

What's new in Condor? What's c

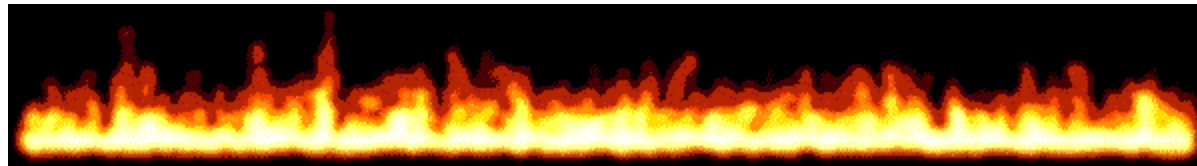
Condor Week 2010

Condor Project
Computer Sciences Department
University of Wisconsin-Madison



What's new in Condor?
What's coming up?

Condor Week 2010



Condor Project
Computer Sciences Department
University of Wisconsin-Madison

Condor Wiki

CONDOR
High Throughput Computing

[Browse] [Help] [Home] [Logout] [Milestone] [Reports] [Search] [Setup] [Ticket] [Timeline]
[Users] [Wiki]
Logged in as tannenba

Ticket #405

[Attach] [Edit] [History] [Rust]

Ticket #405: Remote IWD support for VM universe to avoid file transfer

At CondorWeek 2009, [Wipro made a nice suggestion](#) that we add remote IWD support to the VM universe to avoid file transfer.

Is it possible that this capability is already there if file transfer is not used?

Remarks: [\[Add remarks\]](#)

Properties:

Type:	enhance	Version:	7.3.0
Status:	new	Created:	2009-Apr-22 12:09
Severity:	4	Last Change:	2009-Apr-22 12:09
Priority:	2	Subsystem:	VM
Assigned To:	jfrey	Derived From:	
Creator:	tannenba	Contact:	

Done

Release Situation

- > Stable Series
 - **Current: Condor v7.4.2 (April 6th 2010)**
 - Last Year: Condor v7.2.2 (April 14th 2009)
- > Development Series
 - **Current: Condor v7.5.1 (March 2 2010)**
 - v7.5.2 "any day"
 - Last Year : Condor v7.3.0 (Feb 24th 2009)
- > How long is development taking?
 - v6.9 Series : 18 months
 - v7.1 Series : 12 months
 - v7.3 Series : 8 months

Coming Soon!

Ports

- > Short Version
 - We dropped HPUX 11/PA-RISC in v7.5
- > Long version...

Ports on the Web

condor-7.5.1-Windows-dynamic.tar.gz
condor-7.5.1-MacOSX10.4-x86-dynamic.tar.gz
condor-7.5.1-aix5.2-aix-dynamic.tar.gz
condor-7.5.1-linux-PPC-sles9-dynamic.tar.gz
condor-7.5.1-linux-PPC-yd50-dynamic.tar.gz
condor-7.5.1-linux-ia64-rhel3-dynamic.tar.gz
condor-7.5.1-linux-x86-debian40-dynamic.tar.gz
condor-7.5.1-linux-x86-debian50-dynamic.tar.gz
condor-7.5.1-linux-x86-rhel3-dynamic.tar.gz
condor-7.5.1-linux-x86-rhel5-dynamic.tar.gz
condor-7.5.1-linux-x86_64-debian50-dynamic.tar.gz
condor-7.5.1-linux-x86_64-rhel3-dynamic.tar.gz
condor-7.5.1-linux-x86_64-rhel5-dynamic.tar.gz
condor-7.5.1-solaris29-Sparc-dynamic.tar.gz



Other (better?) choices

- > Improved Packaging
 - www.cs.wisc.edu/condor/yum
 - www.cs.wisc.edu/condor/debian
- > Go native!
 - Fedora, RedHat MRG, Ubuntu
- > Go Rocks w/ Condor Roll!
- > VDT (client side)

No Tarballs!



Ports not on Web but known to work

solaris 5.8 sun4u

suse 10.2 x86

suse 10.0 x86

suse 9 ia64

suse 9 x86_64

suse 9 x86

macosx 10.4 ppc

opensolaris 2009.06 x86_64

Very easy to build anywhere if "clipped"

```
% ./configure --disable-proper --without-globus --without-krb5 --disable-full-port --without-voms --without-srb --without-hadoop --without-postgresql --without-curl --disable-quill --disable-gcc-version-check --disable-glibc-version-check --without-gsoap --without-glibc --without-cream --without-openssl
```

See "Building Condor On Unix" page at
<http://wiki.condorproject.org>

Big new goodies in v7.2

- Job Router
- Startd and Job Router hooks
- DAGMan tagging and splicing
- Green Computing started
- GLEXEC
- Concurrency Limits

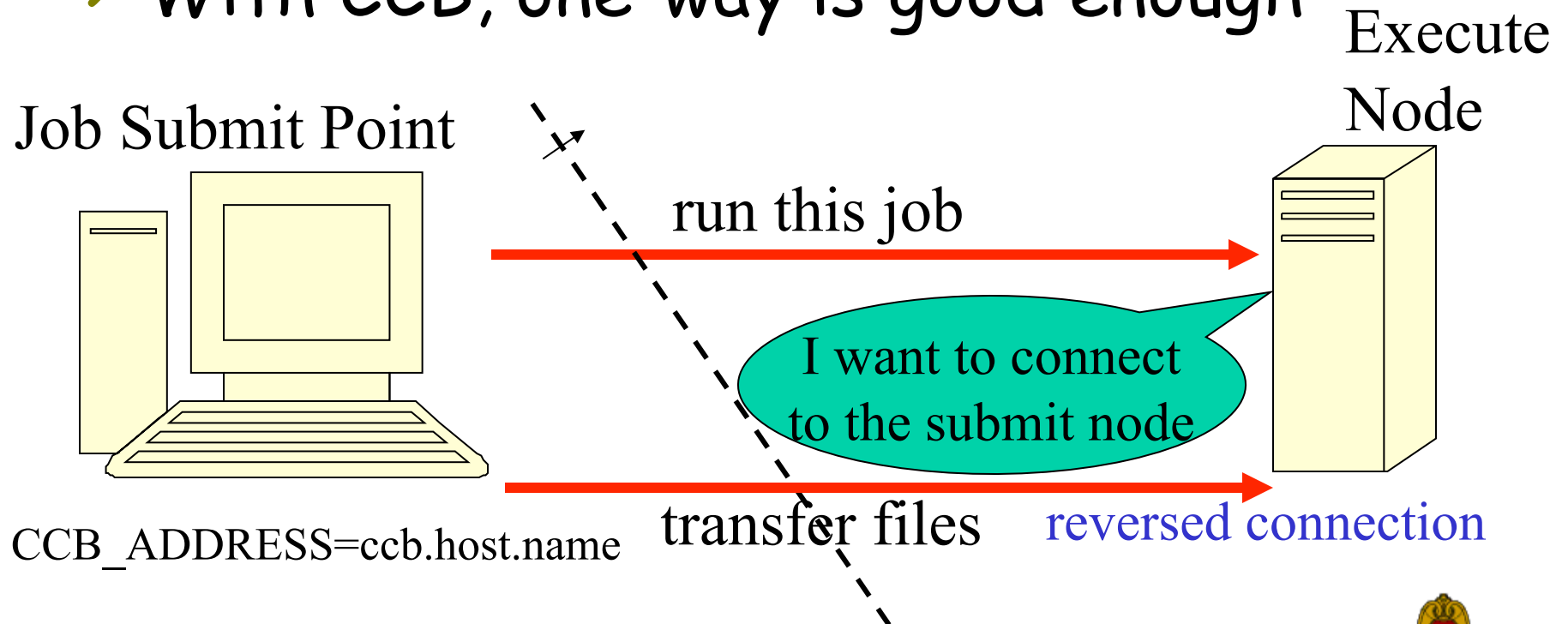
Big new goodies in v7.4

- > Scalability, stability
- > CCB
- > Grid Universe enhancements
- > Green Computing evolution
- > *condor_ssh_to_job*
- > CPU Affinity



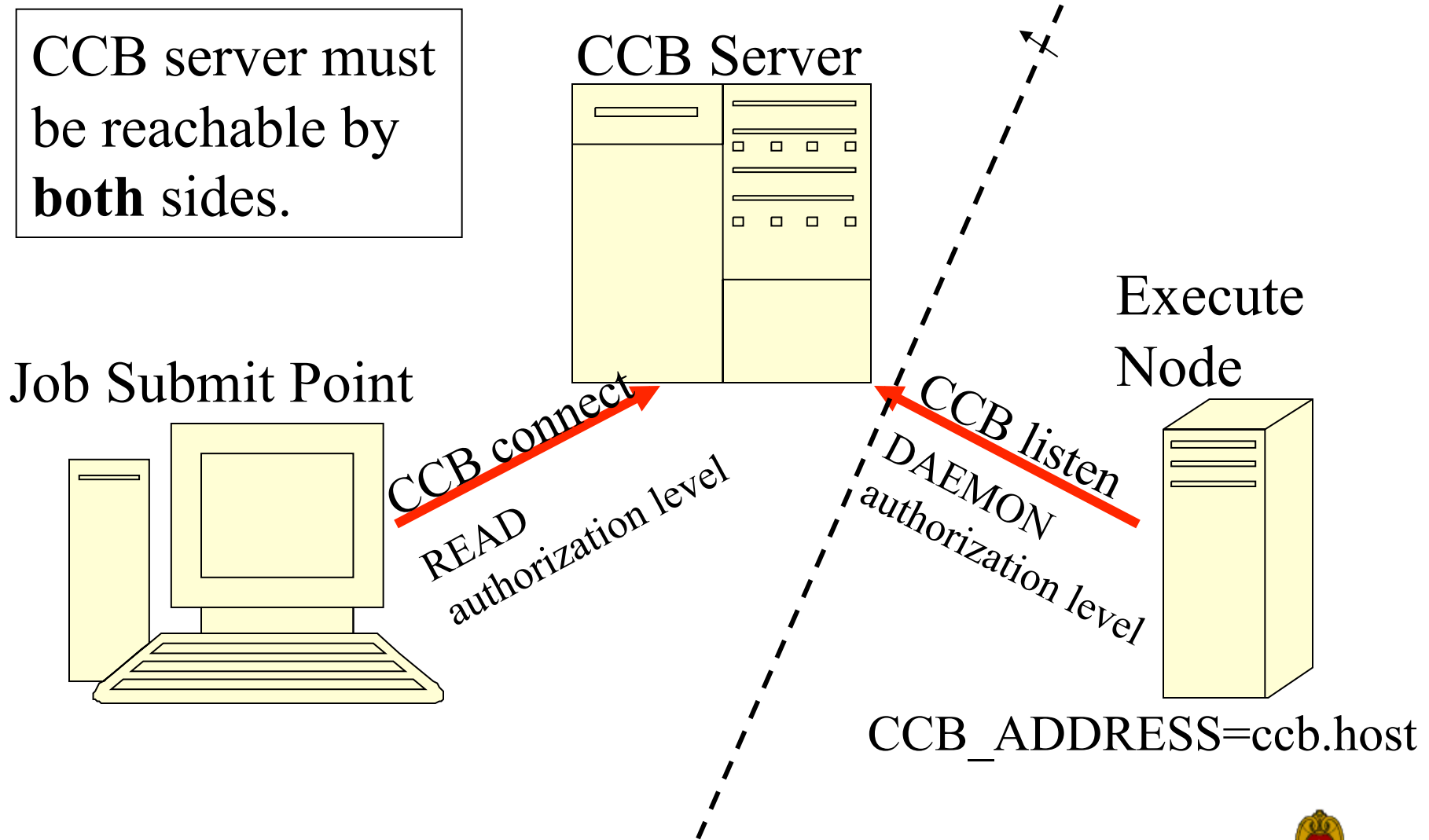
CCB: Condor Connection Broker

- > Condor wants two-way connectivity
- > With CCB, one-way is good enough



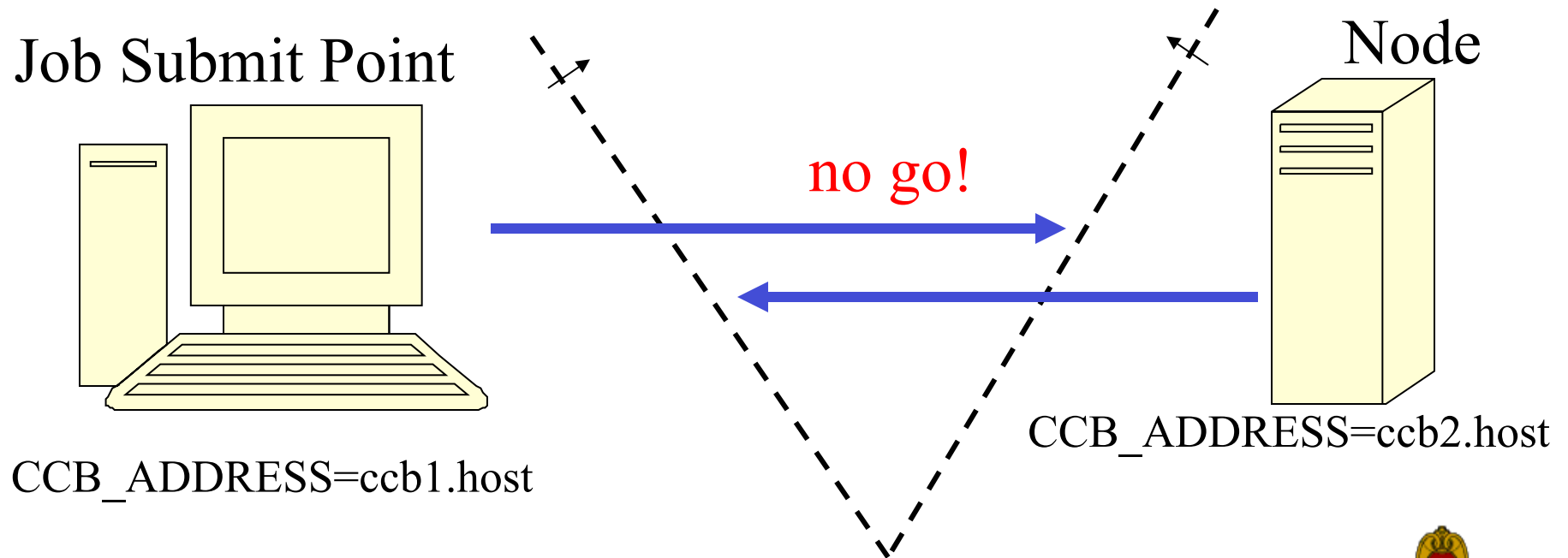
Connecting to CCB

CCB server must be reachable by **both** sides.



Limitations of CCB

1. Doesn't help with standard universe
2. Requires one-way connectivity



Why CCB?

- > Secure
 - supports full Condor security set
- > Robust
 - supports reconnect, failover
- > Portable
 - supports all Condor platforms, not just Linux

Why CCB?

- > Dynamic
 - CCB clients and servers configurable without restart
- > Informative log messages
 - Connection errors are propagated
 - Names and local IP addresses reported (GCB replaces local IP with broker IP)
- > Easy to configure
 - automatically switches UDP to TCP in Condor protocols
 - CCB server only needs one open port

Configuring CCB

> The Server:

- The collector *is* a CCB server
- UNIX: MAX_FILE_DESCRIPTOR=10000

> The Client:



1. CCB_ADDRESS = \$(COLLECTOR_HOST)
2. PRIVATE_NETWORK_NAME = your.domain

(optimization: hosts with same network name don't use CCB to connect to each other)



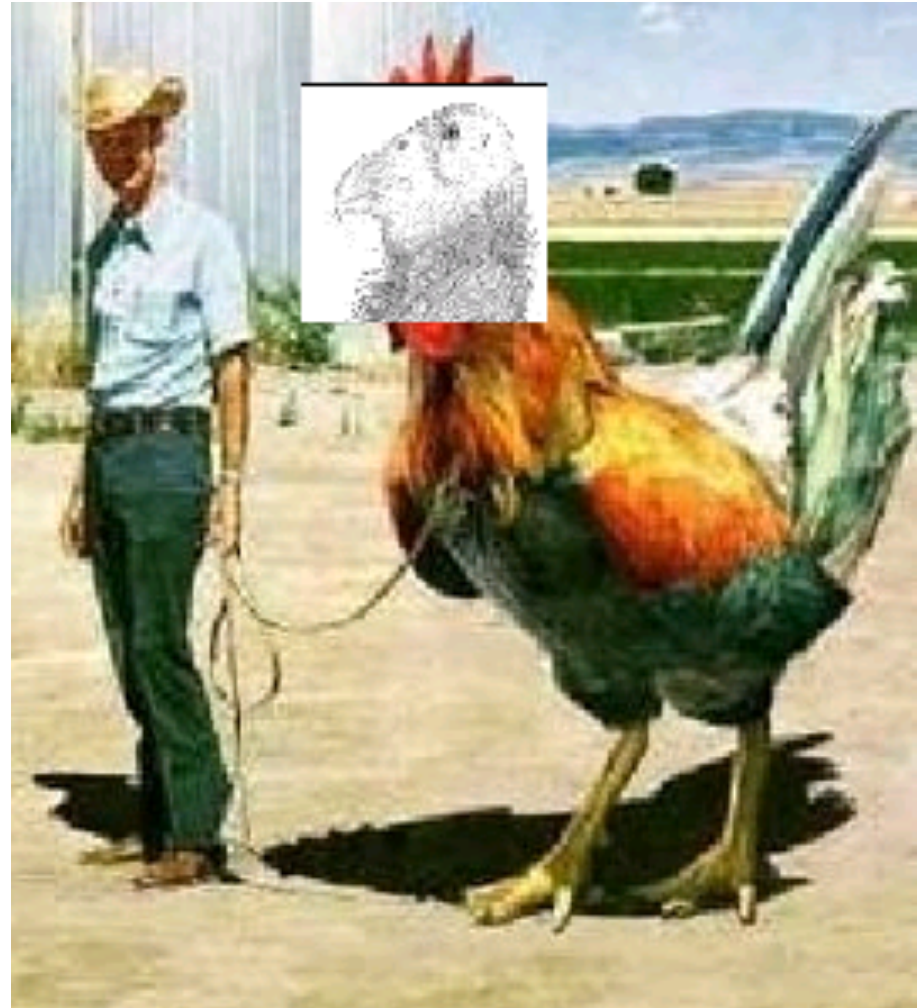
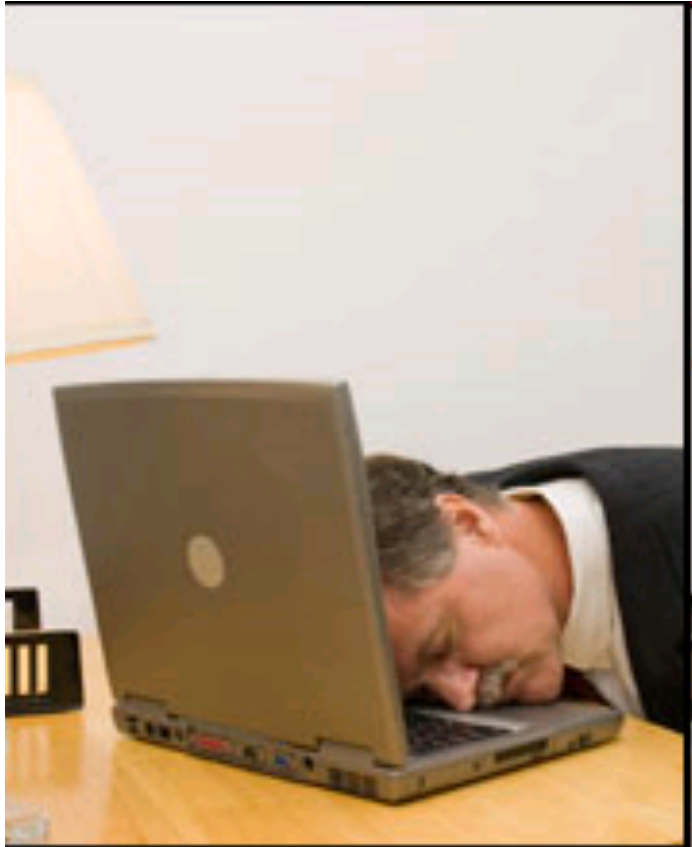
Grid Universe

- > v7.4: Added GT5 and Cream (Igor's talk)
- > v7.5 Improvements
 - Batching Commands
 - Pushing Data to Cream
 - DeltaCloud grid type

Green Computing

- The startd has the ability to place a machine into a low power state. (Standby, Hibernate, Soft-Off, etc.)
 - **HIBERNATE, HIBERNATE_CHECK_INTERVAL**
 - If all slots return non-zero, then the machine can be powered down via condor_power hook
 - A final acked classad is sent to the collector that contains wake-up information
- Machines ads in "Offline State"
 - Stored persistently to disk
 - Ad updated with "demand" information: if this machine was around, would it be matched?

Now what?



condor_rooster

- > Periodically wake up based on ClassAd expression (Rooster_UnHibernate)
- > Throttling controls
- > Hook callouts make for interesting possibilities...

Interactive Debugging

- Why is my job still running?
Is it stuck accessing a file?
Is it in an infinite loop?
- `condor_ssh_to_job`
 - Interactive debugging in UNIX
 - Use `ps`, `top`, `gdb`, `strace`, `lsof`, ...
 - Forward ports, X, transfer files, etc.

condor_ssh_to_job Example

```
% condor_q
```

```
-- Submitter: perdita.cs.wisc.edu : <128.105.165.34:1027> :  
ID      OWNER      SUBMITTED  RUN_TIME  ST PRI SIZE CMD  
1.0     einstein   4/15 06:52 1+12:10:05 R 0  10.0 cosmos
```

```
1 jobs; 0 idle, 1 running, 0 held
```

```
% condor_ssh_to_job 1.0
```

```
Welcome to slot4@c025.chtc.wisc.edu!  
Your condor job is running with pid(s) 15603.
```

```
$ gdb -p 15603
```

```
...
```



How it works

- > ssh keys created for each invocation
- > ssh
 - Uses OpenSSH ProxyCommand to use connection created by ssh_to_job
- > sshd
 - runs as same user id as job
 - receives connection in inetd mode
 - So nothing new listening on network
 - Works with CCB and shared_port

What?? Ssh to my worker nodes??

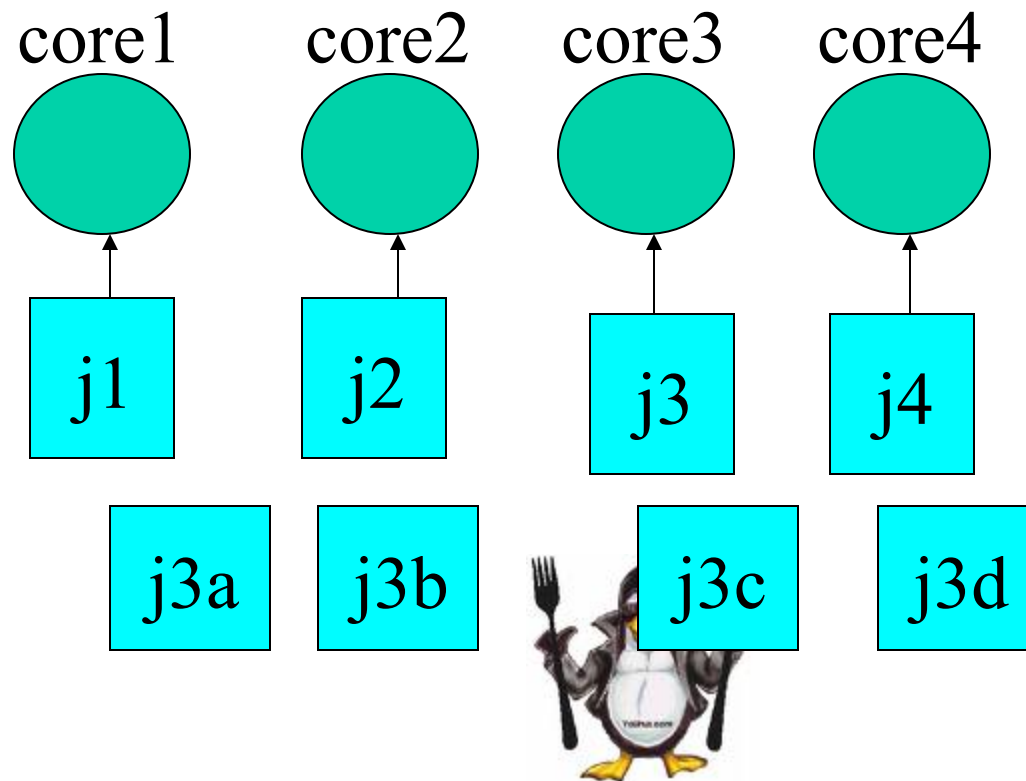
- > Why would any sysadmin allow this?
- > Because the process tree is managed
 - Cleanup at end of job
 - Cleanup at logout
- > Can be disabled by nonbelievers



CPU Affinity

Four core Machine

running four jobs w/o affinity





CPU Affinity to the rescue

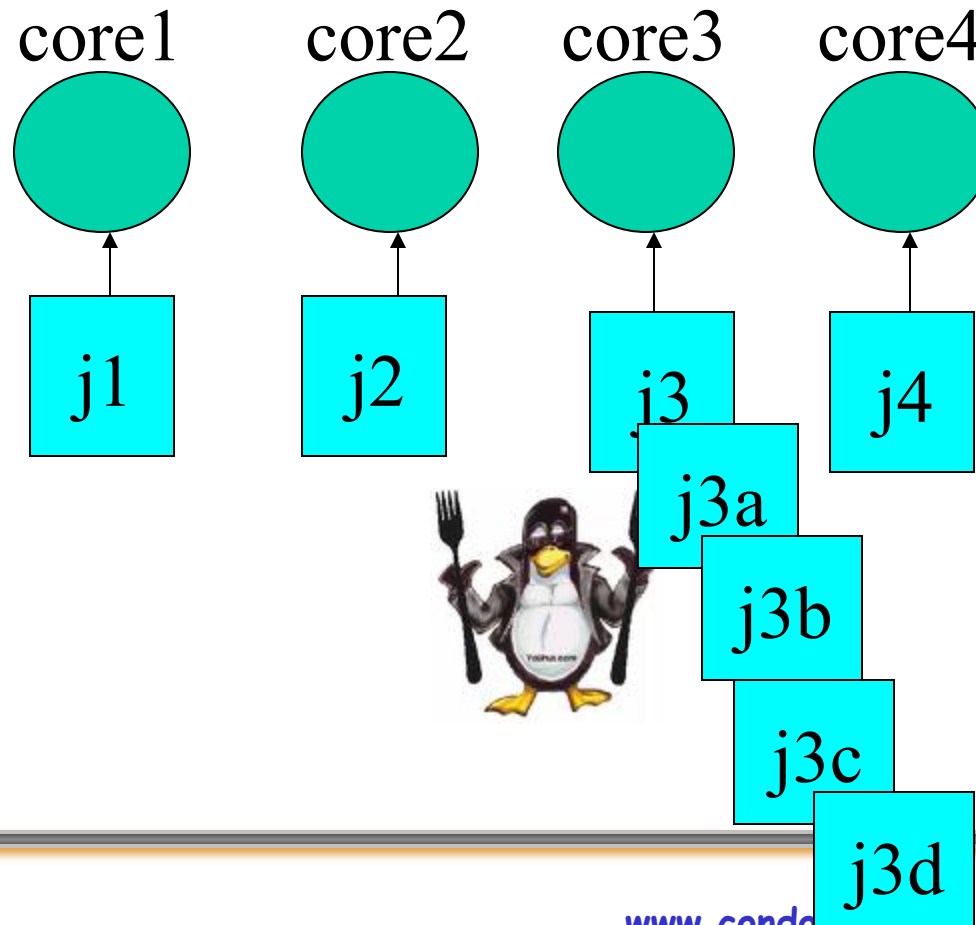
SLOT1_CPU_AFFINITY = 0

SLOT2_CPU_AFFINITY = 1

SLOT3_CPU_AFFINITY = 2

SLOT4_CPU_AFFINITY = 3

Four core Machine running four jobs w/affinity





Please read the following license agreement. Use the scrollbar to read the rest of the agreement.

Terms of License

Any and all dates in these slides are relative from a date hereby *unspecified* in the event of a likely situation involving a frequent condition. Viewing, use, reproduction, display, modification and redistribution of these slides, with or without modification, in source and binary forms, is permitted only after a deposit by said user into PayPal accounts registered to Todd Tannenbaum

....

Do you accept all the terms of the preceding license agreement? If so, click on the Yes push button. If you select No, setup will close.

< Back

Yes

No

Some already mentions...



- > Condor-G improvements (John, Igor)
- > HDFS and Hadoop (Greg)
- > DMTCP (Gene)
- > Scalability (Matt)
- > IPv6 (MinJae)
- > Enterprise Messaging (Vidhya)
- > Plugins, Hooks, and Toppings (Todd)

And non-mentions

- > VOMs
- > DAGMan improvements
 - Automatic execution of rescue DAGs
 - Automatic generation of submit files for nested DAGs

Condor "Snow Leopard"



Some Snow-Leopard Work

- > Easier/faster to build
- > Much work in improving the test suite
 - Easier to make tests
 - Different types of tests
- > Scratch some long-running itches, carry some long-running efforts over the finish line, such as...



Network Port Usage

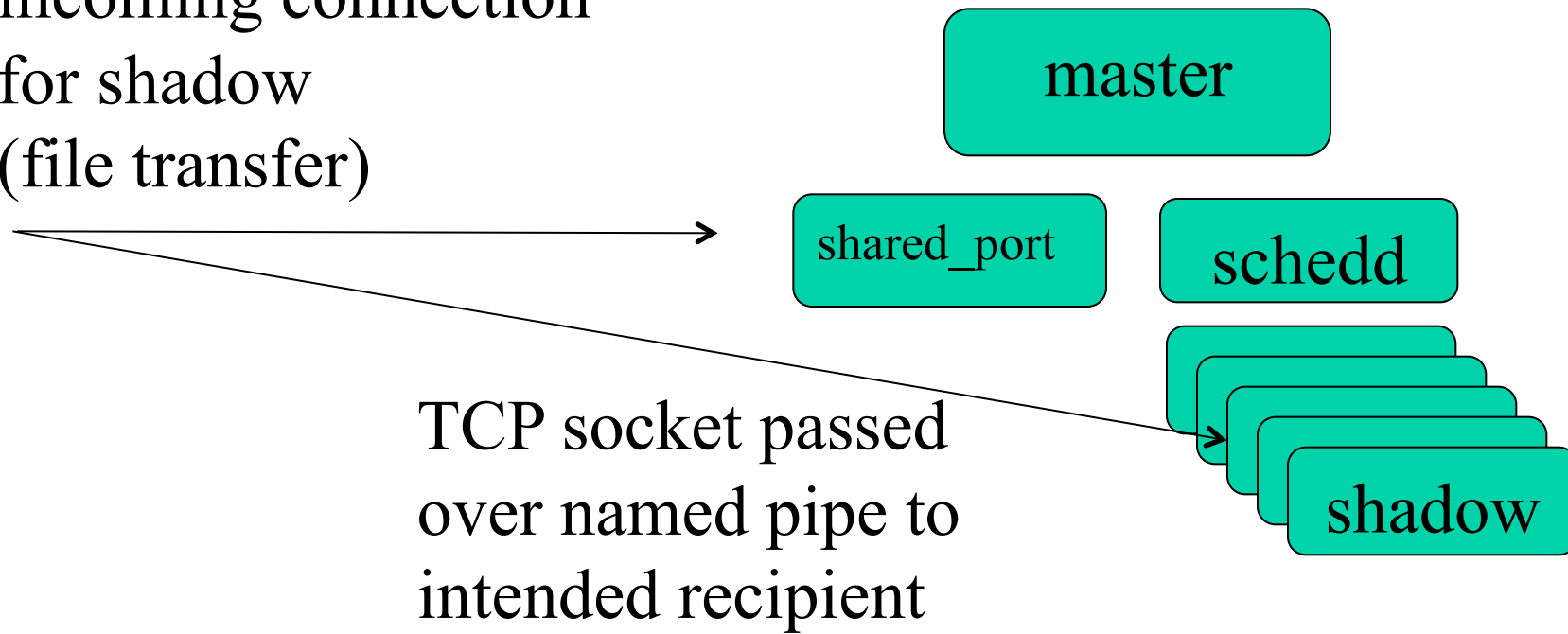
- Condor needs a lot of open network ports for incoming connections
 - Schedd: $5 + 5 * \text{NumRunningJobs}$
 - Startd: $5 + 5 * \text{NumSlots}$
- Not a pleasant firewall situation.
- CCB can make the schedd or the startd (but not both) turn these into outgoing ports instead of incoming

Have Condor listen on just *one* port per machine



How it works

incoming connection
for shadow
(file transfer)





condor_shared_port

- All daemons on a machine can share one incoming port
 - Simplifies firewall or port forwarding config
 - Improves scalability
 - Running now on Unix, Windows support coming

```
USE_SHARED_PORT = True
```

```
DAEMON_LIST = ... SHARED_PORT
```

From CondorWeek 2003:

- > New version of ClassAds into Condor
 - Conditionals !!
 - if/then/else
 - Aggregates (lists, nested classads)
 - Built-in functions
 - String operations, pattern matching, time operators, unit conversions
 - Clean implementations in C++ and Java
 - ClassAd collections
- > This may become v6.8.0



Is this TODD ???!

New ClassAds are now Condor!

- > Library in v7.5 / v7.6
 - Nothing user visible changes (we hope)
- > Take advantage of it in next dev series (v7.7)

Logging in Condor

What's there?



Daemon

d Logs

Logging in Condor

The bad news...

- Different APIs
- Different formats
- Therefore: Different behavior (and also: different bugs)
- Too many different files for different purposes referred to as "logs" (journaling, resource usage,...)

Logging in Condor

Goals?

- Unified log file locking (no more problems with shared FS)
- More unified formats and tracking of lost information due to rotation
- Cleaning up the naming convention (ideas welcome!)
 - Schedd Event Log, Job Event Log, Schedd Journal, Negotiator Journal, Daemon Logs

Condor “AddOns”

Already heard about Condor_QPid
from Vidhya yesterday...

Others? Mike talked about the “Slave
Launcher”...

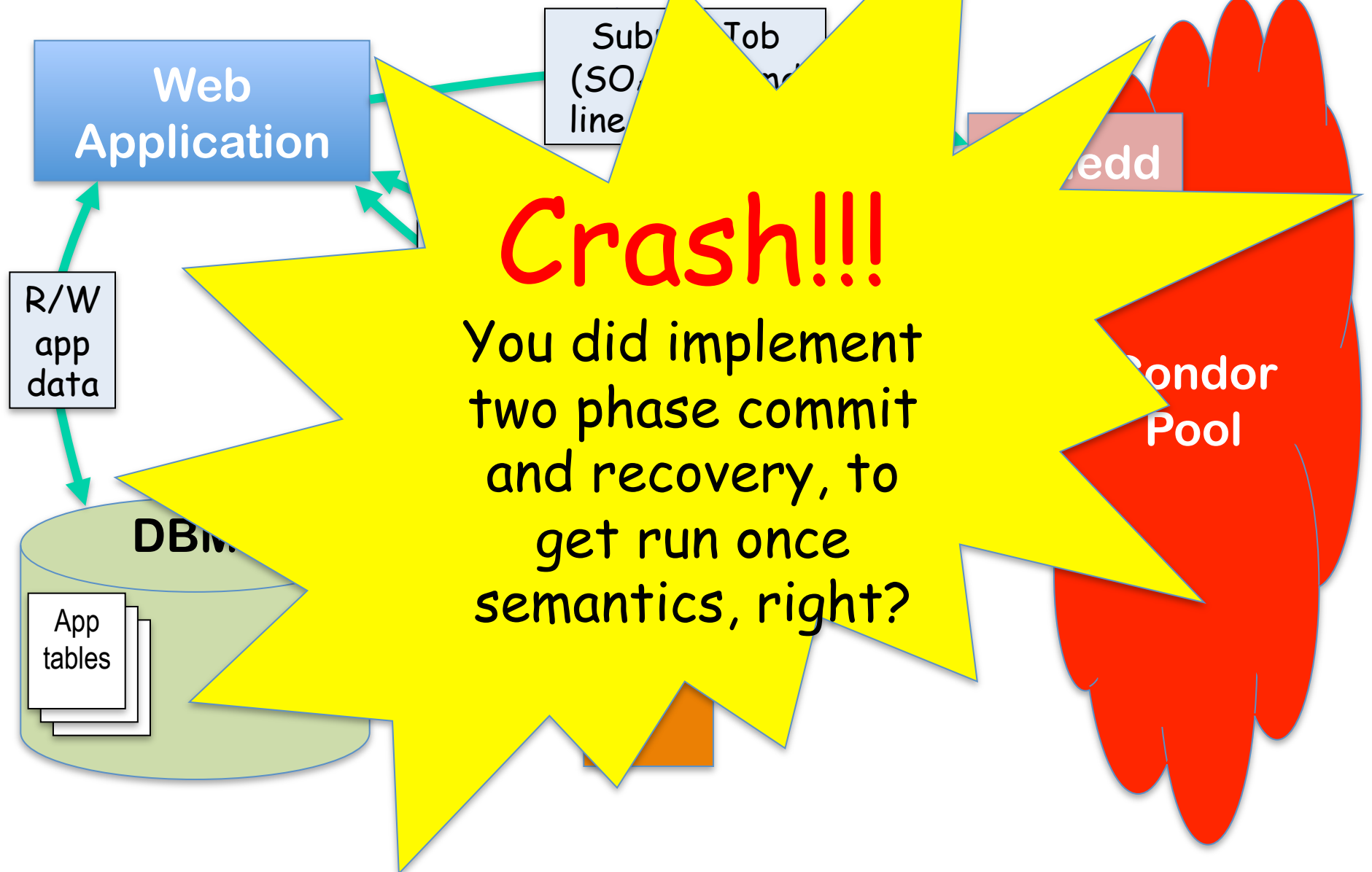
Condor Database Queue

Or
condor_dbq

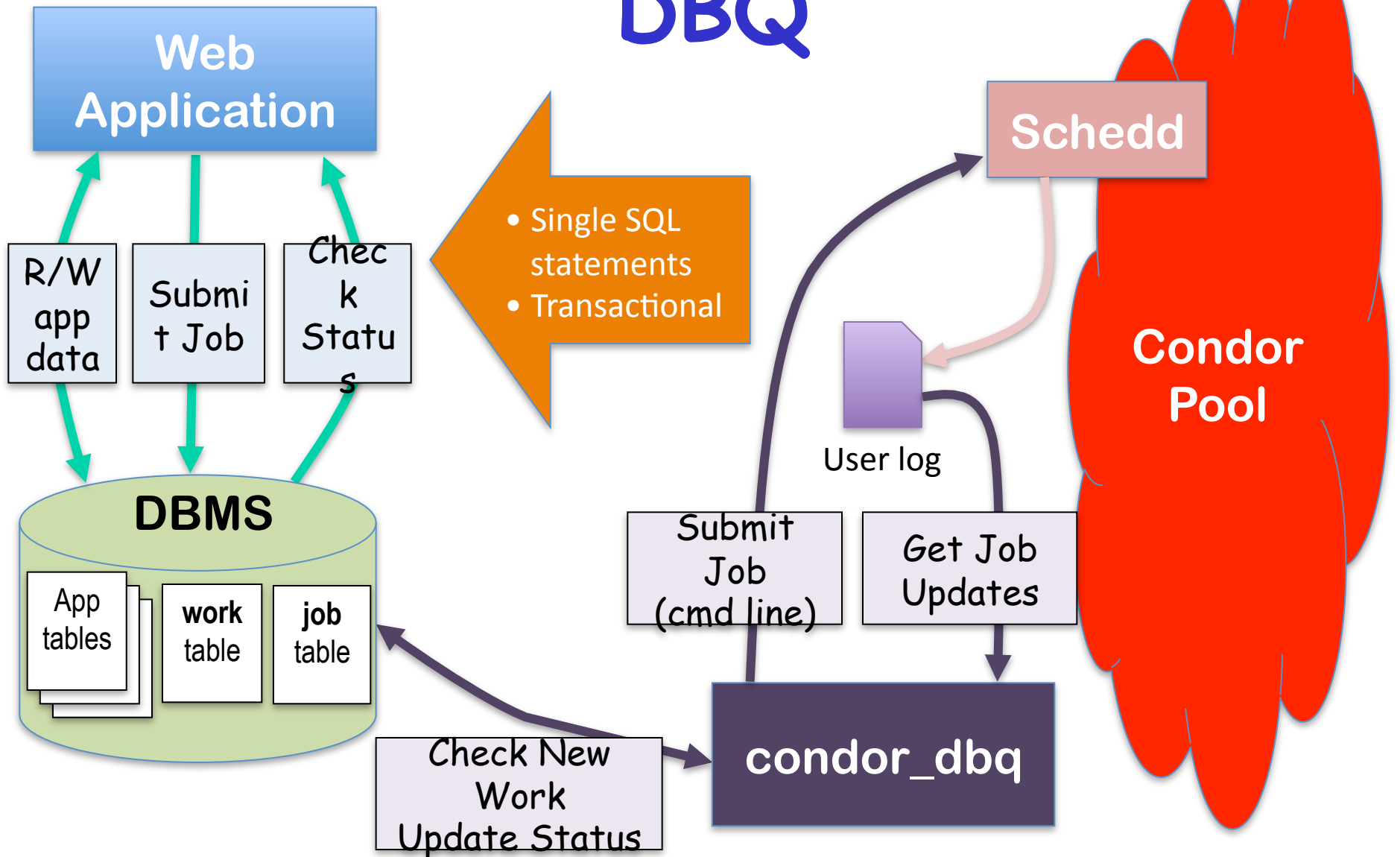
Condor Database Queue

- > Layer on top of Condor
- > Relational database interface to
 - Submit work to Condor
 - Monitor status of submission
 - Monitor status of individual jobs
- > Perfect for applications that
 - Submit jobs to Condor
 - Already use a database

Web App Before Condor



Web App After Condor DBQ



Benefits of Condor DBQ

> Natural simple SQL API

- Submit work

```
insert into work values(condor-submit-  
file)
```

- Check status

```
select * from jobs where work_id = id
```

> Transactions/Consistency comes for free

> DBMS performs crash recovery

Condor DBQ Limitations

- > Overrides log file location
- > All jobs submitted as same user
- > Dagman not supported
- > Only Vanilla and Standard universe jobs supported (others are unknown)
- > Currently only supports PostgreSQL

Condor File Transfer Hooks

- > By default moves files between submit and execute hosts (shadow and starter).
- > New **File Transfer Hooks** - can have URLs grab files from anywhere
 - HTTP (and everything else in curl)
 - HDFS
 - Globus.org
- > Upcoming: How about Condor's SPOOL ?
- > Need to schedule movement? Stork



Virtual Machine Work



- > **Sandboxing**: running vanilla jobs in the VM
 - Isolate the job from execute host.
 - Stage custom execution environments.
 - Sandbox and control the job execution.
 - One way today via Job Router
 - Job router hook picks them up, sets them up inside a VM job, and submits the VM job.

- > **Networking**
 - Particularly of interest for restarts

Fast, quick, light jobs = "tasks"

Copyright 2001 by Randy Glasbergen. www.glasbergen.com

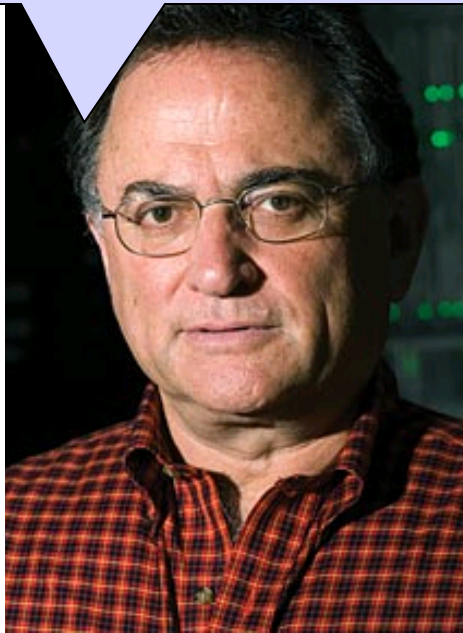


"An aspirin a day will help prevent a heart attack if you have it for lunch instead of a cheeseburger."

- > Options to put a Condor job on a diet
- > Diet ideas:
 - Leave the luggage at home! No job file sandbox, everything in the job ad.
 - Don't pay for strong semantic guarantees if you don't need em. Define expectations on entry, update, completion.
- > Want to honor scheduling policy, however.

High Frequency Computing (HFC)

What?
Meaning?
Lightweight?
½ pound?



Allow condor to handle jobs of short duration that occur frequently.

- > Provides functionality similar to Master/Worker (MW)
- > Still in early development

Condor Wiki Ticket #1095

Some Requirements

- Execute 10 million zero second tasks on 1000 workers in 8 hours
- Each task must contain certain state including GUID and Type
- All interfaces defined using ASCII and sent over raw sockets (Gahp-like)
- Users must be able to query task state

Example Requirements (Cont.)

- Tasks and Workers have attributes to aid in matching
- Workers send heartbeat for hung worker detection by the scheduler
- Workers can be implemented in any language

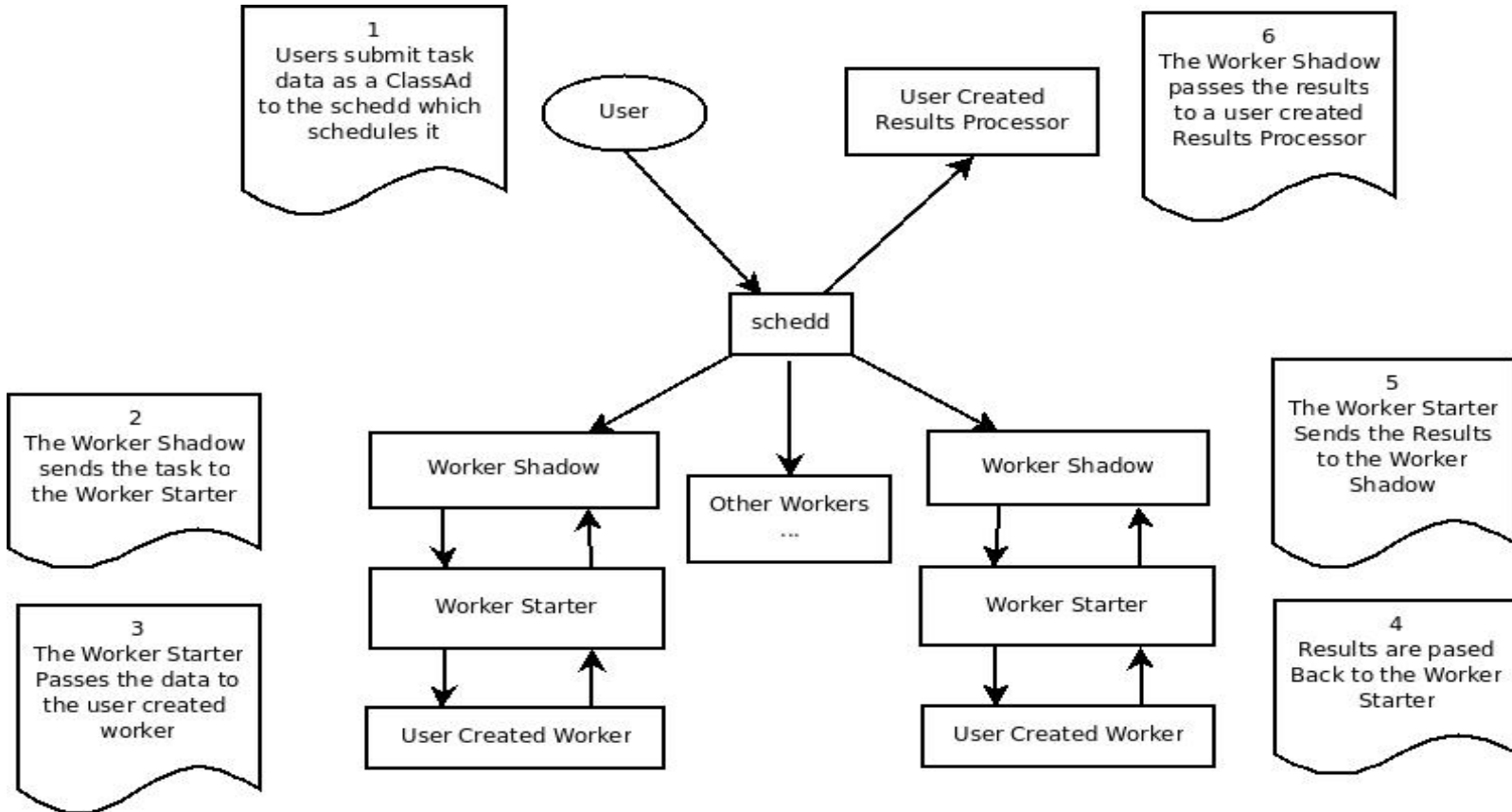
HFC Life of a Task

- Initially, user created workers are scheduled as Vanilla Universe Jobs using Condor
- Users submits tasks to Condor as a ClassAd
- Condor schedules the task and sends it to the appropriate worker

HFC Life of a Task (Cont.)

- Once task processing is complete, the results are sent back to the submit machine, also as a ClassAd.
- The results ad is given to a user created Results Processor.

HFC Architecture



Workflow Help

- > Claim Lifetime
 - Big help for DAGMan
- > Leave behind info to "color" a node
 - Limited # of attributes
 - Lifetime

Looking forward: Ease of Use

- > "There's a knob for that..." (sigh)
- > Pete and Will : a record for every knob
 - Like about:config
 - Allows smaller config file
 - Allows for easier upgrades
- > Quick Start Guides
- > Online Hands-On Tutorials
- > Auto-update



Thank you!

Keep the community chatter
going on condor-users!

