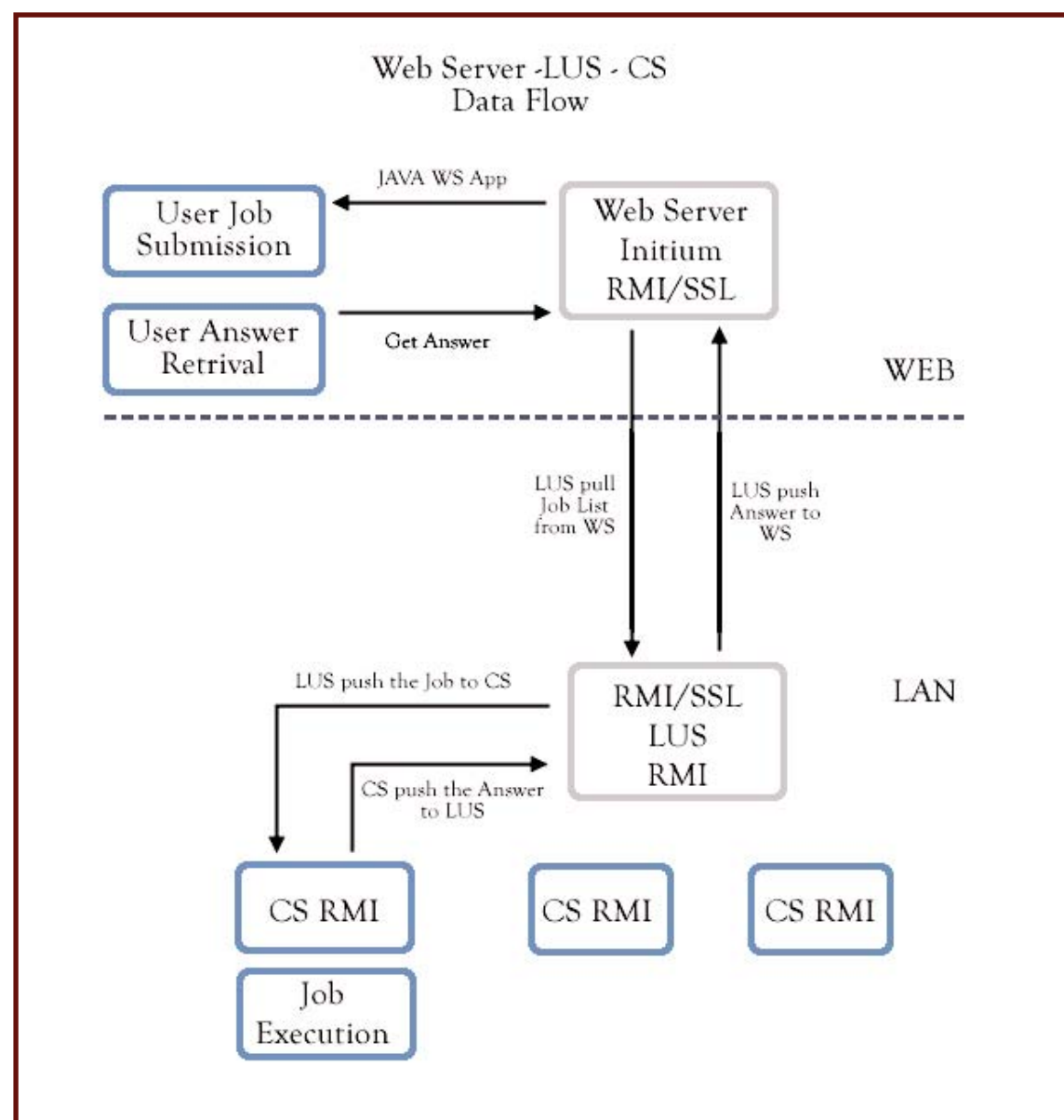


ABSTRACT

We present a framework called the Initium Remote Job Submission (RJS) system. Initium generates a jar file that is signed by a trusted certificate authority CA [Lyon]. The jar is run by a Computation Server (CS), (a remote computer running the Initium Computer Server Software). A Web Server (WS) has a Java Network Launch Protocol (JNLP) file that makes reference to a signed jar file on a web server. The signed jar file contains a job for the computation server. This jar file is called the computation jar. The computation jar is executed on the CS and the answer is sent back to the server using RMI over SSL (RMI/SSL). A look-up server (LUS) is used to register computation servers as they come on line. When a computation server is started, it registers with a LUS. The LUS then updates its list of computation servers, in the cluster. Look-up servers can be started using Java Web Start. They can be contacted with multicast protocols.



RJS data flow

THE GOAL OF THE RESEARCH

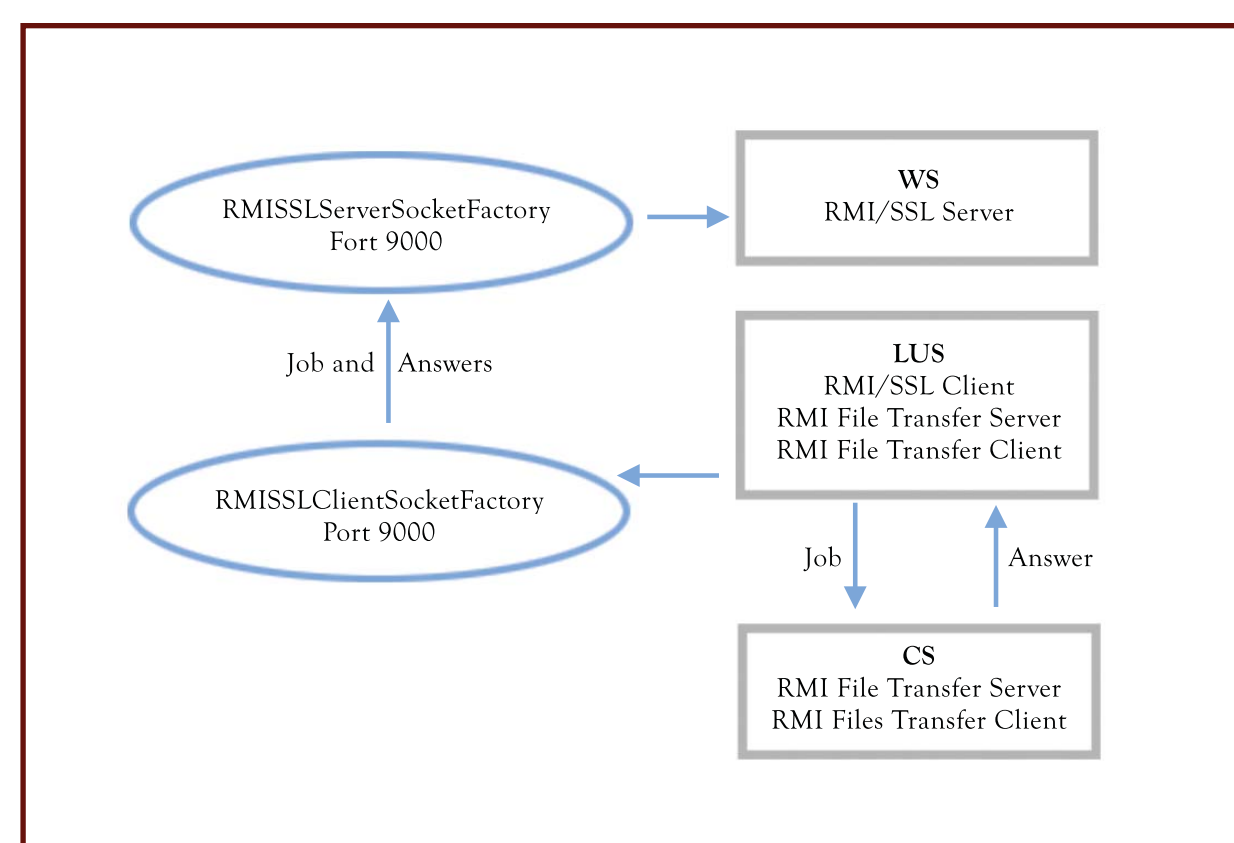
- Heterogeneous computing
- Screen Saver approach for Computation Servers
- Integrated Security
- Java Solution
- Operating system Independent
- Ability to submit jobs to the Grid

MOTIVATION

- Workstations are idle 14 or more hours per workday.
- Waste of CPU cycles, storage capacity, and network bandwidth.
- Security issues in grid computing
- Programs are safe to run on grid computers.

APPROACH

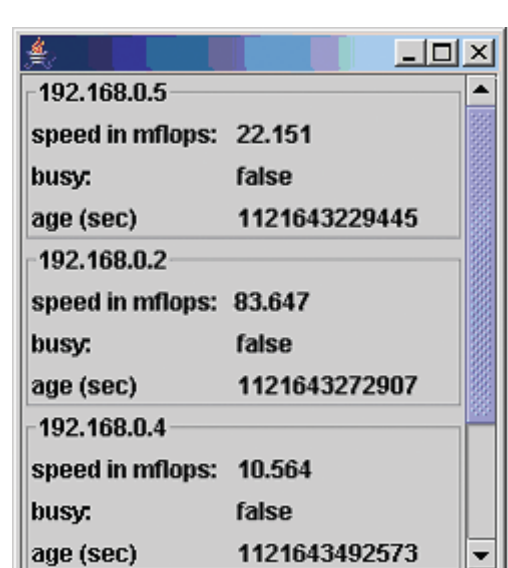
- Java Web Start
- SSL
- Web Server RMI/SSL implementation



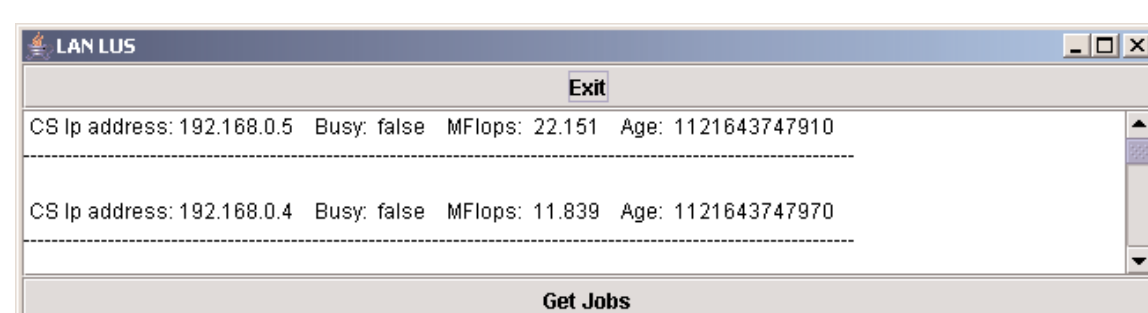
RJS RMI/SSL and RMI design

DEFINITIONS

- Web Server (WS) – jobs and answers holder
- Look up Server (LUS) – jobs and answers dispatcher
- Computation Server (CS) – job executor



The LUS Panel

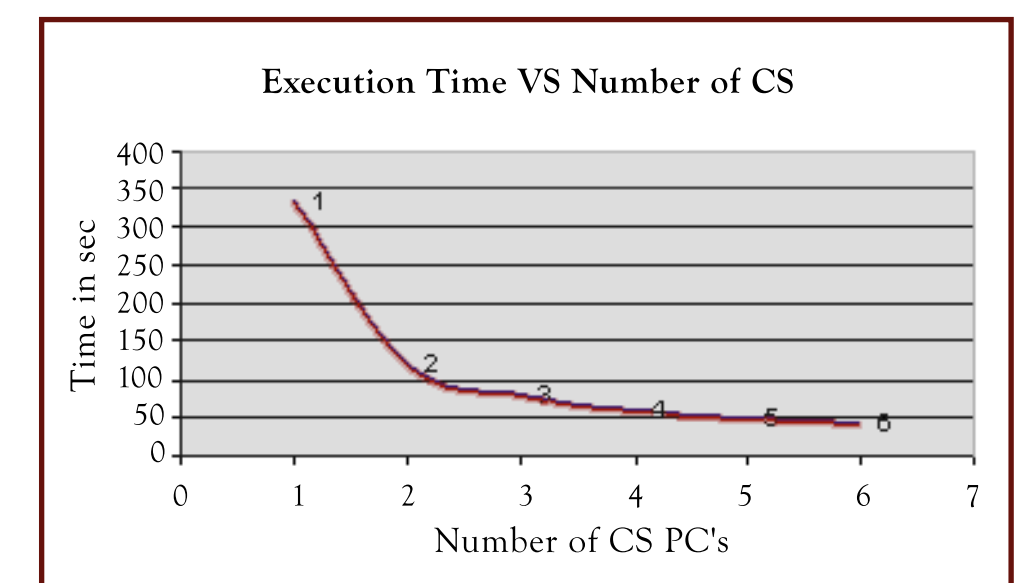


The LUS Control Panel

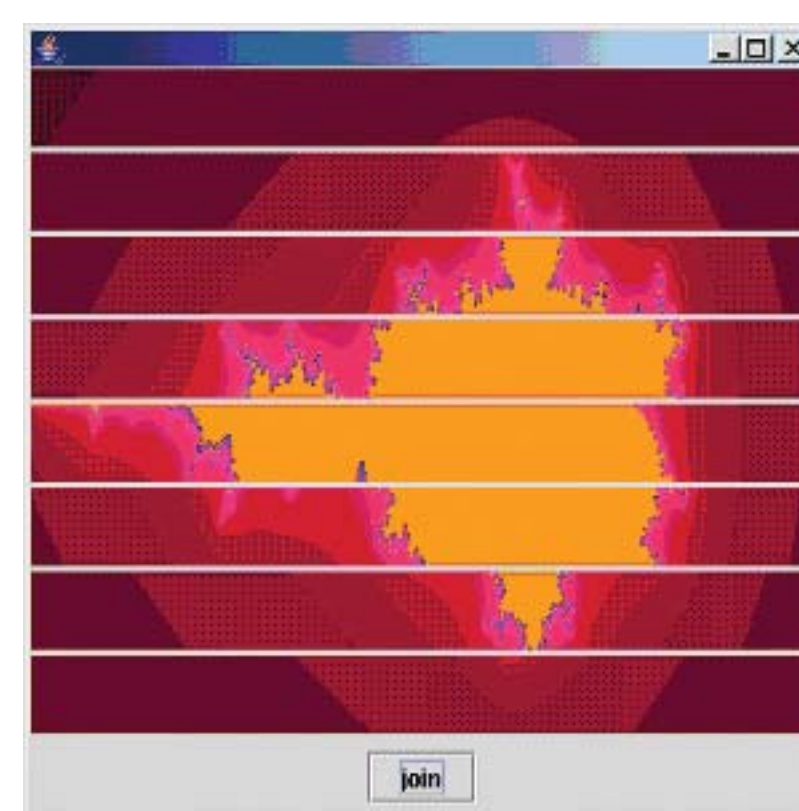
VALIDATION

RJS system was benchmarked computing one Mandelbrot task subdivided into eight jobs. After each test experiment, the system created eight answers jars. Several experiments were run using one to six computation servers.

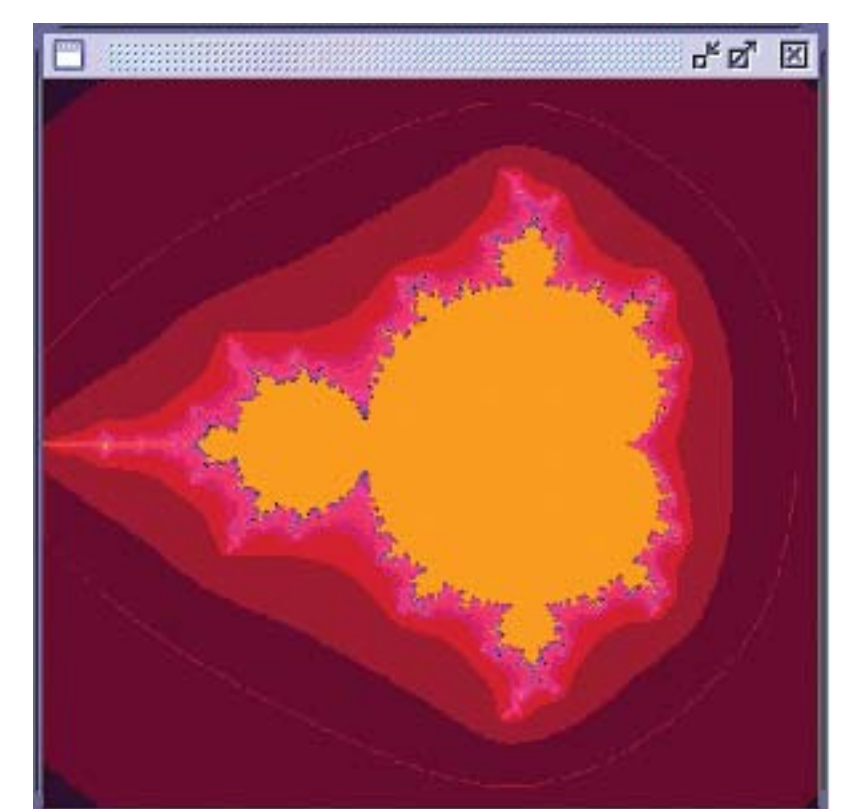
Num of CS	Num of jobs	CS OS /CPU speed	Benchmark speed In M Flops	Num of executed jobs per CS	Time (sec) to complete	AVG Time execution per job
1	8	MAC X 350 MHz	11.83	8	490	60
1	8	PC WIN XP 3.2 GHz	42.17	8	180	22.5
1	8	MAC and XP AVG	27	8	335	42
2	8	MAC X 350 MHz	14.5	2	120	15
		PC WIN XP 3.2 GHz	44.3	6		
3	8	MAC X 350 MHz	14.3	1	80	10
		PC WIN XP 900 MHz	34.33	3		
		PC WIN XP 3.2 GHz	41.24	4		
4	8	MAC X 350 MHz	14.61	1	60	7.5
		PC WIN XP 900 MHz	34.33	2		
		PC WIN XP 1 GHz	34.33	2		
		PC WIN XP 3.2 GHz	45.8	3		
5	8	MAC X 350 MHz	14.61	1	50	6.25
		PC WIN XP 900 MHz	34.33	1		
		PC WIN XP 1 GHz	34.33	1		
		PC WIN XP 1 GHz	34.33	1		
		PC WIN XP 2.4 GHz	41.2	2		
6	8	MAC X 350 MHz	14.61	1	42	5.25
		PC WIN XP 900 MHz	34.33	1		
		PC WIN XP 1 GHz	34.33	1		
		PC WIN XP 1 GHz	34.33	1		
		PC WIN XP 2.4 GHz	41.2	2		
		PC WIN XP 3.2 GHz	45.4	2		



Execution Time versus number of participating CS's



Output from the Jobs



Mandelbrot set

CONCLUSION

- CONTRIBUTIONS
- Ease of deployment
- Portability
- Security

RJS' deployment has many benefits over the grid systems on the market. Unlike many, the RJS system can be deployed and started on a new web server within minutes, through the use of Java Web Start technology. The uniqueness of the RJS lies in the portability feature. Web Start technology enables automatic deployment.

The third advantage of RJS is its security. The system is secure from the job deployment (SSH) to the job execution. All communications sent on unsecured channels is encrypted and transmitted over secured protocol (SSL). Further, the owner of the CS PC can refuse suspicious jobs by not trusting his or her certificates. Finally, the message digest algorithm verifies the integrity of the jar files before web start executes them, avoiding man-in-the-middle attacks.