

## **Dealing with special users**

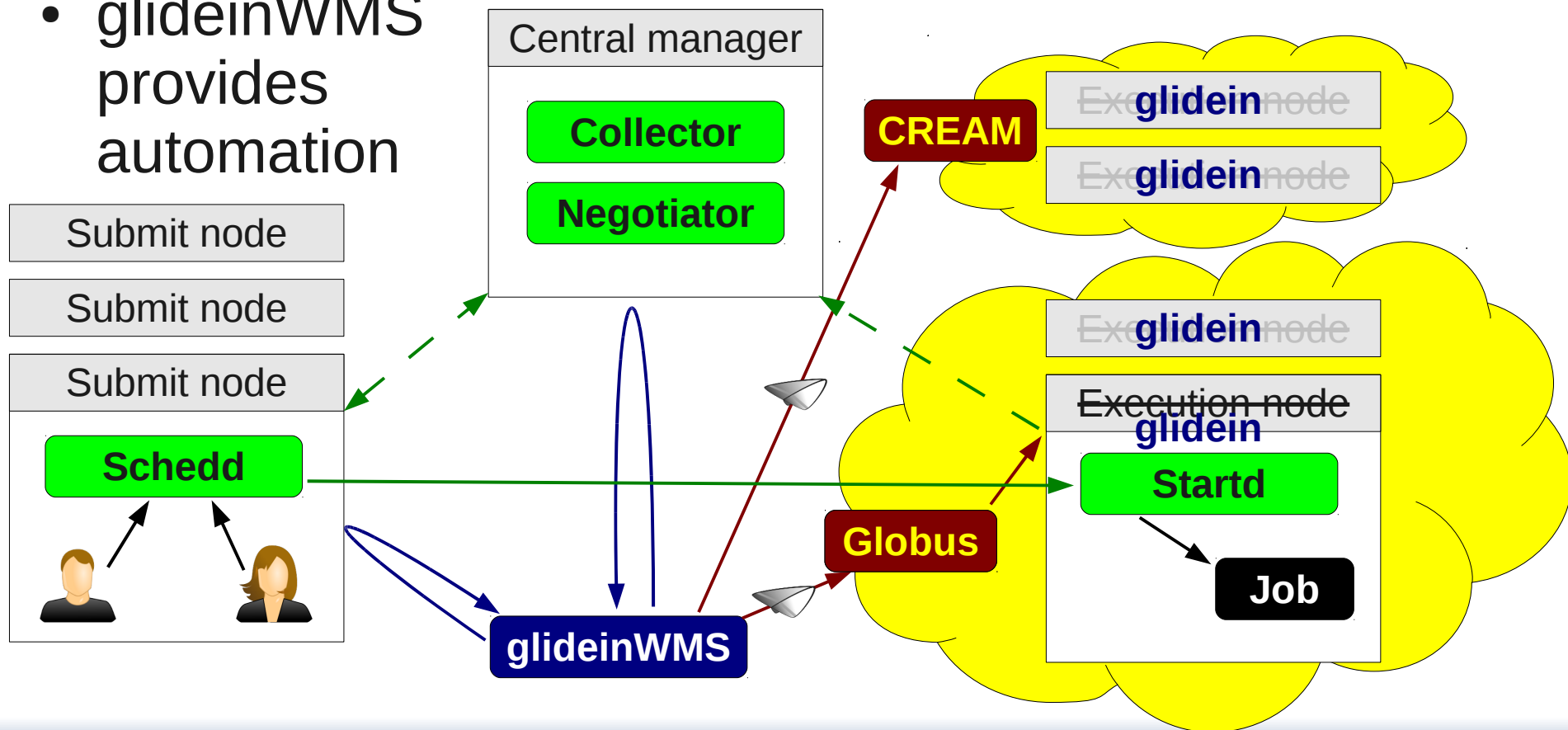
*The CMS T3 use case*

by Igor Sfiligoi (UCSD)

# Refresher - Glideins

- A glidein is just a properly configured Condor execution node submitted as a Grid job

- glideinWMS provides automation



# Base glidein philosophy

Pilot philosophy, really

**Gather resources using a generic credential and then regulate internally the sharing between final users**

# Good or bad idea

- **Definitely a good idea**, most of the time
  - Allows for a centralized policy setting
  - Instantaneous changes
  - Very flexible

# Good or bad idea

- Definitely a good idea, **most of the time**
  - Allows for a centralized policy setting
  - Instantaneous changes
  - Very flexible
- **BUT!**
  - Some sites may still want to give **more priority to some users!**
  - **They will not be happy to not be able to do it!**


# Why special users?

**Those users have bought  
those resources  
out their own pockets**  
(or equivalent)

# How to solve this edge case?

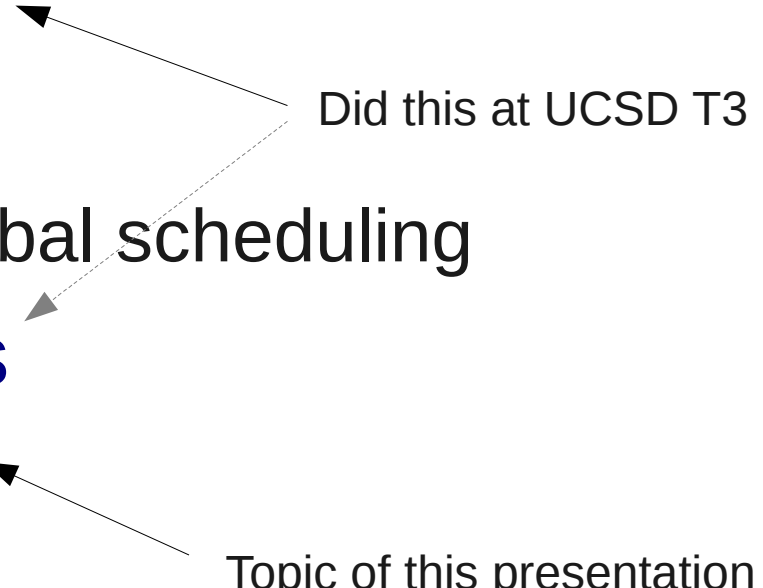
- Users **submit directly** to the site, bypassing the glideins
  - Possible, but not user friendly

# How to solve this edge case?

- Users **submit directly** to the site, bypassing the glideins
    - Possible, but not user friendly
  - Set up a **separate glidein pool** for just these users
    - One x Site!
    - High total overhead, no global scheduling
- Did this at UCSD T3
- 



# How to solve this edge case?

- Users **submit directly** to the site, bypassing the glideins
    - Possible, but not user friendly
  - Set up a **separate glidein pool** for just these users
    - One x Site!
    - High total overhead, no global scheduling
  - Implement **special policies** in the central glidein pool
    - More complex setup
- Did this at UCSD T3
- Topic of this presentation
- 

# How do we know who is special?

- Frontend admin will need to know who is special – and for what site
- Many ways to do it
  - **Explicit lists maintain in the frontend config**
    - Not scalable/unmaintainable
  - **Attribute set at submission**
    - Problem of trust (no authZ at condor level)
    - If automated at submission, just moved the problem
  - **VOMS credentials**
    - Needs support at VO level
    - Possibility of conflict (FQANs used by other services)

# How do we know who is special?

No ideal solution

- Frontend admin will need to know who is special – and for what site
- Many ways to do it
  - **Explicit lists maintain in the frontend config**
    - Not scalable/unmaintainable

Great for small scale
  - **Attribute set at submission**
    - Problem of trust (no authZ at condor level)
    - If automated at submission, just moved the problem

Easiest to config
  - **VOMS credentials**
    - Needs support at VO level
    - Possibility of conflict (FQANs used by other services)

Most promising for large scale

# Special users in a glidein pool

Assuming you know how to identify/group them

- What does it mean to have special users?
  - Some glideins start at a site because they are meant to be used by the special users
    - **Special proxy**
  - The special glideins **must run only** jobs from those **special users!**

# Special proxies

- Two options
  - One special proxy for all sites
    - Simpler
    - But cannot have two “special groups” at a single site
  - Site specific proxy
    - Lots of work for the frontend admin!
- Real life will likely have a mix of the two
  - “Master special proxy” for most sites
  - Plus a few site specific ones

# Special glidein matching

- Must implement policy to **only** match special users to special glideins
  - And remember we have really groups of special users, potentially one per site
- Has to whitelist site special users in site matching policy
  - Both during Site matchmaking (in the frontend)
  - And glidein matchmaking (in the START expr.)
- Exact syntax depends on how we identify special users

# Example with explicit user list

- Will need one frontend group per site
- For each group
  - Filter out special users at condor\_q  
`stringListMember(Owner, "user1, user2, user3", "", "")`
  - Start expression  
`stringListMember(Owner, "user1, user2, user3", "", "")`



Each site can have  
its own "special proxy"

# Example config with attribute

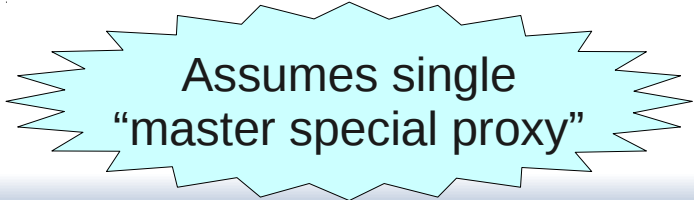
- Assumes the use of special attribute
  - User explicitly tells us where he is special

- Match expression

```
glidein["attrs"]["GLIDEIN_Site"] in job["Special At_Sites"].split(",")
```

- Start expression

```
stringListMember(GLIDEIN_Site, Special_At_Sites, ",")=?=True
```



Assumes single  
“master special proxy”



# Example config with VOMS

- Needs Factory support for reasonable frontend config

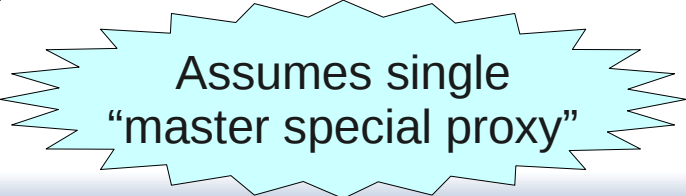
- Special role advertised as e.g.  
CMS\_SiteRole

- Match expression

```
glidein["attrs"]["CMS_SiteRole"] in job["x509UserProxyFQAN"].split(",")
```

- Start expression


```
stringListMember(CMS_SiteRole,x509UserProxyFQAN,",")=?=True
```



Assumes single  
"master special proxy"

# Priority considerations

- Global fair share may be a screwed
  - Without further tweaking, special use would **lower** user priority toward **global use**
- Condor provides a (partial) solution with SlotWeight
  - By setting `SlotWeight=0.001` for special glideins, **global share will not suffer** (too much)
  - **But global fair share will still influence special fair share**
    - Maybe not a problem?



No idea  
how to get  
both right

# Security considerations

- We now have glideins with different privileges in the same Condor pool
  - Increased risk of misuse
- But the “special glideins” just affect priority
  - Do not grant access to special files etc.
  - So **should be acceptable**

# Follow up discussion

# 4<sup>th</sup> alternative

- Eric proposed a 4<sup>th</sup> alternative
  - A site publishes the list of user DN's that are special
    - This is the “poor man” alternative to VOMS attributes
  - This information would somehow need to get to the Factory admins:
    - Should we find space in the InfoSys (e.g. BDII)?
    - Discovered on the WN (like CMSSW now)?  
(then somehow propagated to the Factory?)
    - Simply oral communication?

# Example for 4<sup>th</sup> alternative

- **Factory** advertises special users as e.g. `SpecialUsers`

- **Match expression**

```
job["x509userproxysubject"] in glidein["attrs"]["SpecialUsers"].split(",")
```

- **Start expression**

```
stringListMember(x509userproxysubject, SpecialUsers, ",")=?=True
```

# Pointers

- The official project Web page is <http://tinyurl.com/glideinWMS>
- glideinWMS development team is reachable at [glideinwms-support@fnal.gov](mailto:glideinwms-support@fnal.gov)

# Acknowledgments

- The glideinWMS is a CMS-led project developed mostly at FNAL, with contributions from UCSD and ISI
- The glideinWMS factory operations at UCSD is sponsored by OSG
- The funding comes from NSF, DOE and the UC system