CLS: Cat-Based Localization System

CRAIG STEFFEN, NCSA

HUST WORKSHOP SUPERCOMPUTING 2025
ST. LOUIS, MISSOURI, USA

parfu.net

CLS History

- Parfu development 2018-2022
- Testing via EXCEDE and then ACCESS allocations 2023/24
- Bechmark testing on ACCESS systems:
  - Darwin
  - Bridges
  - Ookami
  - Rockfish
  - Anvil
  - Frontera
  - Expanse

CLS Question (from parfu testing on 6+ systems):

- How to easily localize the same simple script on many systems?
- How can we make this easier?

Job Script Parsing: Two Passes

```
#!/bin/bash
#SBATCH --account=grpXXX
#SBATCH --mailuser=Alice@gmail.com
#SBATCH --time=2:00:00
LOGDIR=/work/grpXXX/data
MYITERS=5
srun myexec ${LOGDIR} n=${MYITERS}
```

```
#!/bin/bash
#SBATCH --account=grpXXX
#SBATCH --mailuser=Alice@gmail.com
#SBATCH --time=2:00:00
LOGDIR=/work/grpXXX/data
MYITERS=5
srun myexec ${LOGDIR} n=${MYITERS}
```

Job Script has two sections:
Newly Define: Static Section and Active Section

```
#!/bin/bash
#SBATCH --account=grpXXX
#SBATCH --mailuser=Alice@gmail.com
#SBATCH --time=2:00:00
LOGDIR=/work/grpXXX/data
MYITERS=5
srun myexec ${LOGDIR} n=${MYITERS}
```

←Static section

Job Script has two sections:
Newly Define: Static Section and Active Section

```
#!/bin/bash

#SBATCH --account=grpXXX

#SBATCH --mailuser=Alice@gmail.com

#SBATCH --time=2:00:00

LOGDIR=/work/grpXXX/data

MYITERS=5

srun myexec ${LOGDIR} n=${MYITERS}
```

←Active section

Job Script 3 Statement Roles: System Localization Statements

```
#!/bin/bash
#SBATCH --account=grpXXX
#SBATCH --mailuser=Alice@gmail.com
#SBATCH --time=2:00:00
LOGDIR=/work/grpXXX/data
MYITERS=5
srun myexec ${LOGDIR} n=${MYITERS}
```

← system localization (in the static section)

← system localization (in the dynamic section)

Job Script 3 Statement Roles: User or Personal Statements

```
#!/bin/bash

#SBATCH --account=grpXXX

#SBATCH --mailuser=Alice@gmail.com
```

←personal (in the static section)

```
#SBATCH --time=2:00:00

LOGDIR=/work/grpXXX/data

MYITERS=5
```

←personal (in the dynamic section)

```
srun myexec ${LOGDIR} n=${MYITERS}
```

Job Script 3 Statement Roles:
Core or Algorithm Statements

```
#!/bin/bash

#SBATCH --account=grpXXX

#SBATCH --mailuser=Alice@gmail.com

#SBATCH --time=2:00:00                       ←core (in the static section)

LOGDIR=/work/grpXXX/data

MYITERS=5

srun myexec ${LOGDIR} n=${MYITERS}           ←core (in the dynamic section)
```

2 Sections & 3 Roles:
Script exists as 6 Separate Script *Fragments*

```
#!/bin/bash
#SBATCH --account=grpXXX                ←Static System
#SBATCH --mailuser=Alice@gmail.com      ←Static Personal
#SBATCH --time=2:00:00                  ←Static Core
LOGDIR=/work/grpXXX/data                ←Active System
MYITERS=5                               ←Active Personal
srun myexec ${LOGDIR} n=${MYITERS}      ←Active Core
```

CLS Implementation:
Core Script file as Two fragments

- myscript.CLS:

```
#SBATCH --time=2:00:00
===========
srun myexec ${LOGDIR} n=${MYITERS}
```

CLS Implementation:
Other Four Fragments as Separate Files on Disk

Localization and personal files referred to in environment variables:

CLS_STATIC_SYS_FILE

CLS_ACTIVE_SYS_FILE

CLS_STATIC_PERS_FILE

CLS_ACTIVE_PERS_FILE

CLS Implementation Example: System Localization Fragment

- system_local.CLS

```
#!/bin/bash

#SBATCH --account=grpXXX
```

CLS Implementation:
Final Script Synthesized from Fragments

- $ cls_go.pl myscript.CLS > my_job_script.bash
  - (the other four fragments are pulled in according to environment variables)
- cls_go.pl combines the two fragments in the core file with
  - the two localization fragment files and
  - the two personalization files
- the two localization files and the two personalization files must exist but may be empty

CLS Deployment Strategy

- myscript.CLS can be deployed (in git hub) identically across many systems and users
- Each system has a separate pair of localization files to set up the main script for a given group on that system.
- For instance, the **static localization** fragment will have:
  - #SBATCH --account=   statement for that groups allocation
  - #SBATCH --partition=   specifying what partition that group uses

Advantages of CLS

- Core file can be functional and completely independent of user or which system it’s run on
- Localization can be created once per group for multiple users
- Users can manage their own configuration information
- Users can still have per-user per-run tweaks in their own files, independent of localization and core scripts
- New versions of the core script can be rolled out with (ideally no) site-tweaking

CLS Use History

- Deployed in a Contracted Project
- Significantly aided my testing of the parfu project (which has now ended)
  - I was able to deploy several rapid-fire changes to the main archive test script without having to re-localize it to each system

Download CLS (Includes Example Files)

- https://github.com/ncsa/parfu_archive_tool/tree/main/CLS
- (linked at parfu.net)
- **parfu_archive_tool**/CLS/system_loc/rockfish/**loc_active_rockfish.CLS**

```
FP_SYSTEM="rockfish“
# system-specific file system array definitions go here
TAR_DIR="${DATA}/transfer“
TAR_FS="G“
TAR_STRIPE="0“
ARC_DIR[0]="${DATA}/arc“
TGT_DIR[0]="${DATA}/data"
```

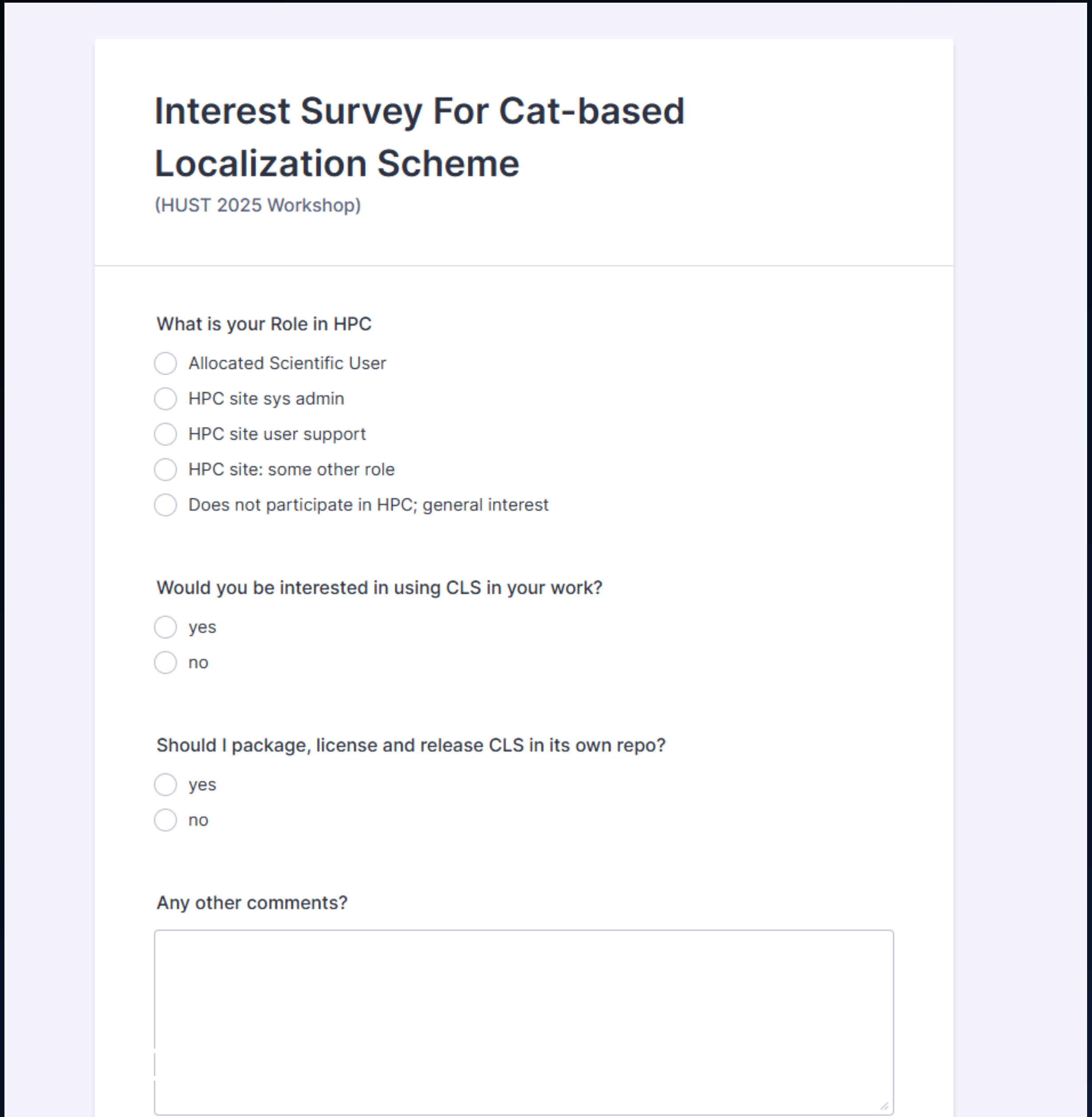
Question For The Audience

- Would it be worth packaging CLS ***Separately?***

Please take the survey: https://form.jotform.com/253194578627066

- **also linked at parfu.net**

Interest Survey



Interest Survey For Cat-based Localization Scheme

(HUST 2025 Workshop)

What is your Role in HPC

○ Allocated Scientific User
○ HPC site sys admin
○ HPC site user support
○ HPC site: some other role
○ Does not participate in HPC; general interest

Would you be interested in using CLS in your work?

○ yes
○ no

Should I package, license and release CLS in its own repo?

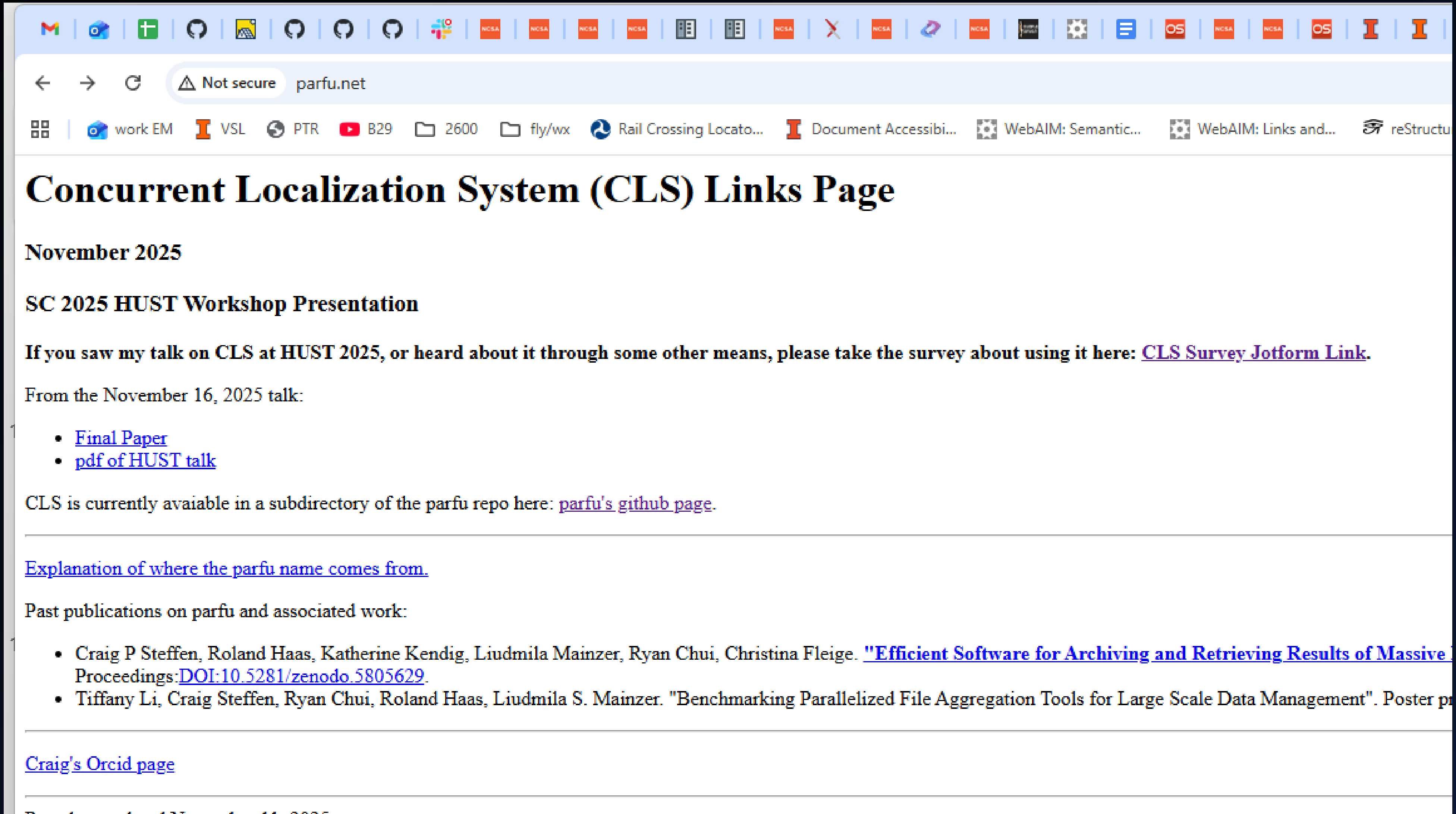
○ yes
○ no

Any other comments?

Links and Information

- parfu.net has:
  - A link to the survey
  - The final paper
  - A pdf of this talk
  - Other links to related material

parfu.net Front Page



Concurrent Localization System (CLS) Links Page

November 2025

SC 2025 HUST Workshop Presentation

If you saw my talk on CLS at HUST 2025, or heard about it through some other means, please take the survey about using it here: CLS Survey Jotform Link.

From the November 16, 2025 talk:

- Final Paper
- pdf of HUST talk

CLS is currently avaiable in a subdirectory of the parfu repo here: parfu's github page.

Explanation of where the parfu name comes from.

Past publications on parfu and associated work:

- Craig P Steffen, Roland Haas, Katherine Kendig, Liudmila Mainzer, Ryan Chui, Christina Fleige. **"Efficient Software for Archiving and Retrieving Results of Massive** Proceedings:DOI:10.5281/zenodo.5805629.
- Tiffany Li, Craig Steffen, Ryan Chui, Roland Haas, Liudmila S. Mainzer. "Benchmarking Parallelized File Aggregation Tools for Large Scale Data Management". Poster pr

---

Craig's Orcid page

Thanks: parfu development

- Parfu was developed as part of the **Blue Waters** sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) the State of Illinois, and as of December, 2019, the National Geospatial-Intelligence Agency. Blue Waters was a joint effort of the University of Illinois at Urbana-Champaign and its National Center for Supercomputing Applications.
- 1. Brett Bode, Michelle Butler, Thom Dunning, William Gropp, Torsten Hoefler, Wen-mei Hwu, and William Kramer (alphabetical). *The Blue Waters Super-System for Super-Science*. Contemporary HPC Architectures, Jeffery Vetter editor. Sitka Publications, November 2012.Edited by Jeffrey S . Vetter, Chapman and Hall/CRC 2013, Print ISBN: 978-1-4665-6834-1, eBook ISBN: 978-1-4665-6835-8
- 2. Kramer, William, Michelle Butler, Gregory Bauer, Kalyana Chadalavada, Celso Mendes, *Blue Waters Parallel I/O Storage Sub-system*, High Performance Parallel I/O, Prabhat and Quincey Koziol editors, CRC Publications, Taylor and Francis Group, Boca Raton FL, 2015, Hardback Print ISBN 13:978-1-4665-8234-7.

Thanks: CLS development and presentation

- *This research used the Delta advanced computing and data resource which is supported by the National Science Foundation (award OAC 2005572) and the State of Illinois. Delta is a joint effort of the University of Illinois Urbana-Champaign and its National Center for Supercomputing Applications.*

Thanks to People and Organizations

- Greg Bauer
- Bill Kramer
- Brett Bode
- National Science Foundation
- HUST Committee
- Submission Reviewers