

# 분산 처리 시스템 입문

한국과학기술정보연구원  
대용량데이터허브실

배상욱

2016.12.27.<화>

# Contents



- 분산처리 시스템 소개
- **HTCondor** 소개
- **HTCondor** 설치 및 설정
- **Simple Example**

---

# 분산처리 시스템 소개

---



# 분산처리 시스템 개념

- 개념

- 여러대의 컴퓨터를 통신 회선으로 연결하여 하나의 작업을 처리하는 방식

- 구성 이점

- 자원공유
- 연산속도 향상
- 신뢰성 향상



# 응용 분야

- 기초 연구 실험 분야
- 사회기반시설, 안전 분석 분야
- 고부가가치 데이터 분석 분야
- 의료, 생명 데이터 분석 분야

---

# HTCCondor 소개

---



# HTCondor 개념

- 개념

- 컴퓨팅을 필요로 하는 작업들을 분산시켜 병렬처리 하기 위한 소프트웨어 프레임워크
- Wisconsin-Madison 대학의 HTCondor 팀에 의하여 개발되었고 현재 Apache 라이선스 2.0하에 오픈 소스 형태로 배포
- 1988년에 처음으로 제안 되었으며, 지난 28년 간 지속적으로 기능 추가 및 버그 수정 과정을 수행

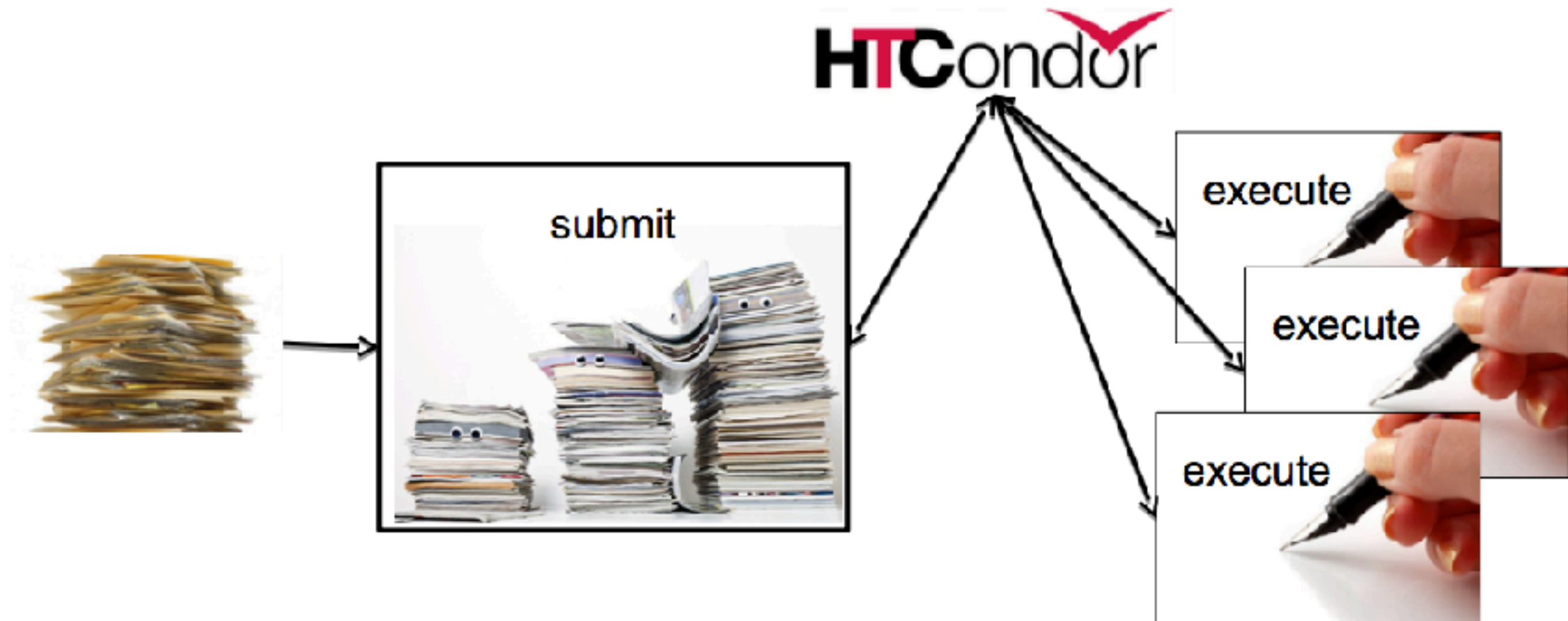
- 특징

- 호환성 : C 프로그래밍 된 오픈소스로, 다른 프로그램에 비하여 높은 호환성
- 성능 : Job ClassAd에 해당 작업 실행에 필요한 자원량을 명시하여 최대한 작업들을 균등 분배
- 다양한 기능 : flocking, checkpoint

# HTCondor 개념

- 동작과정

- 작업(job)을 submit 노드에 제출
- HTCondor 가 모든 컴퓨팅 및 작업(job)을 고려하여 적절하게 스케줄링

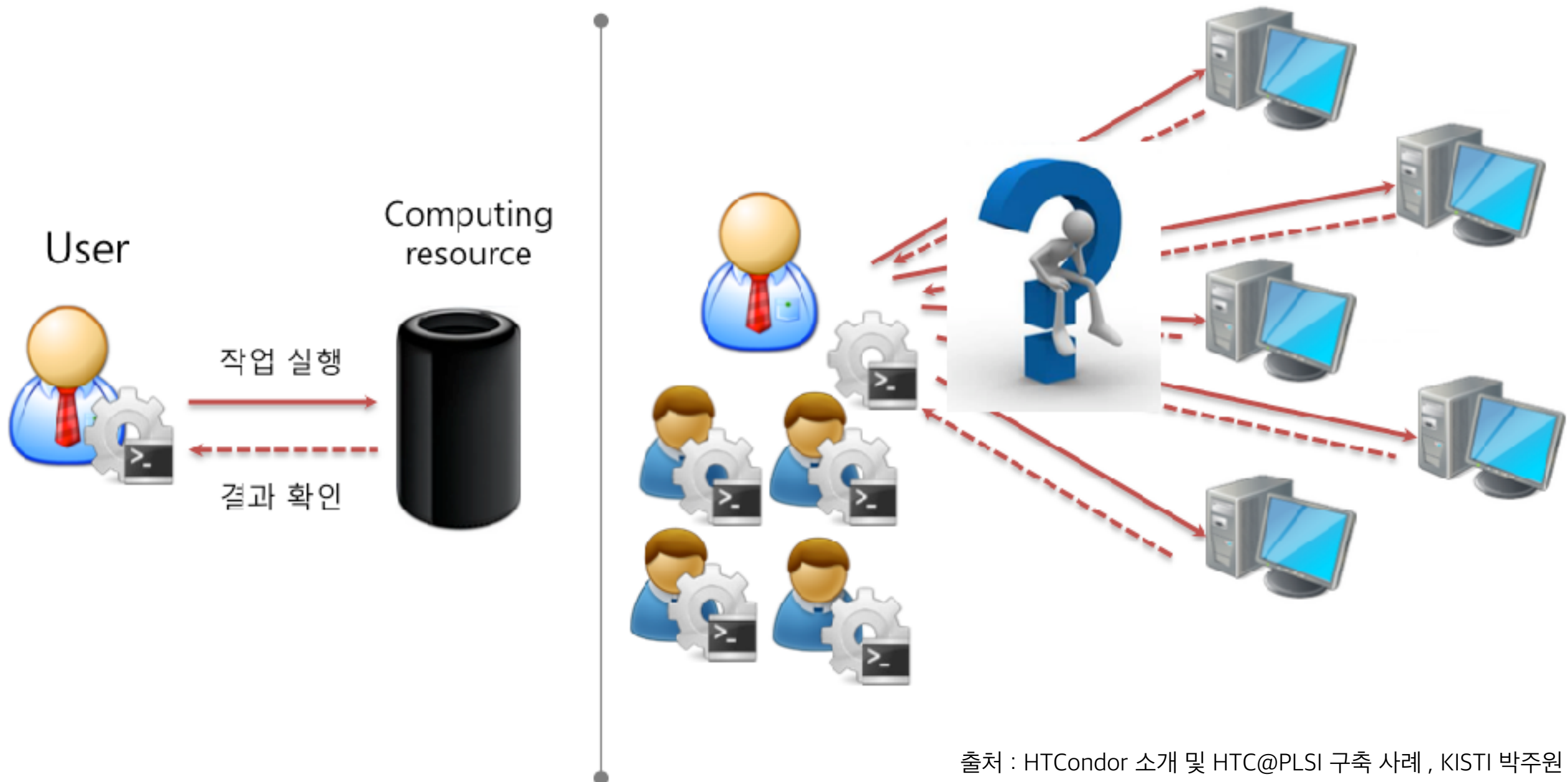


출처 : AN INTRODUCTION TO Using HTCondor -Christina Koch (Condor Week 2016)



# HTCondor 특징

- ClassAD



출처 : HTCondor 소개 및 HTC@PLSI 구축 사례 , KISTI 박주원

# HTCondor 특징

- **ClassAD (Classified Advertisements)**

- Condor에서 사용하는 유연하고 표현성 높은 언어 (자원 요청, 자원제공 등)

- Job ClassAD, Machine ClassAD 정보를 표현

## Job ClassAD

```
✓ ...  
✓ Out = "out/out.2699"  
✓ Cmd = "monte_int"  
✓ TransferInput = "data.2699"  
✓ UserLog = "log.2699"  
✓ Owner = "p377han"  
✓ Requirements = ( TARGET.Arch ==  
  "X86_64" ) && ( TARGET.OpSys ==  
  "LINUX" ) && ( TARGET.Disk >=  
  RequestDisk ) &&  
  ( TARGET.Memory >= RequestMemory )  
  && ( TARGET.HasFileTransfer  
✓ ...
```

## Machine ClassAD

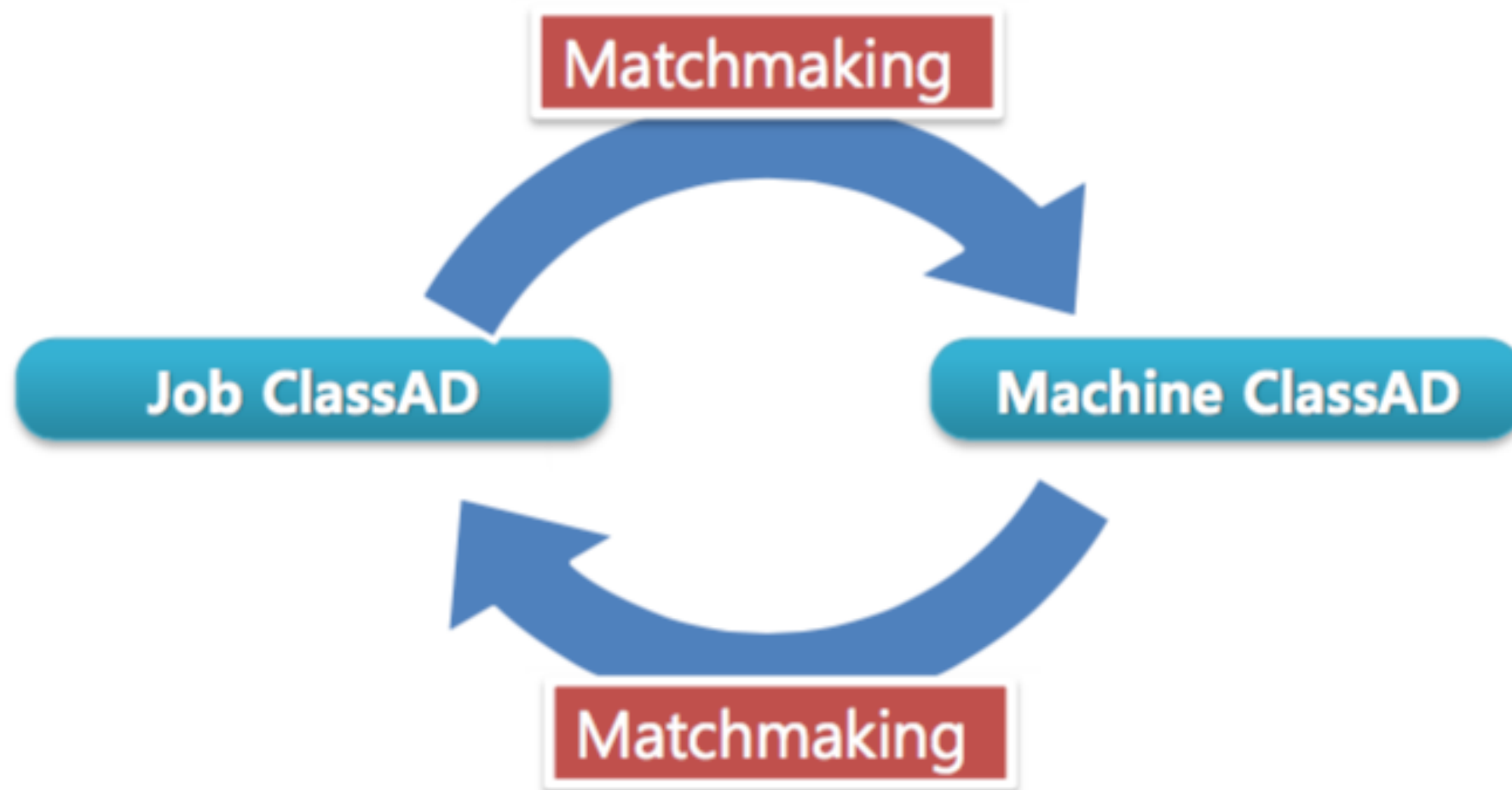
```
✓ ...  
✓ Machine = "glory254.plsi.or.kr"  
✓ OpSysAndVer = "CentOS5"  
✓ JavaVersion = "1.4.2"  
✓ CondorVersion = "$CondorVersion:  
  7.8.8 Mar 20 2013 BuildID: 110288  
  $"  
✓ HardwareAddress =  
  "00:1b:24:78:37:65"  
✓ COLLECTOR_HOST_STRING = "glory-  
  mg01.plsi.or.kr"  
✓ SubnetMask = "255.255.255.0"  
✓ ...
```

출처: HTCondor 소개 및 HTC@PLSI 구축 사례, KISTI 박주원

# HTCondor 특징

- **Matchmaking**

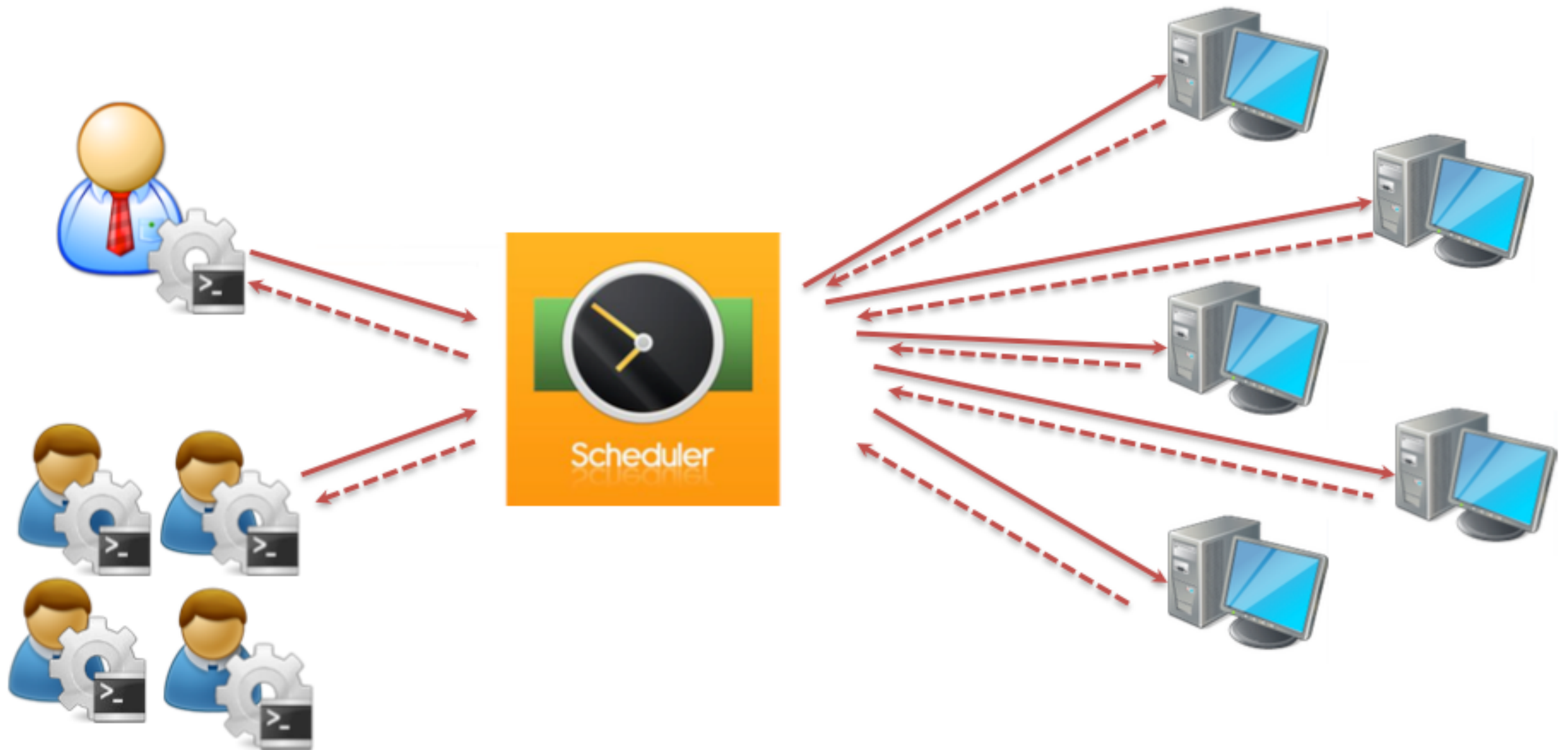
- matchmaker는 job ClassAds와 machine ClassAd를 match 시킴



출처 : HTCondor 소개 및 HTC@PLSI 구축 사례 , KISTI 박주원

# HTCondor 특징

- **Check Point**

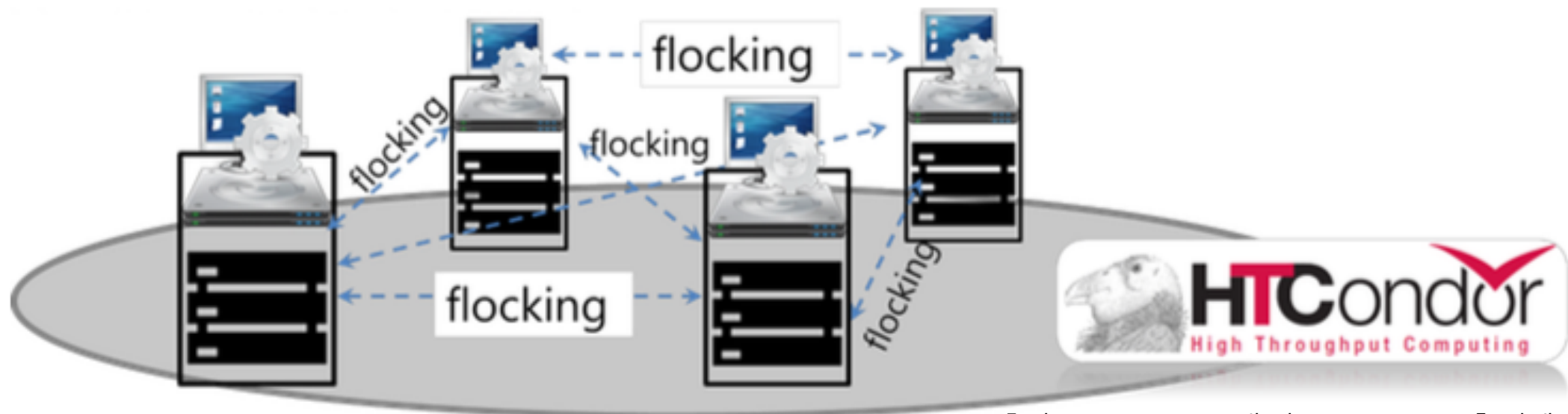


출처 : HTCondor 소개 및 HTC@PLSI 구축 사례 , KISTI 박주원

# HTCondor 특징

- **Flucking**

- HTCondor의 장점 중 하나
- 클러스터 간 작업 연동 기능
- ex> A.condor 라는 HTCondor pool과 B.condor라는 HTCondor pool 이 존재할 경우 A에 작업을 제출했다더라도 B에 자원이 남아 있을 경우 B로 작업이 전달되어 작업을 실행



출처 : HTCondor 소개 및 HTC@PLSI 구축 사례 , KISTI 박주원

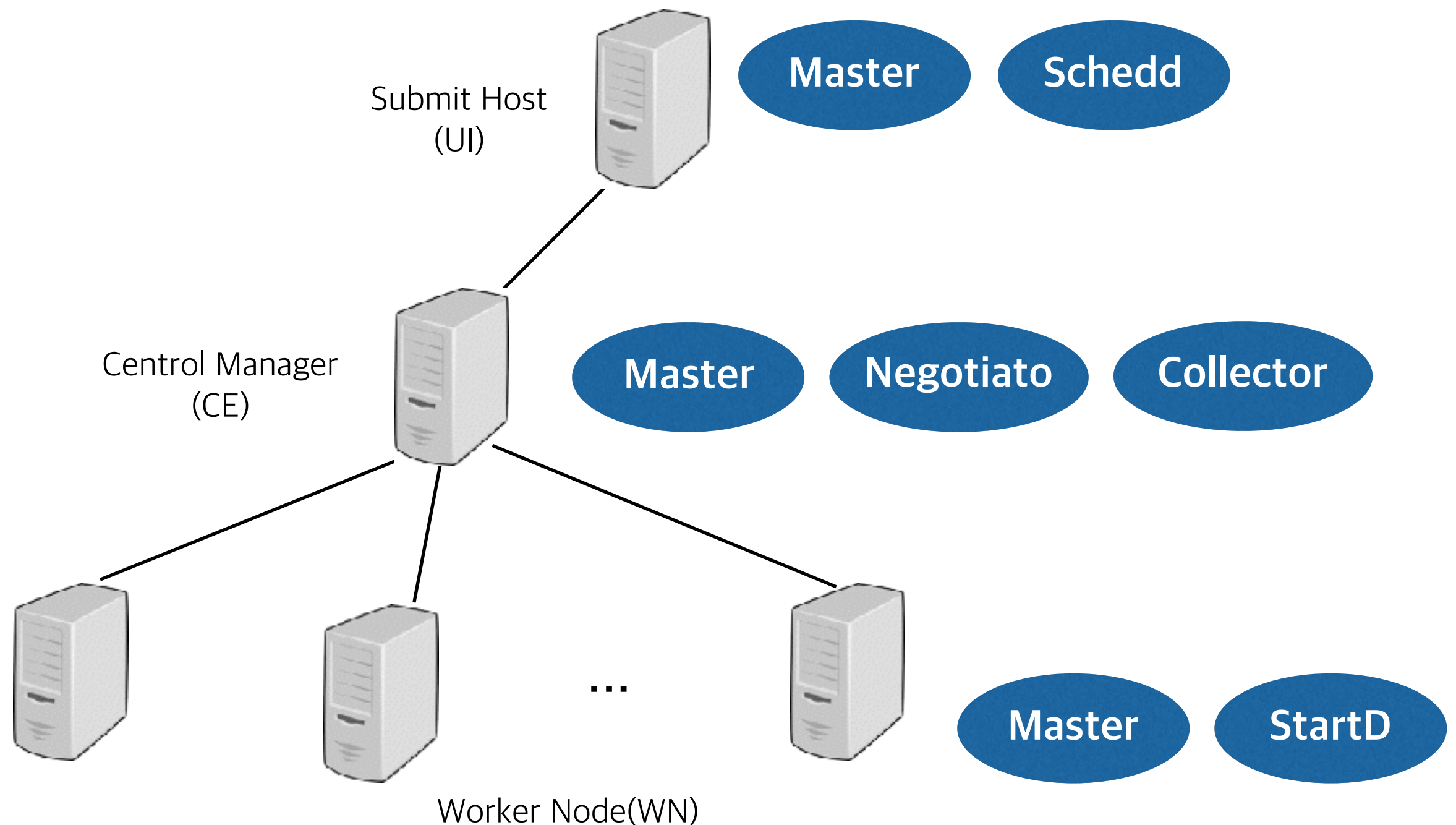


# 용어 설명

용어	설명	비고
Job	어떤 한 시뮬레이션, Work 같은 개념. 유닉스 프로세스랑 비슷	
ClassAds	Condor에서 사용하는 유연하고 표현성 높은 언어 (자원요청, 자원제공 등)	
Machine or Resource	클러스터 같은 실제 Job을 돌리는 노드	
Match Making	Job을 Machine에 할당	
Centrol Manager	일종의 master node, 실제 Job을 중앙에서 Machine에 Matching시킴	Head Node CE
Submit Host	Job 을 submit한 node	Submit Node UI
Execute Host	실제 Job을 담당받아 돌리는 node	Worker Node

# HTCondor 구조

- **Condor Daemon Layout**



# HTCondor 구조

- **condor\_master**



Master

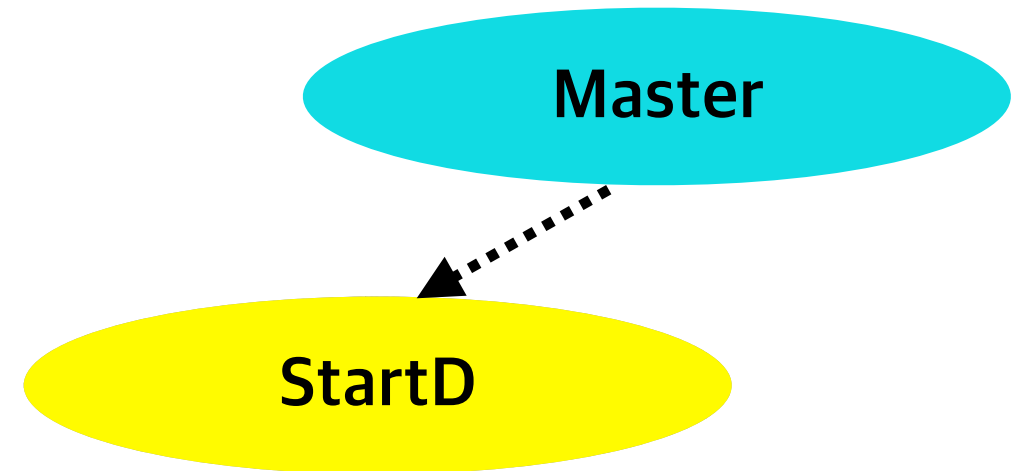
- 기본적인 Daemon
- 모든 Condor Daemon의 시작
- 문제가 있거나 daemon의 종료가 있을 경우, 데몬을 재 시작하고, 관리자에게 email을 전송



# HTCondor 구조

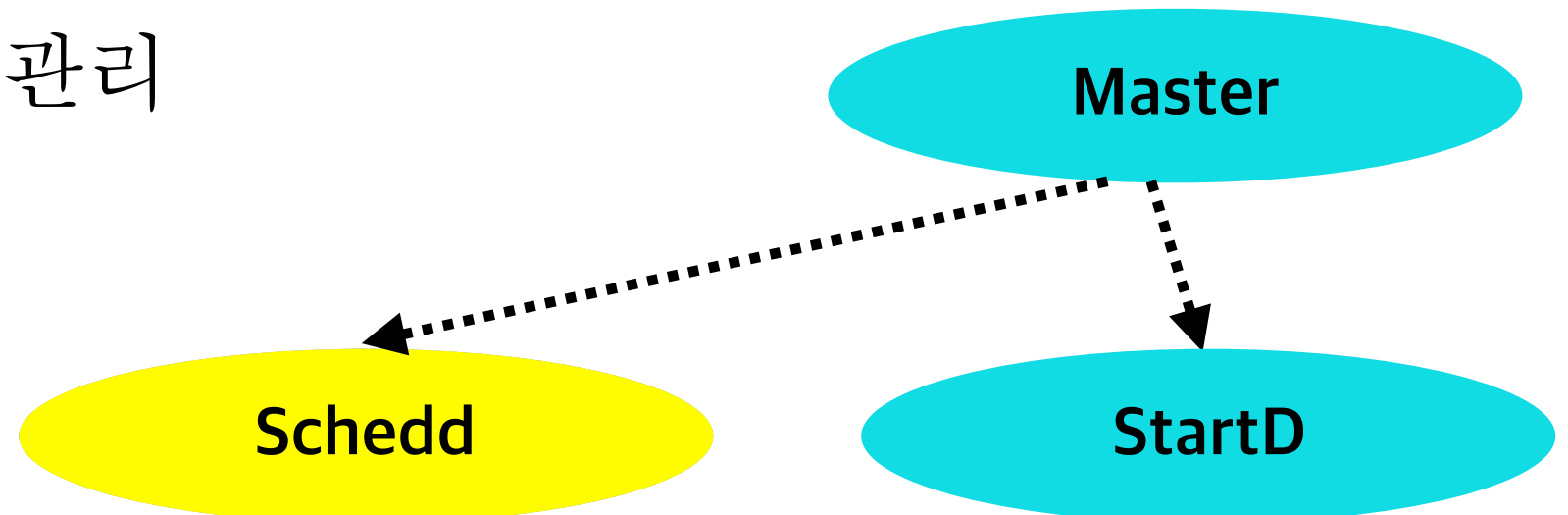
- **condor\_startd**

- Job의 시작, 중지, 재시작
- 실제 작업 수행



- **condor\_schedd**

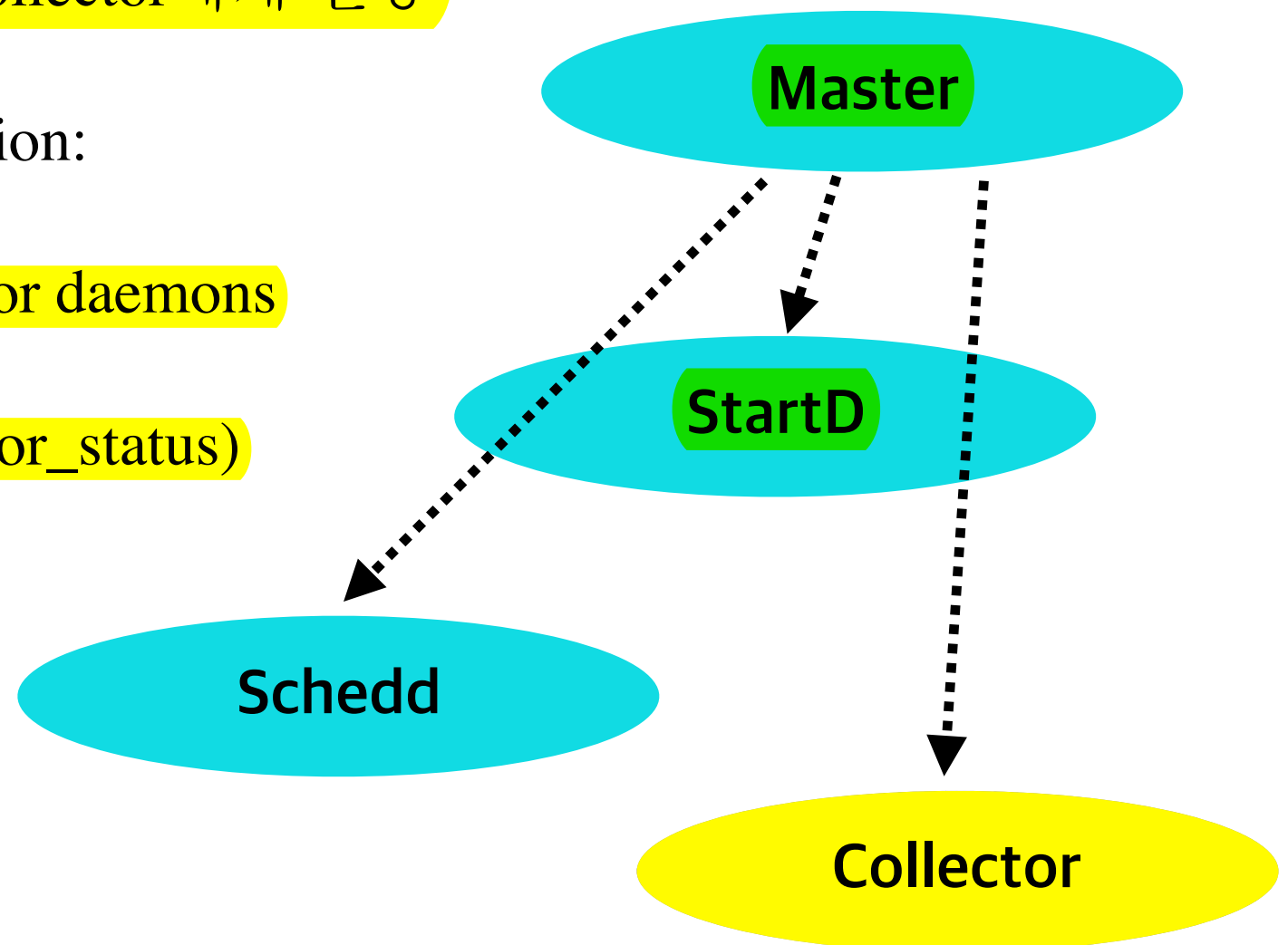
- User command를 수행
  - ✓ ex>condor\_submit, condor\_rm, condor\_q 등
- 큐에 있는 job을 관리



# HTCondor 구조

- **condor\_collector**

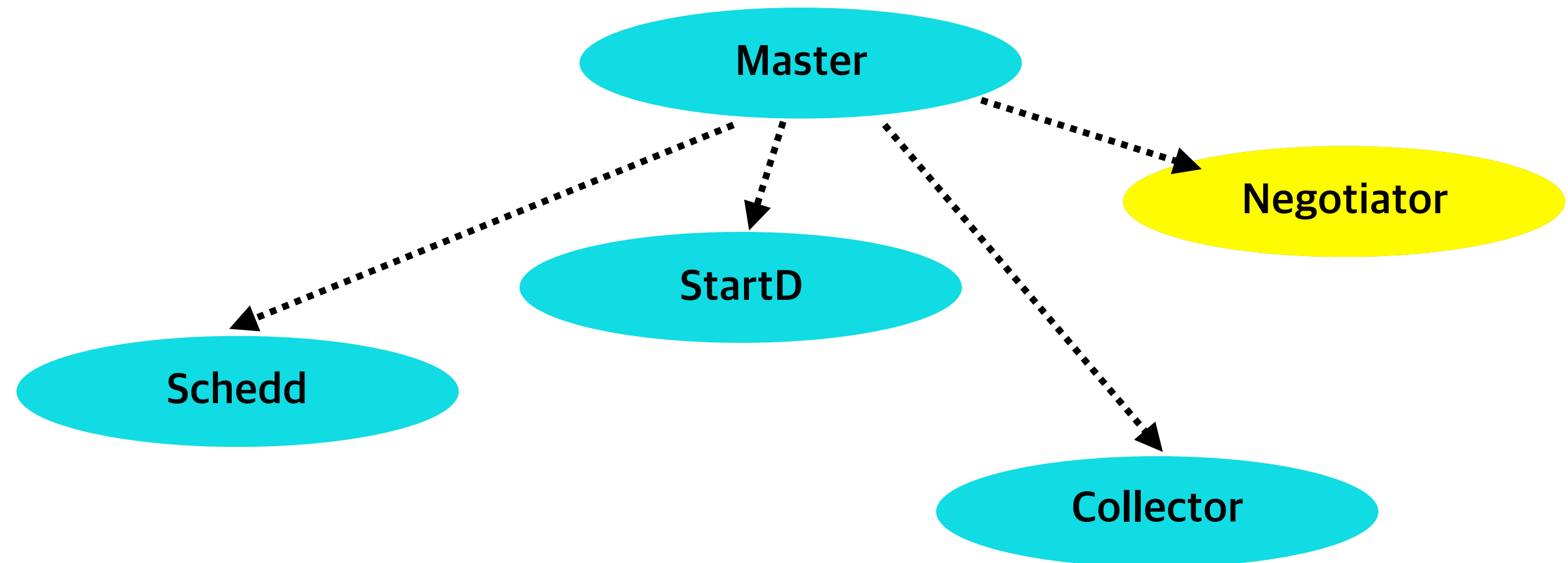
- 모든 다른 condor daemon 으로 부터 정보 수집
- 주기적으로 “ClassAD”를 collector에게 전송
- Services queries for information:
  - ✓ Queries from other Condor daemons
  - ✓ Queries from users (condor\_status)



# HTCondor 구조

- **condor\_negotiator**

- condor에서 “matchmaking”을 수행
- collector로부터 사용가능한 상태의 머신 정보를 수신
- 복수의 job에 대한 적절한 분배



# HTCondor 설정

- **/etc/condor/condor\_config**
- 일반적인 설정
  - /etc/condor/config.d/cluster.conf
    - ✓ HTCondor 환경을 구성하기 위한 기본적인 설정
    - ✓ ex> Master Name, Domain Name 등
- 데몬 설정
  - /etc/condor/config.d/local.conf
    - ✓ 환경 구성에 따른 노드별 데몬 설정
    - ✓ ex> MASTER, COLLECTOR, NEGOTIATOR, SCHEDD

# HTCondor 설정

- 일반적인 설정 (cluster.conf)

```
1 # The following should be your cluster domain. This is an arbitrary string used by Condor, not necessarily matching your IP domain
2 UID_DOMAIN = yourdomain.org
3 # Human readable name for your Condor pool
4 COLLECTOR_NAME = "OSG Cluster Condor at $(UID_DOMAIN)"
5 # A shared file system (NFS), e.g. job dir, is assumed if the name is the same
6 FILESYSTEM_DOMAIN = $(UID_DOMAIN)
7 # Here you have to use your network domain, or any comma separated list of hostnames and IP addresses including all your
8 # condor hosts. * can be used as wildcard
9 ALLOW_WRITE = *.yourdomain.org
10 ALLOW_READ = *.yourdomain.org
11 CONDOR_ADMIN = root@$(FULL_HOSTNAME)
12 # The following should be the full name of the head node (Condor central manager)
13 CONDOR_HOST = gc-hn.yourdomain.org
14 # Port range should be opened in the firewall (can be different on different machines)
15 # This 9000-9999 is coherent with the iptables configuration in the Firewall documentation
16 IN_HIGHPORT = 9999
17 IN_LOWPORT = 9000
18 # This is to enforce password authentication
19 SEC_DAEMON_AUTHENTICATION = required
20 SEC_DAEMON_AUTHENTICATION_METHODS = password
21 SEC_CLIENT_AUTHENTICATION_METHODS = password,fs,gsi
22 SEC_PASSWORD_FILE = /var/lib/condor/condor_credential
23 ALLOW_DAEMON = condor_pool@*
24 ## Scheduling parameters for the startd
25 TRUST_UID_DOMAIN = TRUE
26 # start as available and do not suspend, preempt or kill
27 START = TRUE
28 SUSPEND = FALSE
29 PREEMPT = FALSE
30 KILL = FALSE
31 # In this setup we use the config directory instead of the local config
32 REQUIRE_LOCAL_CONFIG_FILE = False
```

# HTCondor 설정

- 데몬 설정 (**local.conf**)
  - ex1> DAEMON\_LIST = MASTER, COLLECTOR, NEGOTIATOR
  - ex2> DAEMON\_LIST = MASTER, SCHEDD
  - ex3> DAEMON\_LIST = MASTER, STARTD

---

# HTCCondor 설치 및 설정

---



# 구성 환경

- 노드 정보

	Host Name	IP
Central Manager (CE)	<u>group01-mn.g1.gsdc.org</u>	10.0.12.10
Submit Host (UI)	<u>group01-wn04.g1.gsdc.org</u>	10.0.12.14
Execute Host (WN)	<u>group01-wn01.g1.gsdc.org</u>	10.0.12.11



Submit Node



Head Node



Work Node



# HTCondor 설치 및 설정 과정

- 설치 환경 확인
- **HTCondor** 설치
- **HTCondor** 설정
- 기타 설정
  - 방화벽, SELINUX
- 서비스 시작
- 계정 생성
- 테스트

# 설치 #1

- 버전 확인

- #cat /etc/redhat-release

```
[wookie@group01-mn simple_job]$ cat /etc/redhat-release
Scientific Linux release 7.2 (Nitrogen)
[wookie@group01-mn simple_job]$
```

- 설치 준비

- wget <http://research.cs.wisc.edu/htcondor/yum/repo.d/htcondor-stable-rhel7.repo>

```
[root@group01-mn yum.repos.d]# wget http://research.cs.wisc.edu/htcondor/yum/repo.d/htcondor-stable-rhel7.repo
--2016-12-20 01:17:46-- http://research.cs.wisc.edu/htcondor/yum/repo.d/htcondor-stable-rhel7.repo
Resolving research.cs.wisc.edu (research.cs.wisc.edu)... 128.105.7.58
Connecting to research.cs.wisc.edu (research.cs.wisc.edu)|128.105.7.58|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 167 [text/plain]
Saving to: 'htcondor-stable-rhel7.repo'

100%[----->] 167      --.-K/s   in 0s

2016-12-20 01:17:46 (8.83 MB/s) - 'htcondor-stable-rhel7.repo' saved [167/167]

[root@group01-mn yum.repos.d]# wget http://research.cs.wisc.edu/htcondor/yum/RPM-GPG-KEY-HTCondor
--2016-12-20 01:18:03-- http://research.cs.wisc.edu/htcondor/yum/RPM-GPG-KEY-HTCondor
Resolving research.cs.wisc.edu (research.cs.wisc.edu)... 128.105.7.58
Connecting to research.cs.wisc.edu (research.cs.wisc.edu)|128.105.7.58|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1752 (1.7K) [text/plain]
Saving to: 'RPM-GPG-KEY-HTCondor'

100%[----->] 1,752      --.-K/s   in 0s

2016-12-20 01:18:03 (87.7 MB/s) - 'RPM-GPG-KEY-HTCondor' saved [1752/1752]
```

# 설치 #2

- 설치

- #sudo yum install condor.x86\_64

```
[root@group01-mn yum.repos.d]# yum install condor.x86_64
Loaded plugins: langpacks
epel/x86_64/metalink | 4.6 kB 00:00:00
epel | 4.3 kB 00:00:00
htcondor-stable | 2.9 kB 00:00:00
sl | 3.7 kB 00:00:00
sl-fastbugs | 2.9 kB 00:00:00
sl-security | 2.9 kB 00:00:00
(1/5): epel/x86_64/primary_db | 4.4 MB 00:00:00
(2/5): epel/x86_64/updateinfo | 692 kB 00:00:00
(3/5): sl-security/x86_64/updateinfo | 173 kB 00:00:01
(4/5): htcondor-stable/primary_db | 270 kB 00:00:02
(5/5): sl-security/x86_64/primary_db | 4.0 MB 00:00:04
Resolving Dependencies
--> Running transaction check
```

...

```
Installed:
condor.x86_64 0:8.4.10-1.el7

Dependency Installed:
condor-classads.x86_64 0:8.4.10-1.el7      condor-external-libs.x86_64 0:8.4.10-1.el7      condor-procd.x86_64 0:8.4.10-1.el7
perl-Archive-Tar.noarch 0:1.92-2.el7      perl-Compress-Raw-Bzip2.x86_64 0:2.061-3.el7      perl-Compress-Raw-Zlib.x86_64 1:2.061-4.el7
perl-Data-Dumper.x86_64 0:2.145-3.el7      perl-IO-Compress.noarch 0:2.061-2.el7      perl-IO-Zlib.noarch 1:1.10-286.el7
perl-Package-Constants.noarch 1:0.02-286.el7      ssmtp.x86_64 0:2.64-14.el7

Complete!
[root@group01-mn yum.repos.d]#
```

# 기본적인 설정 내용

- 일반적인 설정
  - `/etc/condor/config.d/cluster.conf`
    - ✓ HTCondor 환경을 구성하기 위한 기본적인 설정
    - ✓ ex> Master Name, Domain Name 등
- 데몬 설정
  - `/etc/condor/config.d/local.conf`
    - ✓ 환경 구성에 따른 노드별 데몬 설정
    - ✓ ex> MASTER, COLLECTOR, NEGOTIATOR, SCHEDD

# 일반적인 설정

- `#sudo vim /etc/condor/config.d/cluster.conf`

```
1  UID_DOMAIN = g1.gsdc.org
2  COLLECTOR_NAME = "GSDC School"
3  FILESYSTEM_DOMAIN = g1.gsdc.org
4  ALLOW_WRITE = *.g1.gsdc.org
5  ALLOW_READ = *.g1.gsdc.org
6  CONDOR_ADMIN = wookiee@kisti.re.kr
7  CONDOR_HOST = group01-mn
8  IN_HIGHPORT = 9999
9  IN_LOWPORT = 9000
10 SEC_DAEMON_AUTHENTICATION = required
11 SEC_DAEMON_AUTHENTICATION_METHODS = password
12 SEC_CLIENT_AUTHENTICATION_METHODS = password,fs,gsi
13 SEC_PASSWORD_FILE = /var/lib/condor/condor_credential
14 ALLOW_DAEMON = condor_pool@*
15 NEGOTIATOR_INTERVAL = 20
16 TRUST_UID_DOMAIN = TRUE
17 START = TRUE
18 SUSPEND = FALSE
19 PREEMPT = FALSE
20 KILL = FALSE
```



# 데몬 설정

- **#sudo vim /etc/condor/config.d/local.conf**

- CE

```
1 DAEMON_LIST = MASTER, COLLECTOR, NEGOTIATOR
```

- WN

```
1 DAEMON_LIST = MASTER, STARTD
```

- UI

```
1 DAEMON_LIST = MASTER, |SCHEDD
```

# 서비스 시작

- 방화벽 설정

- `#sudo firewall-cmd --permanent --add-port=9000-9999/tcp`
- `#sudo firewall-cmd --permanent --add-port=9000-9999/udp`

- 서비스 시작

- Head Node & Submit node & Worker node
  - ✓ `#sudo systemctl start condor.service`

- 계정 생성

- Head Node & Submit node & Worker node
  - ✓ `adduser #####`
  - ✓ `passwd #####`

# 테스트

- **#condor\_status**

```
[wookie@school-13-ui ~]$ condor_status
Name                OpSys      Arch    State      Activity LoadAv Mem    ActvtyTime
slot1@school-13-wn  LINUX      X86_64  Unclaimed  Idle      0.000  938    0+00:04:41
slot2@school-13-wn  LINUX      X86_64  Unclaimed  Idle      0.000  938    0+00:05:05
Total Owner Claimed Unclaimed Matched Preempting Backfill
X86_64/LINUX      2         0         0         2         0         0         0
Total             2         0         0         2         0         0         0
[wookie@school-13-ui ~]$
```

- **#condor\_q**

```
[wookie@school-13-ui ~]$ condor_q

-- Submitter: school-13-ui.experiment.com : <172.16.13.1:9486> : school-13-ui.experiment.com
ID      OWNER      SUBMITTED  RUN_TIME ST PRI SIZE CMD
0 jobs; 0 completed, 0 removed, 0 idle, 0 running, 0 held, 0 suspended
[wookie@school-13-ui ~]$
```



[http://www.evernote.com/l/  
AuHI6VnGu9BK0ZVfm\\_6a8z2jC  
KfZwrjIGmA/](http://www.evernote.com/l/AuHI6VnGu9BK0ZVfm_6a8z2jCKfZwrjIGmA/)

---

# Simple Example

---



# Job submission #1

- **Execution File**

```
#!/bin/bash  
  
echo `date`  
sleep 5  
echo `date`  
~
```

- **Descriptions File**

```
#####  
#Submit description file for hello program#  
#####  
Executable      = hello  
Universe        = standard  
Output          = hello.out  
Log             = hello.log  
Queue  
~
```

# Job submission #2

- Descriptions File

- Universe

- ✓ 실행 파일을 수행할 환경 지정

```
#####  
#Submit description file for hello program#  
#####  
Executable      = hello  
Universe         = standard  
Output           = hello.out  
Log              = hello.log  
Queue  
~
```

- ✓ standard : 리모트에서 수행 중인 job의 상태를 호스트에 저장할 수 있는 환경으로 **checking-point**와 **remote system call** 가능. standard 환경에서 실행되는 파일은 반드시 condor\_compile과 링크되어야 함
    - ✓ vanilla: condor\_compile로 링크될 수 없는 경우로 셸스크립트가 대표적인 예 (일반적인 경우 사용)
    - ✓ grid: 그리드 Job을 실행할 때 사용
    - ✓ java: Java 프로그램을 실행하는 환경

# Job submission #3

- Descriptions File

- Universe (con't)

- ✓ local: 경량의 Job을 바로 수행해보기 위한 사용환경으로 리모트 시스템과 매치메이킹(Match Making)을 하지않고 호스트에서 바로 수행

- ✓ scheduler: local과 유사하나 condor\_stardd 데몬이 Job 을 핸들링하지 않는 차이점이 존재

- ✓ parallel: MPI Job을 수행하기 위한 환경

- ✓ vm: VMWare나 Xen과 같은 가상머신을 수행하기 위한 환경

- Output

- ✓ 실행결과 저장할 곳 지정

```
#####  
#Submit description file for hello program#  
#####  
Executable      = hello  
Universe         = standard  
Output           = hello.out  
Log              = hello.log  
Queue  
~
```

# Job submission #4

- Descriptions File

- Log

- ✓ 로그 저장할 곳을 지정

- Queues

- ✓ Job을 Queue에 넣음. 몇 번 수행할지 지정. 지정하지 않으면 한개의 Job 수행

```
#####  
#Submit description file for hello program#  
#####  
Executable      = hello  
Universe        = standard  
Output          = hello.out  
Log             = hello.log  
Queue  
~
```

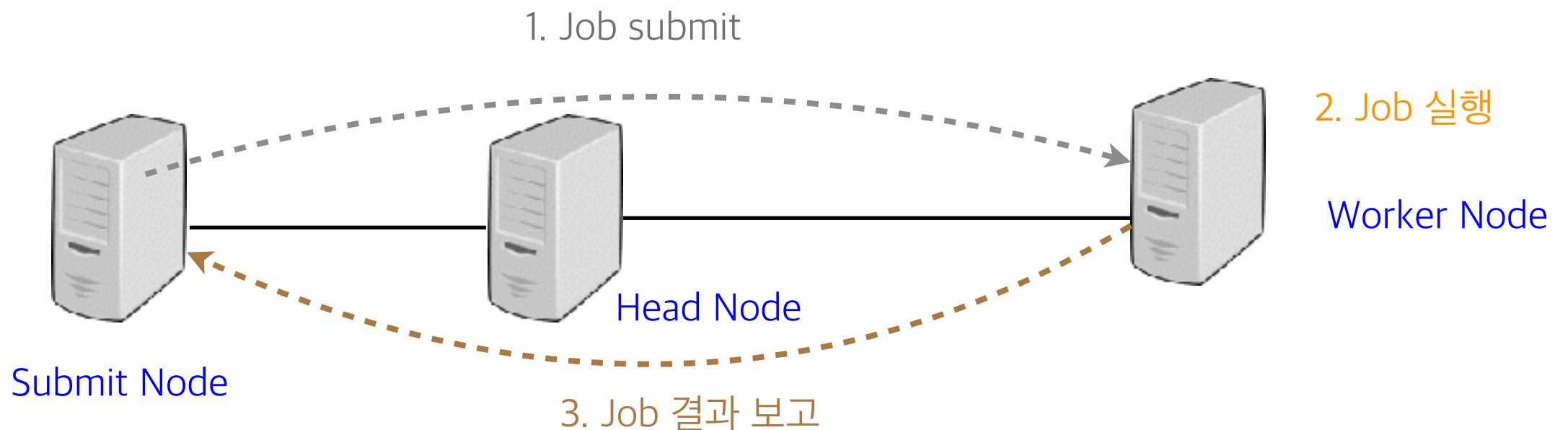
# 예제 #1

현재 시간을 출력하고, 5초간 슬립 상태였다가 다시 현재의 시간 출력

- **#cat out.txt**

```
[wookie@school-13-ui ~]$ cat out.txt  
Tue Feb 3 02:59:48 UTC 2015  
Tue Feb 3 02:59:53 UTC 2015  
[wookie@school-13-ui ~]$
```

- 환경구성



## 예제 #2

현재 시간을 출력하고, 5초간 슬립 상태였다가 다시 현재의 시간 출력

- `#mkdir /home/XXX/test1`
- `#vim /home/XXX/test1/date.sh`
- `#vim /home/XXX/test1/date.jds`

```
#!/bin/bash
```

```
echo `date`
```

```
sleep 5
```

```
echo `date`
```

```
~
```

```
executable = date.sh
```

```
universe = vanilla
```

```
output = out.txt
```

```
error = error.txt
```

```
log = log.txt
```

```
should_transfer_files = YES
```

```
when_to_transfer_output = ON_EXIT
```

```
transfer_input_files = date.sh
```

```
queue
```



# 예제 #3

현재 시간을 출력하고, 5초간 슬립 상태였다가 다시 현재의 시간 출력

- `#cd /home/XXX/test1`
- `#condor_submit date.jds`

```
[wookie@school-13-ui ~]$ condor_submit date.jds
Submitting job(s).
1 job(s) submitted to cluster 3.
```

- `#condor_q`

```
[wookie@school-13-ui ~]$ condor_q

-- Submitter: school-13-ui.experiment.com : <172.16.13.1:9046> : school-13-ui.experiment.com
ID      OWNER      SUBMITTED  RUN_TIME ST PRI SIZE CMD
  3.0    wookie     2/3 02:59   0+00:00:04 R  0   0.0  date.sh

1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended
[wookie@school-13-ui ~]$ condor_q
```



# 예제 #4

현재 시간을 출력하고, 5초간 슬립 상태였다가 다시 현재의 시간 출력

- ls

```
[wookie@fermicloud054 test5]$ ls  
date.jds  date.sh  error.txt  log.txt  out.txt
```

- #cat out.txt

```
[wookie@fermicloud054 test5]$ cat out.txt  
Fri Jan 30 01:06:01 CST 2015  
Fri Jan 30 01:06:06 CST 2015  
[wookie@fermicloud054 test5]$ █
```

# Tip #1

- **Help! I'm buried in Email!**
  - jds 파일

```
executable = count.sh  
  
universe = vanilla  
  
output = out/out.txt  
error = out/err.txt  
log = out/log.txt  
  
notification = complete  
notify_user=wookie@kisti.re.kr  
  
should_transfer_files = YES  
when_to_transfer_output = ON_EXIT  
transfer_input_files = count.sh  
queue
```

출처 : [http://htcondorkorea.blogspot.kr/2014/07/blog-post\\_14.html](http://htcondorkorea.blogspot.kr/2014/07/blog-post_14.html)

# Tip #2

- **Removing Processes**

- 하나의 프로세스 삭제

- ✓ `condor_rm 9.47`

- 하나의 클러스터 삭제

- ✓ `condor_rm 9`

- 모든 job 삭제

- ✓ `condor_rm -a`

# Tip #3

- who used condor?

- condor\_q -global

```
[root@ui20 ~]# condor_q -global

-- Schedd: ui10.sdform.kr : <134.75.124.121:14353>
ID          OWNER          SUBMITTED    RUN_TIME ST PRI SIZE CMD
265049.0    goni                    12/14 11:56   0+00:00:01 H  0   97.7 run_test.sh 0 5 0
272342.0    hyunyong                12/18 01:35   4+08:11:05 R  0  976.6 run000095_001.sh
272343.0    hyunyong                12/18 01:36   4+08:11:04 R  0  976.6 run000097_001.sh
[root@ui20 ~]#
```

- condor\_status -submitters

```
[root@ldg-ui home]# condor_status -submitters
Name                               Machine                               RunningJobs IdleJobs HeldJobs
eddy@sdform.kr                     ldg-ui.sdform.kr                     3           0           0
jeongcho.kim@sdform.kr             ldg-ui.sdform.kr                     41          0           0
wskim@sdform.kr                    ldg-ui.sdform.kr                     40          0           0
                                     RunningJobs IdleJobs HeldJobs
                                     eddy@sdform.kr 3 0 0
                                     jeongcho.kim@sdform. 41 0 0
                                     wskim@sdform.kr 40 0 0
                                     Total 84 0 0
[root@ldg-ui home]#
```

# Tip #4

- **Are the Requirements Reasonable?**
  - `condor_q -analyze`

```
---  
24133.000: Request is held.  
  
Hold reason: Error from slot9@cms-t3-wn1004.sdfarm.kr: STARTER at 134.75.124.54 failed to send file(s) to <134.75.124.127:13474>: error reading from /cms_cond  
or/lib/condor/execute/dir_48812/lep2_pt_tight.png: (errno 2) No such file or directory; SHADOW failed to receive file(s) from <134.75.124.54:9709>  
  
[wookie@ui20 ~]$
```



# Tip #5

- 현재 러닝 중인 **machine** 또는 **slot** 확인

- condor\_status -run

```
[root@ui120 condor]# condor_status -run
```

Name	OpSys	Arch	LoadAv	RemoteUser	ClientMachine
slot24@wn4031.s	LINUX	X86_64	0.000	youn@*.sdfarm.kr	ui10.sdfarm.kr
slot10@wn4032.s	LINUX	X86_64	0.000	youn@*.sdfarm.kr	ui10.sdfarm.kr
slot1@wn4034.sd	LINUX	X86_64	0.000	youn@*.sdfarm.kr	ui10.sdfarm.kr
slot3@wn4034.sd	LINUX	X86_64	0.000	youn@*.sdfarm.kr	ui10.sdfarm.kr
slot16@wn4035.s	LINUX	X86_64	0.000	youn@*.sdfarm.kr	ui10.sdfarm.kr
slot20@wn4035.s	LINUX	X86_64	0.000	youn@*.sdfarm.kr	ui10.sdfarm.kr
slot21@wn4035.s	LINUX	X86_64	0.000	youn@*.sdfarm.kr	ui10.sdfarm.kr
slot11@wn4045.s	LINUX	X86_64	0.000	youn@*.sdfarm.kr	ui10.sdfarm.kr

- 사용 가능한 **machine** 또는 **slot** 확인

- condor\_status -available

```
[root@ui120 condor]# condor_status -available
```

Name	OpSys	Arch	State	Activity	LoadAv	Mem	ActivityTime
slot10@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.000	3628	0+12:04:27
slot11@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.000	3628	0+12:04:28
slot12@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.000	3628	0+12:04:29
slot13@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.000	3628	0+12:04:30
slot14@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.000	3628	0+12:04:31
slot15@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.070	3628	0+16:04:40
slot16@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.000	3628	0+12:04:25
slot17@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.000	3628	0+12:04:26
slot18@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.000	3628	0+12:04:27
slot19@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.000	3628	0+12:04:28
slot1@wn4031.sdfo	LINUX	X86_64	Unclaimed	Idle	0.000	3628	0+00:04:00

...

```
slot5@wn4045.sdfo LINUX | X86_64 | Unclaimed | Idle | 0.000 | 3628 | 0+13:18:46 || slot6@wn4045.sdfo | LINUX | X86_64 | Unclaimed | Idle | 0.000 | 3628 | 0+13:18:47 |
| slot7@wn4045.sdfo | LINUX | X86_64 | Unclaimed | Idle | 0.000 | 3628 | 0+00:18:17 |
| slot8@wn4045.sdfo | LINUX | X86_64 | Unclaimed | Idle | 0.000 | 3628 | 0+13:18:41 |
| slot9@wn4045.sdfo | LINUX | X86_64 | Unclaimed | Idle | 0.000 | 3628 | 0+13:18:42 |

```

Machines	Owner	Claimed	Unclaimed	Matched	Preempting
X86_64/LINUX	280	0	0	280	0
Total	280	0	0	280	0



# Tip #6

- 상태 리스트업
  - condor\_status -master

```
[wookie@ui10 condor]$ condor_status -master
Name

cms-sejong.sdfarm.kr
ui10.sdfarm.kr
ui20.sdfarm.kr
wn4031.sdfarm.kr
wn4032.sdfarm.kr
wn4033.sdfarm.kr
wn4034.sdfarm.kr
wn4035.sdfarm.kr
wn4036.sdfarm.kr
wn4037.sdfarm.kr
wn4038.sdfarm.kr
wn4045.sdfarm.kr
[wookie@ui10 condor]$
```

- condor\_status -schedd

```
[wookie@ui10 condor]$ condor_status -schedd
Name                Machine    TotalRunningJobs TotalIdleJobs TotalHeldJobs
cms-sejong.sdfarm.kr cms-sejong      0              0             1
ui10.sdfarm.kr       ui10.sdfar     7              0             0
ui20.sdfarm.kr       ui20.sdfar     0              0             0
                    TotalRunningJobs TotalIdleJobs TotalHeldJobs
Total                7              0             1
[wookie@ui10 condor]$
[wookie@ui10 condor]$
```

# Tip #7

- **submit 정보 확인**
  - `condor_q -submitter XXX`

```
[root@ui20 condor]# condor_q -submitter youn -format "%s" Owner -format " %d\n" ClusterID -format "%d\n" ProcID
youn      23451.6
youn      23451.81
youn      23451.95
youn      23451.130
youn      23451.143
youn      23451.155
youn      23451.203
[root@ui20 condor]#
```

- `condor_q -analyze`

```
[root@ui10 /]# condor_q -ana 23451.6

-- Submitter: ui10.sdfarm.kr : <134.75.124.121:13134> : ui10.sdfarm.kr
---
23451.006: Request is running.

[root@ui10 /]#
```

# Reference

**[1] HTCondor 소개 및 HTC@PLSI 구축 사례 , KISTI 박주원**

**[2] <https://twiki.grid.iu.edu/bin/view/Documentation/Release3/InstallCondor>**

**[3] Condor by Example, Douglas Than**

**[4] Installation Guide for CMS Tier 3 Farm, KISTI**



감사합니다.