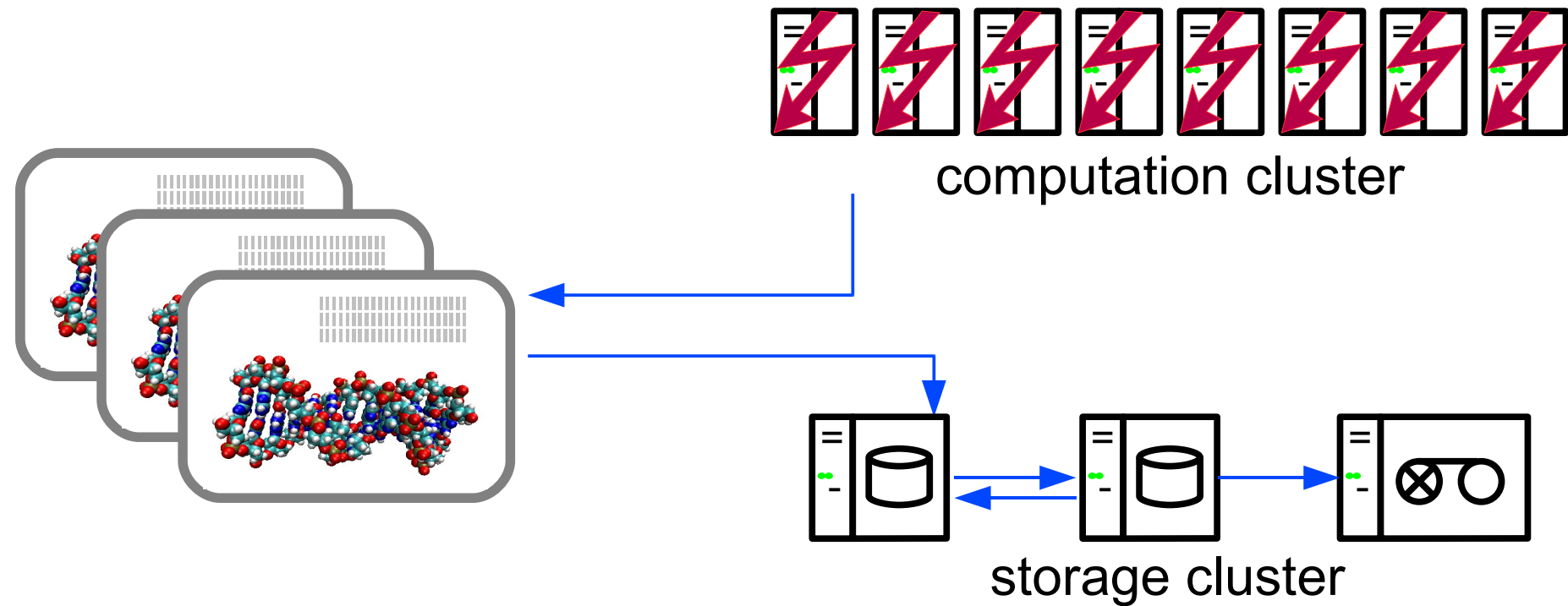
---

```
type = dna  
temperature = 300  
integrator = langevin
```



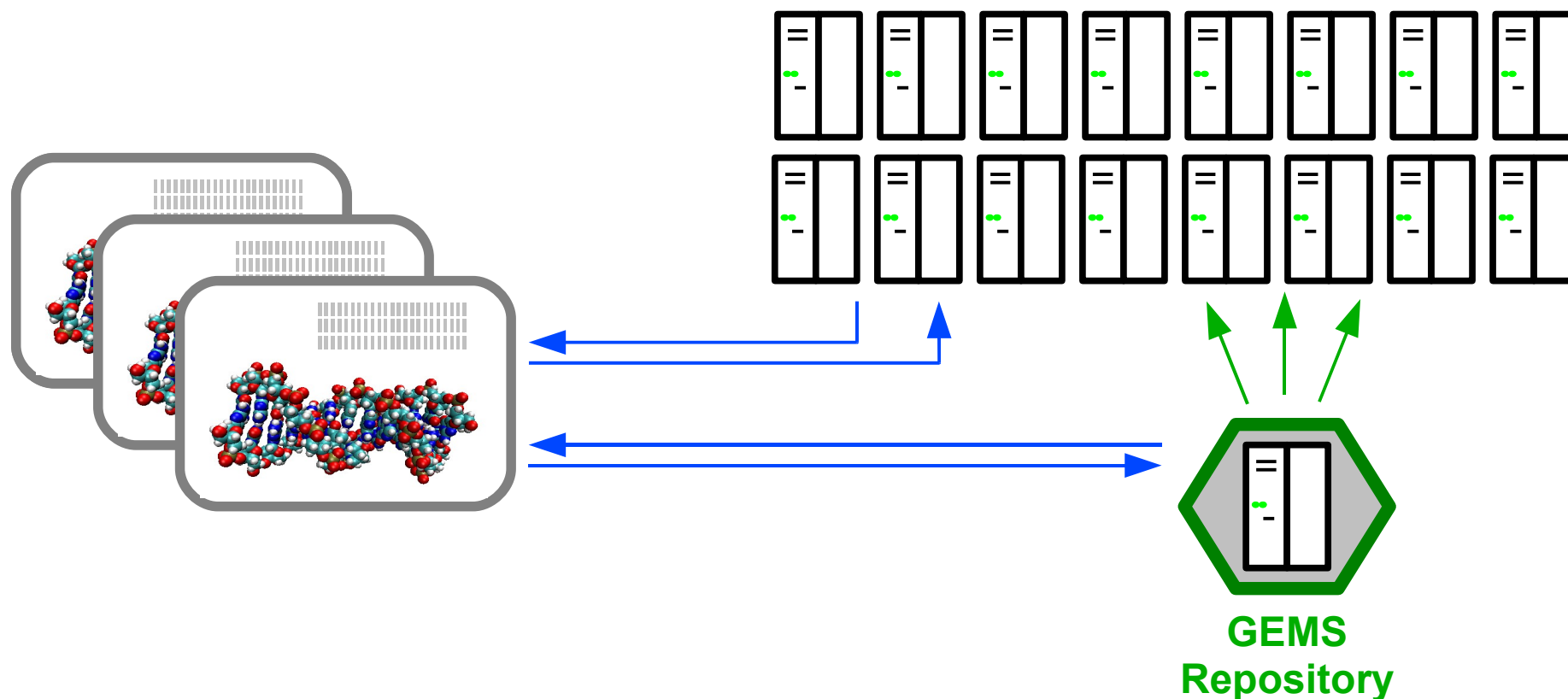
- Start with user data requirements

# Scientific Repositories



- Cluster-based simulation storage infrastructure

# Scientific Repositories



- The desktop grid reality – we just have a lot of computers
- GEMS: Grid-Enabled Molecular Simulation



- Where can I put my data?
- How can I keep it available?
- How will I find it again?

# Motivation



- User needs:
  - Large scale file space
  - File organization
  - Storage site management
  - Access for existing software
  - **Workflow structures for large scale projects**
- Available resources:
  - Large, uncontrolled storage network
  - A searchable, parameterized replica management system

# Outline

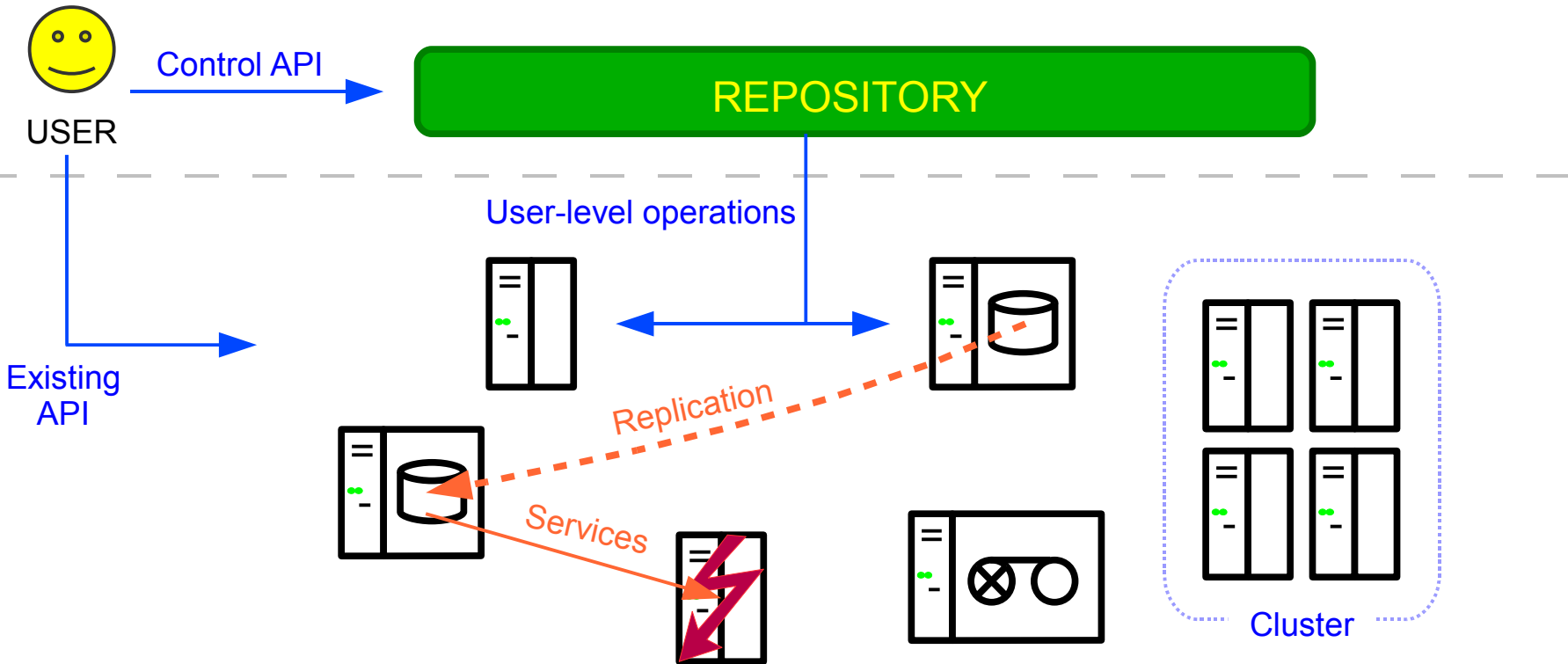


## I. Runtime repositories

- Database abstractions
- Basic utilities

## II. Workflow-like structures

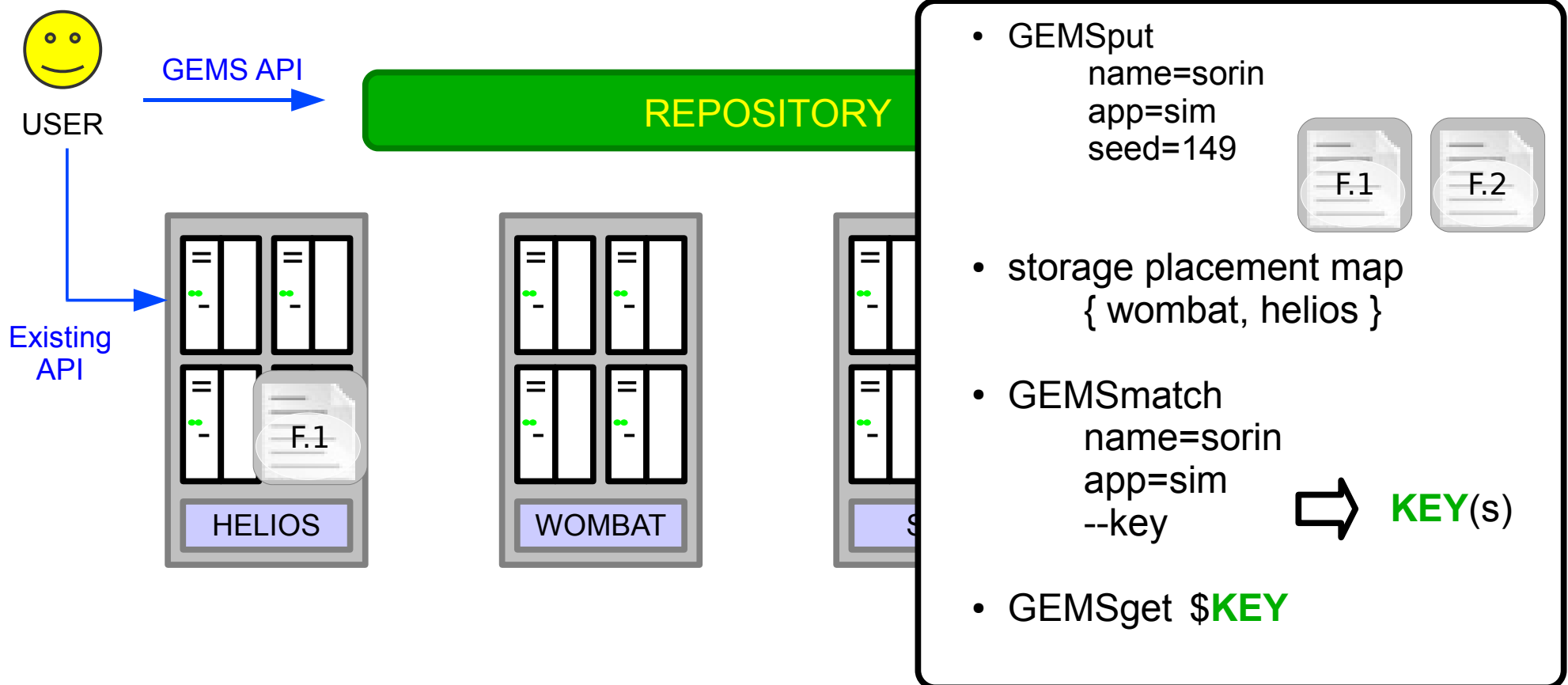
- Application in hyperdynamics
- Event-driven workflows
- Scalable job submission



- Repository users expect a high level abstraction layer
- The controller enables user-level administration of a dynamic system



# Puts & Gets



- Automatic replica placement and location

## Access Control

Full

## Details

Value

[illegible]

GEMS Search

[jwozniak@nd.edu](mailto:jwozniak@nd.edu)

[pbrenne1@nd.edu](mailto:pbrenne1@nd.edu)

<http://gipse.cse.nd.edu/GEMS>

Server: gems.cse.nd.edu

Account: unix:jwozniak

View: Params

Settings:

Storage

Access Control

Simple Config Full

Key	Value
app	wwd_Nov12_ClusTraj01
trial	20
direction	f
Match	Put

Preview Details

Previous

G E M S Results 1-10 of 51

[Next](#)[Download](#)[Details](#)

Owner: unix:pbrennel

Params: simname = wwd; job = 5969.0; scientist\_fname = pbrenne1;  
 app = wwd\_Nov12\_ClusTraj01; time = 03:20PM; date = 11/13/2007; segment = 19;  
 trial = 20; scientist\_lname = ; gems\_time = 3; direction = f; compute = 5325;  
 xfer = 1916604; host = loco03.cse.nd.edu  
 gems\_tps.sh (1.2 KB); gems\_clients.sh (176 B); tools.sh (546 B); jobs.db (1 B);  
 wwd.lang.conf.19 (1.3 KB); wwd.out.bin.pos.xyz.18 (116.2 KB);  
 screen.out (21.7 KB); wwd.out.bin.vel.xyz.18 (116.2 KB);  
 wwd.out.energy.header.19 (195 B); put\_stats.log (0 B); execseg.sh (7.3 KB);  
 GEMSpout.xml (2.2 KB); wwd\_mineq.psf (548.0 KB); segs.db (140 B);  
 wwd.out.dcd.19 (654.8 KB); JOB.5969.0 (0 B); wwd.lang.conf (1.3 KB);  
 wwd.out.energy.19 (19.2 KB); gems\_clients\_java.sh (3.2 KB);  
 par\_all27\_prot\_lipid.inp (141.9 KB); wwd.out.bin.pos.xyz.19 (116.2 KB);  
 gems\_clients\_gcj.sh (3.0 KB); wwd.out.bin.vel.xyz.19 (116.2 KB);  
 param.log (0 B); condor.submit (659 B)

Config key: 1033028408

[Download](#)[Details](#)

Owner: unix:pbrennel

Params: simname = wwd; job = 5969.0; scientist\_fname = pbrenne1;  
 app = wwd\_Nov12\_ClusTraj01; time = 05:50AM; date = 11/15/2007; segment = 44;  
 trial = 20; scientist\_lname = ; gems\_time = 3; direction = f; compute = 5686;  
 xfer = 1916598; host = loco03.cse.nd.edu  
 gems\_tps.sh (1.2 KB); wwd.o  
 gems\_clients.sh (176 B); to  
 put\_stats.log (0 B); execse  
 wwd.out.dcd.44 (654.8 KB);  
 wwd\_mineq.psf (548.0 KB); s  
 JOB.5969.0 (0 B); wwd.out.b  
 wwd.out.bin.vel.xyz.43 (116  
 gems\_clients\_java.sh (3.2 K  
 gems\_clients\_gcj.sh (3.0 KB); wwd.out.energy.44 (19.2 KB); param.log (0 B);  
 condor.submit (659 B)

Config key: 1056874176

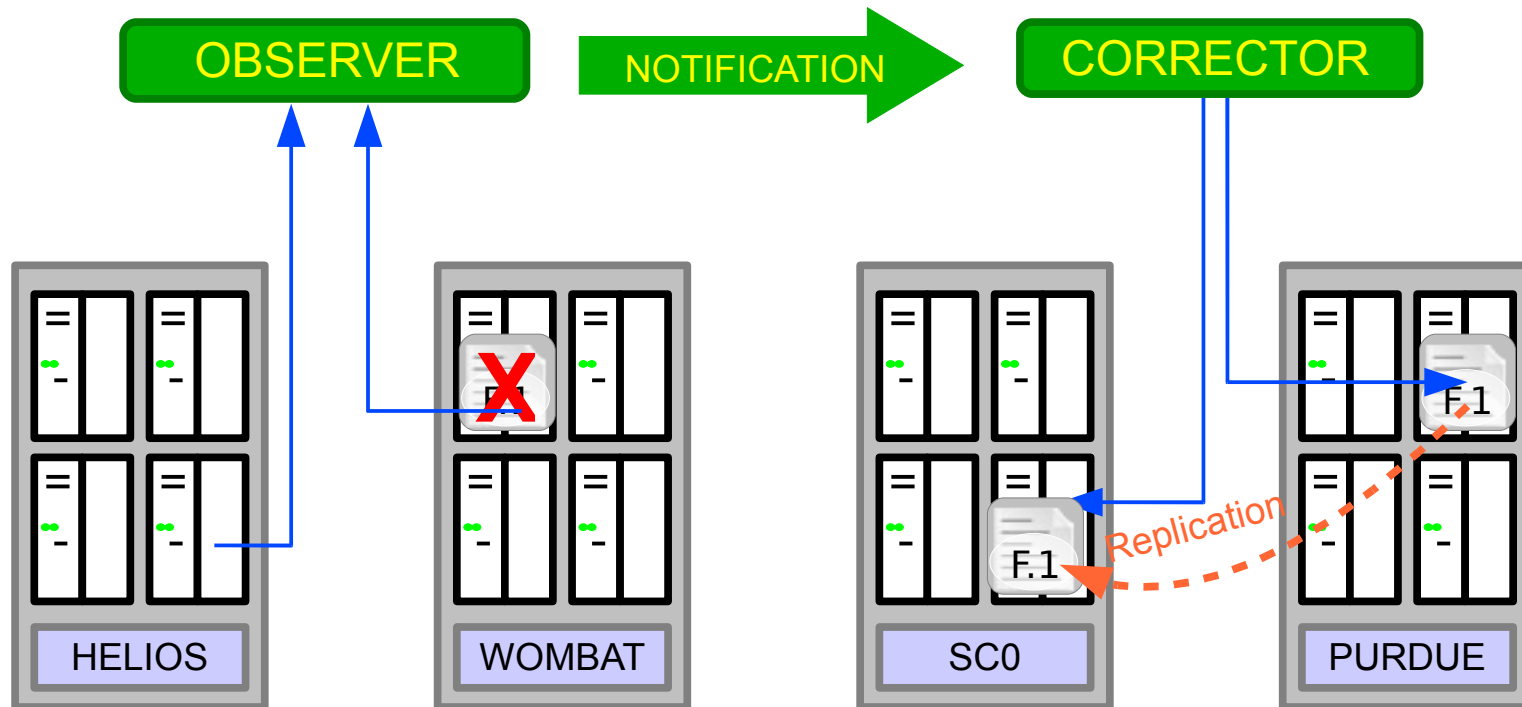
[Download](#)[Details](#)

Owner: unix:pbrennel

Params: simname = wwd; job = 5969.0; scientist\_fname = pbrenne1;  
 app = wwd\_Nov12\_ClusTraj01; time = 12:41AM; date = 11/13/2007; segment = 9;

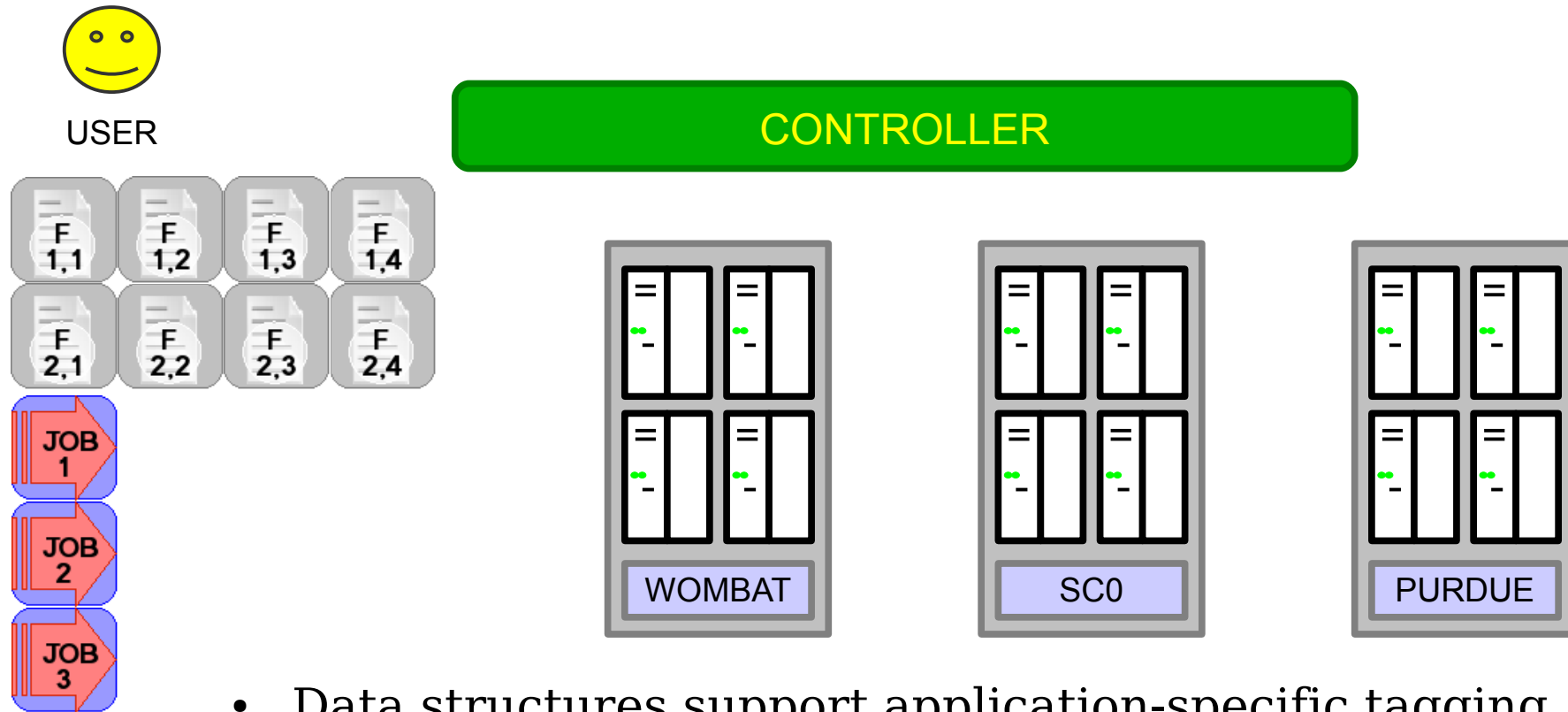
- Are these files available?

# Fault Prioritization



- Automatically detect, prioritize, and correct faults

# Parameterized Storage Organization



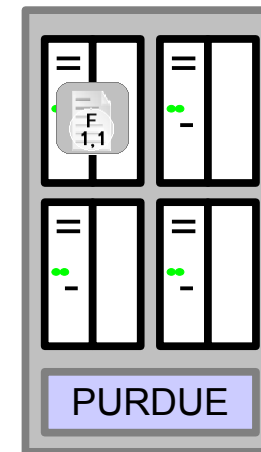
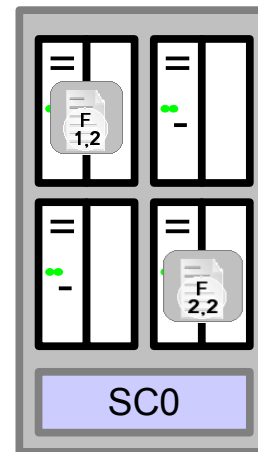
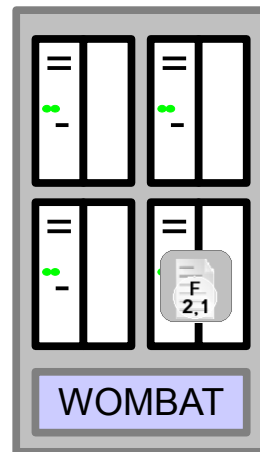
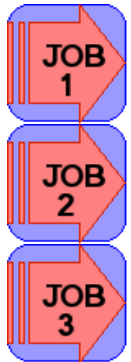
- Data structures support application-specific tagging and searches

# Parameterized Storage Organization



USER

CONTROLLER



- File placement managed by user-supplied topology information

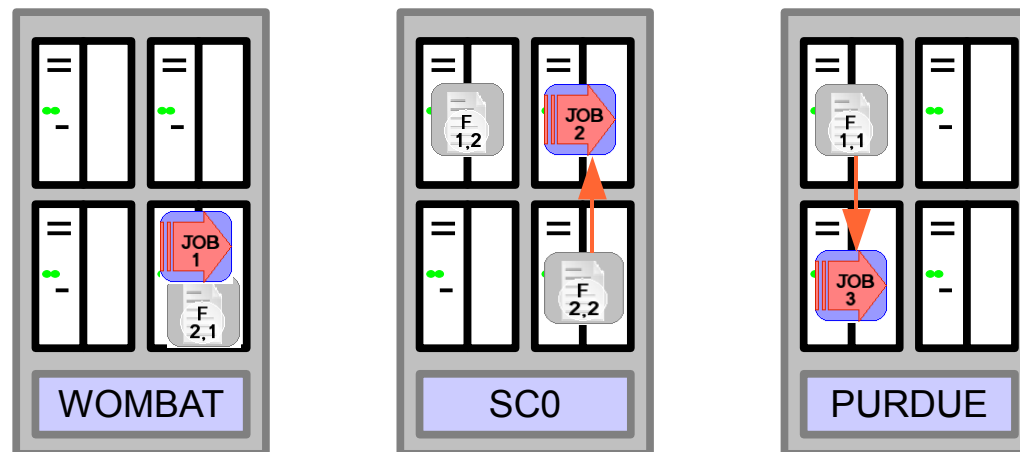


# Parameterized Storage Organization



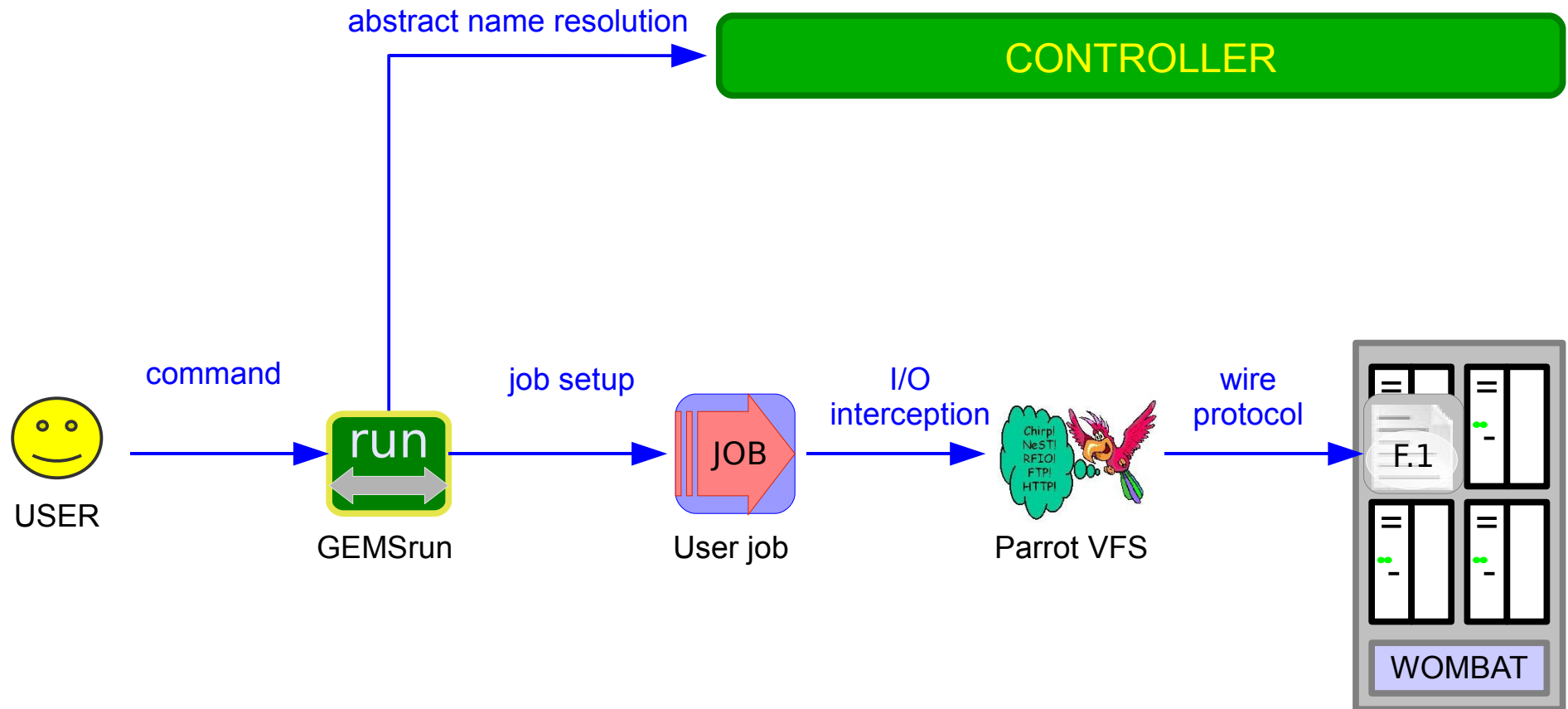
USER

CONTROLLER



- Topology information may be tapped when placing jobs or accessing replicas

# Computation *Among* Replicas



- Replica-aware computing framework

# Replica System Methods

---

---

- Basic methods:
  - Simple puts & gets
  - Replica location
  - Replica access site evaluation
- Streaming methods:
  - Advanced disk space reservation
  - I/O setup
  - I/O translation

# Replica System Examples

---

---

- Archive creation

*obtain sink:*

```
> GEMSreserve 100MB
```

*pipe command output:*

```
> tar c dir | sink
```

*or use parrot:*

```
> parrot tar cf sink dir
```

- Job I/O setup

```
> GEMSrun
```

```
--input INPUT /$KEY/file.1
```

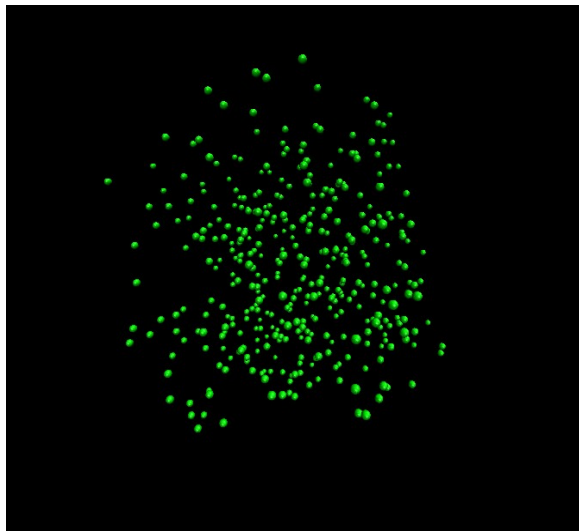
```
name=Justin job=3
```

```
--output OUTPUT file.2
```

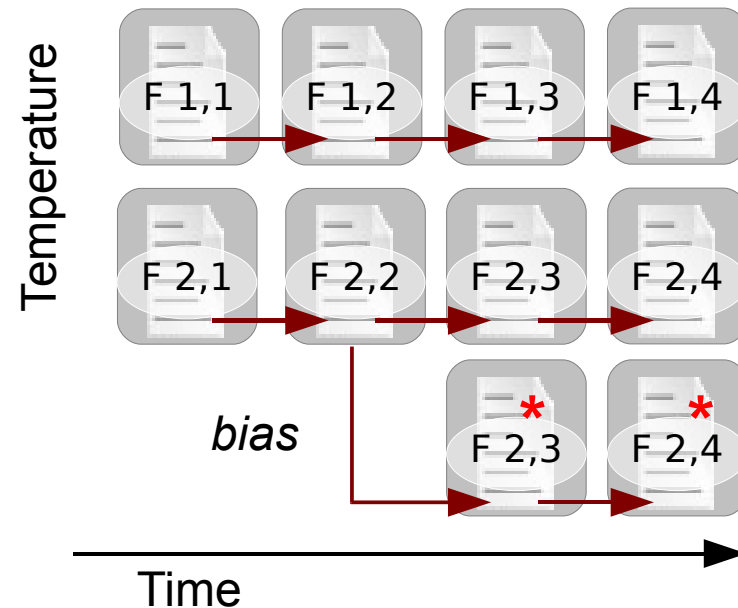
```
--exec job INPUT OUTPUT
```

- Simple shell setup for streaming replica system I/O

# Example application: Hyperdynamics



400 Argons



- Enhanced, user-steered molecular simulation technique
- Algorithm implementation enhanced by the *data sweep* abstraction

Refresh

7

8

9

10

11

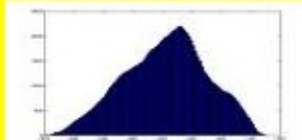
twain.helios.nd.edu  
key: 2142381621  
level: 0  
lineage: :  
compute: 10m48s  
timescale: 50000



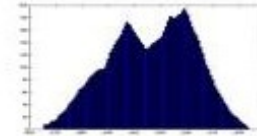
twain.helios.nd.edu  
key: 51605540  
level: 0  
lineage: :  
compute: 10m35s  
timescale: 50000



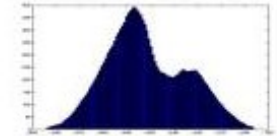
twain.helios.nd.edu  
key: 465148803  
level: 0  
lineage: :  
compute: 10m31s  
timescale: 50000



cse-cvr1-29.cse.nd.edu  
key: 1344730329  
level: 0  
lineage: :  
compute: 9m19s  
timescale: 50000



cse-cvr1-29.cse.nd.edu  
key: 412815248  
level: 0  
lineage: :  
compute: 9m17s  
timescale: 50000



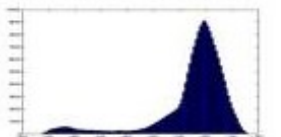
twain.helios.nd.edu  
key: 1860007973  
level: 1  
lineage: :10:  
compute: 17m16s  
timescale: 868484



twain.helios.nd.edu  
key: 1380553986  
level: 1  
lineage: :10:  
compute: 17m15s  
timescale: 761964



cvr1-ws-24.cse.nd.edu  
key: 2015789696  
level: 1  
lineage: :4:  
compute: 15m34s  
timescale: 200320



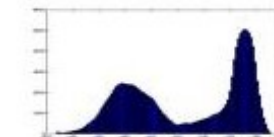
cvr1-ws-24.cse.nd.edu  
key: 1000670646  
level: 2  
lineage: :4:4:  
compute: 15m37s  
timescale: 121898



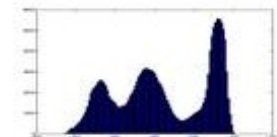
cvr1-ws-24.cse.nd.edu  
key: 285343853  
level: 2  
lineage: :4:4:  
compute: 15m44s  
timescale: 1.53157e+06



cvr1-ws-24.cse.nd.edu  
key: 1350273373  
level: 2  
lineage: :4:4:  
compute: 15m44s  
timescale: 1.64307e+06



cvr1-ws-24.cse.nd.edu  
key: 519937607  
level: 2  
lineage: :4:4:  
compute: 20m32s  
timescale: 202712





# Parameterized Workflow

- $u, r$  : Parameter sweep variables  
(e.g. user, random seed)
- Parameter sweep dependency over time:  
(checkpointing)

$$S(u, r, t) :$$

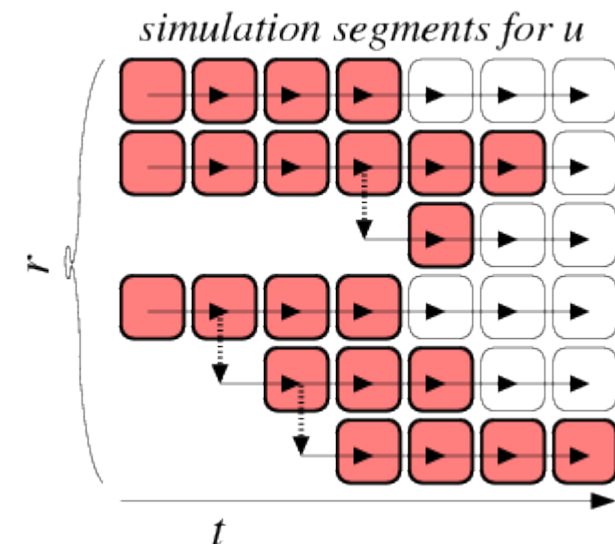
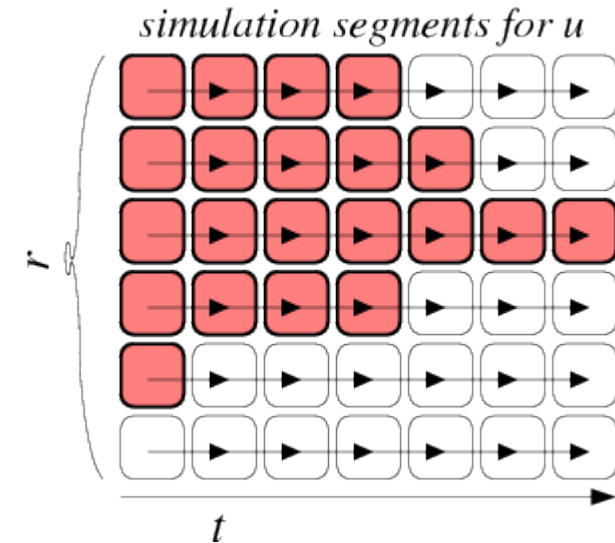
$$S(u, r, t - 1)$$

- Parameter dependency with branches:

$$S(u, r, t, \text{branch} = p) :$$

$$S(u, r, t - 1, \text{branch} = p)$$

$$\text{or } S(\text{identity} = p, \text{time} = t - 1)$$



# Notification Tools

---

---

- Workflow element script

*wait for match:*

```
> GEMSnotify user=sorin  
             r=4  t=3
```

*generate new record:*

```
> GEMSrun ...
```

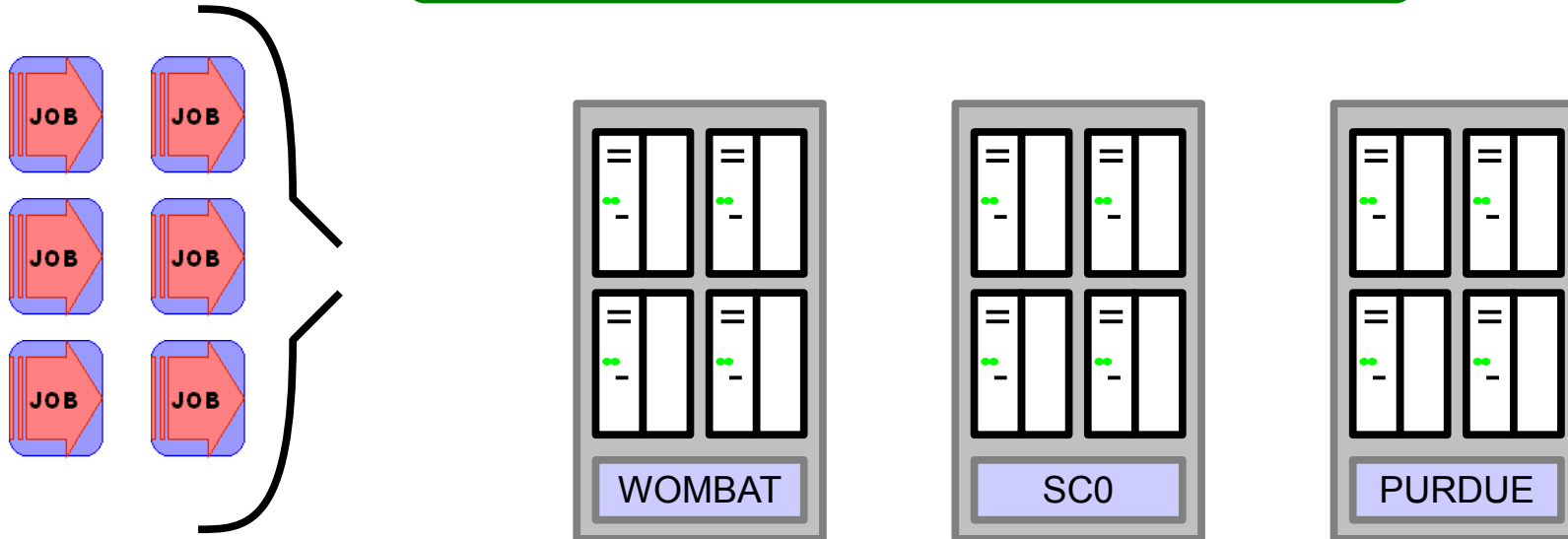
- Simply parameterize and send to background...  
(how much space is in the background?)

# Scalable Notification



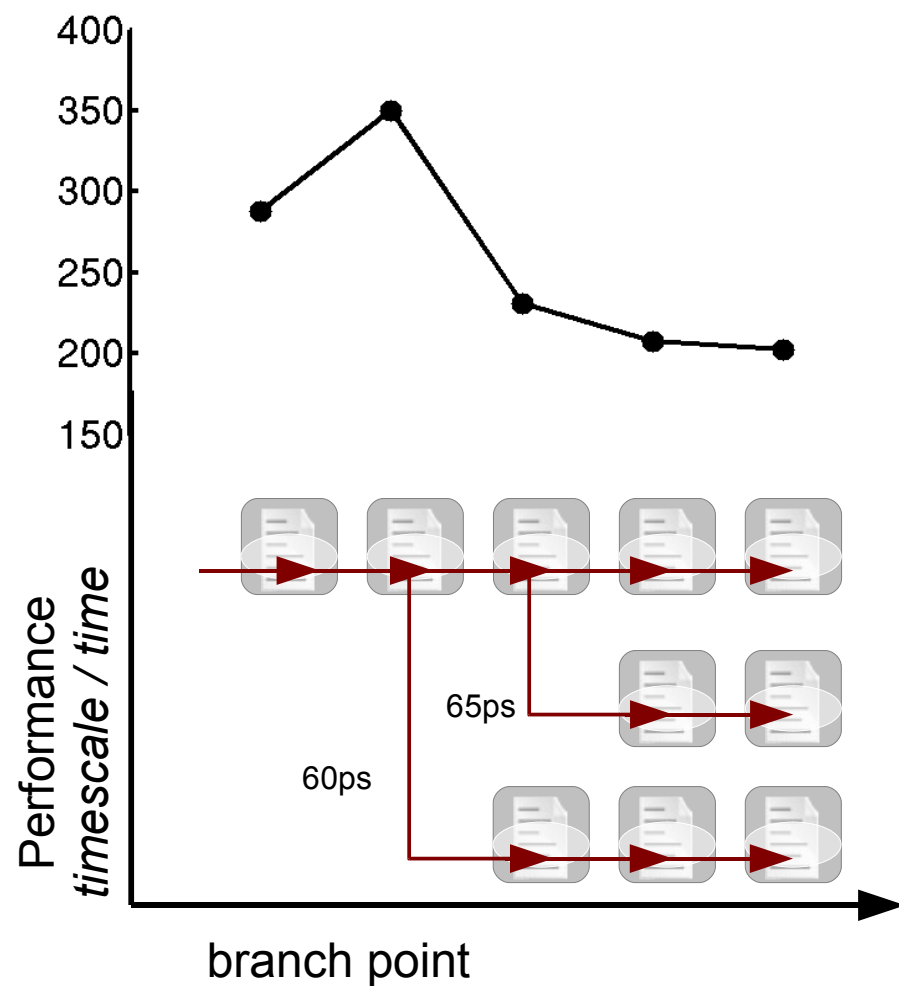
USER

CONTROLLER



- Bottlenecking procedure allows for progress while limiting consumption of system resources.

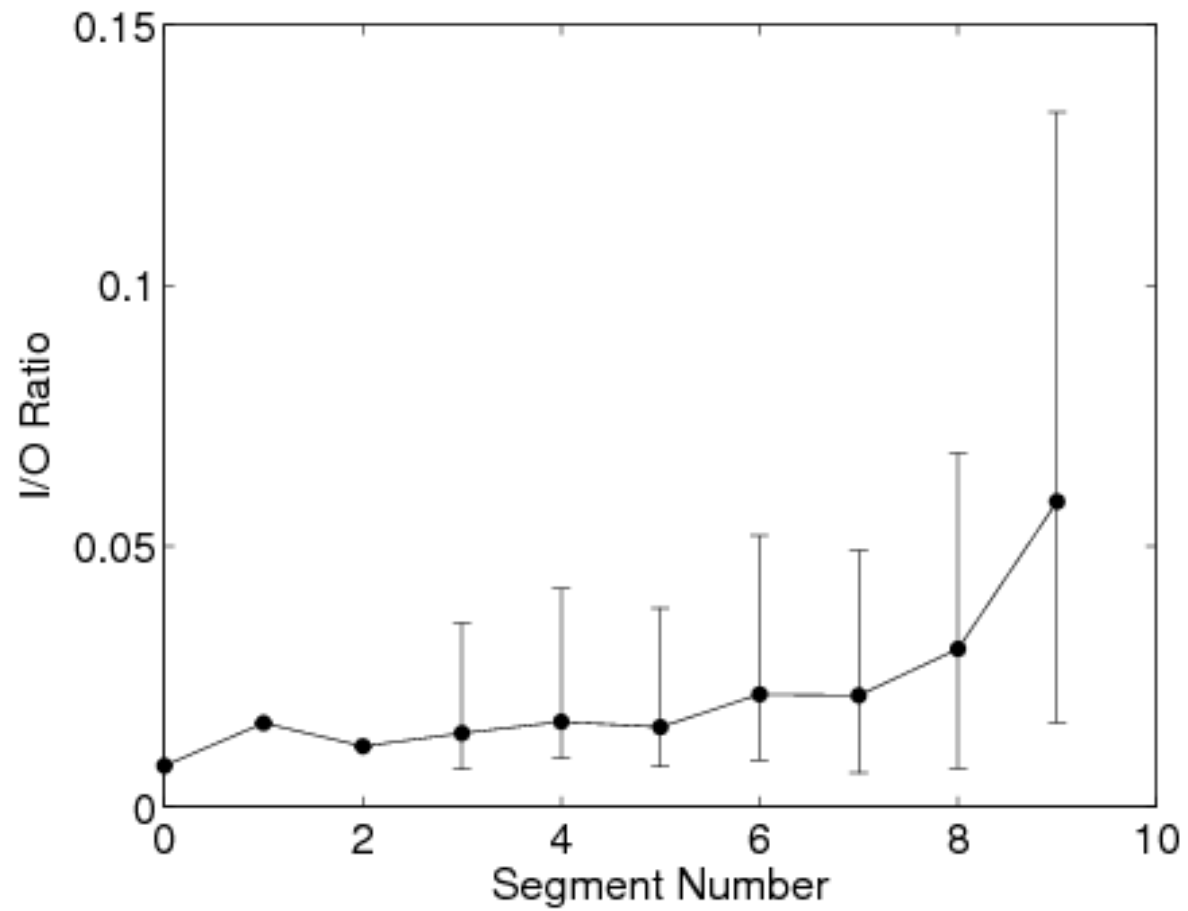
# Hyperdynamics Results



Method	Total (hours)	Turnaround (hours)
HYD-DEPTH	1.8	1.8
HYD-BREADTH	5800	1.4
HYD-EXPLORE	6.2	1.4

- Results for small simulated system (400 Argons)

# I/O Ratio



- I/O Ratio

# Recap



- I. Storage management for distributed repositories
  - Virtual clustering, puts & gets
  - Fault prioritization, control loop framework
  - Parameterized abstraction layer
  - Data access for computation
- II. Workflow model based on parameterized objects
  - Hyperdynamics application
  - Workflow formulation
  - Notification tools
  - Scalability

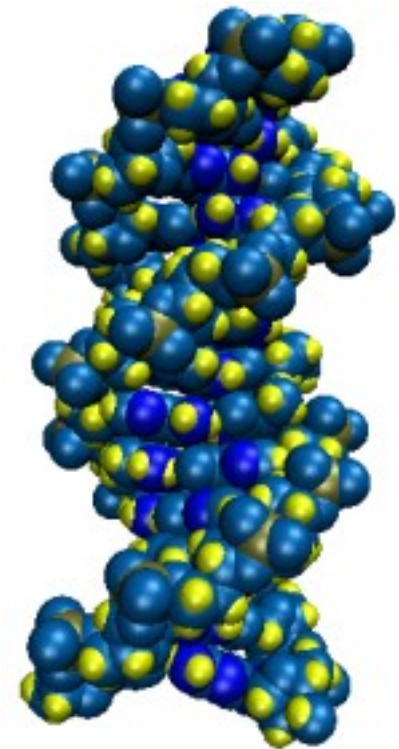


# Summary of Results

---

---

- A controller model can help users **administrate** *ad hoc* storage networks.
- Prioritized storage management can improve data durability.
- Replica management systems can be integrated within a computation infrastructure.
- Parameterized workflows can form a simple building block for distributed data operations
- GEMS is open source:  
<http://sourceforge.net/projects/gems-nd>



# Acknowledgments



- Collaborators:
  - Paul Brenner
  - Santanu Chatterjee
  - Douglas Thain
  - Aaron Striegel
  - Jesus Izaguirre
- NSF DBI-0450067

# Future Work



- Formalization of the parameter generation and arithmetic
- Real-world applications of *grid derivation*.
- Repository interoperability for hybrid systems.
- Performance analysis for more complex cases.

# Icons

- Icon workshop

