28th Annual I IEEE High Performance Extreme Computing Virtual Conference | 23 - 27 Sept 2024

# A Deployment Tool for Large Scale Graph Analytics Framework: Arachne

Garrett Gonzalez-Rivas, Zhihui Du, David A. Bader

Department of Data Science

New Jersey Institute of Technology

Newark, NJ, USA

Supported by NSF grant: CCF-2109988



## **Motivations**

- 1. Existing tools (such as NetworkX) cannot scale to handle the size of graphs/datasets that modern data scientists need to work on.
- 2. Highly productive tools must be able to integrate easily into existing workflows and working environments.
- 3. 'Time-to-productivity' is an important metric to consider when adopting a solution.



## Need

arkouda//st/neutral/instruction/sg.chp//home/michaels/key/arkouda//src/Seguence#igo.chp//home/michaels/home/michae rtMsg.chpl /home/michaelshewarega/Desktop/test/arkouda//s rc/StatSMsg.chpl/home/michaelshewarega/Desktop/test/arkouda//s rc/StatSMsg.chpl/home/michaelshewarega/Desktop/test/arko uda//src/TimeClasSMsg.chpl/home/michaelshewarega/Desktop /test/arkouda//src/TransferMsg.chpl/home/michaelshewareg/mstall/linx Versions of Clibe educating was by thousings , dow/shortsmicrosoftest/arkouda//src/Induc/shcwareg/m/install/linx free\_formerset /a/Desktop/test/arkouda//src/UniqueMsg.chpl /home/michaelsexareg/m/install/linx free\_formerset /a/Desktop/test/arkouda//src/IniqueMsg.chpl /home/michaels ;/flat -I/home/garretgg/Desktop/test/arkouda//src/IniqueMsg.chpl /home/michaels ;/flat -I/home/garretg /amms.chpl =MawArravMimmel ==Support/linterum\_espungst untime//include/gpu/none -I/home/garrettgr/chapel-1.31.0/ -sMaxArravDims=1 -sSupportsUint8=true -sSuppor ethtstätisted beckbindiatildjedig expandesi heseivat a charecoelides sUint16=false -sSupportsUint32=false -sSupportsUint64=tru -1.31.0/runtime -e -sSupportsInt8=false -sSupportsInt8=false -sSupportsInt1tgr/chapel-1.31.0/runtime//include/tasks/filo -1/home/ga t32=false -sSupportsInt64=true -sSupportsFloat32=false -s ≥/garrettgr/chapel-1.31.0/runtime//include/qio -I/home/ga SupportsInt64=true -sSupportsBool=true -o arkouda\_server s/cstdlib -I/home/garrettgr/chapel-1.31.0/runtime//include/qio -I/home/ga coline a second states and the pip init : 0.088 seconds /third-party/utf8-decoder parseAndConvertUast : 2.863 seconds 4/cpu-native/loc-flat/comm-nore/tasks-fifo/tmr-generic/un checkUast : 0.171 seconds a**pepelinterno reantitati a traditati a periode a seconda a seconda a seconda a seconda a seconda a seconda a s** one/re2-bundlco/fa-mone/lab\_pic+nore readExternC : 0.014 seconds localeModels/ilat -I/home/gariettgr/chap expandExternArrayCalls : 0.093 seconds cleanup : 0.135 seconds l-1.31.0/runtime/include/comm/none -I/home/garrettgr/chap coneResolve : 2.072 seconds Anleoked.rather than doing so in Arachne's tgr/chapel-1.31.0/runtime/include/tasks/fifo -I/home/gar flattenClasses : 0.017 seconds arrettgr/chapel-1.31.0/Funtime/include/gio -I/home/garret normalize : 2.764 seconds checkNormalized : 0.048 seconds ilib -I/home/garrettgr/chapel 1.31.0/runtime/include/mem/ buildDefaultFunctions : 0.268 seconds -party/utf8-decoder -I/home/garrettgr/chapel-1.31.0/third directory and then going back to Arkouda's reateTaskFunctions : 0.108 seconds haelshewarega/Desktop/test/arkouda-njit/arachne/ -none/inclu<mark>d</mark>e -I/home/garrett<mark>c</mark>r/ch<u>apel-1.31.0/third-party</u> erver/Utils.chpl:17: In function 'fastLocalSubdomain': ,include -I/home/garrettgr/charel-1.31.0/third-party/gmp/i /home/michaelshewarega/Desktop/test/arkouda-njit/arachne/ test/arkouda-std=c++17 -c /home/gariettgr/arkouda-2023.10.06//sr server/Utils.chpl:19: error: unresolved call 'unmanaged da-std=c++17 -c /home/gariettgr/arkouda-2023.10.06//sr Jda-2023.10.06//src/ArrowFunctions.o -I/home/garrettgr/ana schPL\_HOME/modules/dists/BlockDist.chpl:546: note: this cattgr/anaconda3/envs/arkouda-cev/lib home/michaelshewarega/Desktop/test/arkouda-njit/arachne/ directory to run the command. ichaelshewarega/Desktop/test/arkouda-njit/arachne/ npilation: '-L/home/garrettgr/anaconda3/envs/arkouda-dev/ te did not match: BlockDom.locDoms Utils.chpl:19: note: because call includes 1 argum a-2023.10.06//src/ArrowFunctions.cpp:1: HOME/modules/dists/BlockDist.chpl:546: note: but <sup>vFunctions. h:7]</sup>heserstep/si-weret separatetep/cooleplatel2dagurs and 50 minutes can only accept 0 arguments 'home/michaelshewarega/Desktop/test/arkouda-njit/arachne erver/Utils.chpl:19: note: other candidates are: CHPL\_HOME/modules/dists/SparseBlockDist.chpl:89: note: before the error was revealed during the eBlockDom.locDoms SparseBioCockUom.toccUoms /home/michaelshewarega/Desktop/test/arkouda-njit/arachn e/server/Utils.chpl:65: called as fastLocalSubdomain(bloc j] Error 1 kArray: [domain(1,int(64),one)] int(64)) from function 'g arkouda-2023.10.06' final step of the installation. /home/michaelshewarega/Desktop/test/arkouda-njit/arachn Juda-2023.10.06//src/ArrowFunctions.o] Error 2 e/server/BuildGraphMsg.chpl:98: called as generateRanges( )uda-2023.10.06 graph: <u>shared SegGraph</u>, key: <u>string</u>, key2insert: <u>string</u>, array: <u>[domain(1,int(64),one)] int(64)</u>) generic instantiations are underlined in the above uch file or directory make: \*\*\* [Makefile:359: arkouda\_server] Error 1 make: Leaving directory '/home/michaelshewarega/Desktop/t

When installing Arachne, it is not uncommon to get an error several hours from when a mistake was made, and for the error to be seemingly unrelated.



# **Solution Requirements**

- Handle the installation, configuration, and building of Arachne automatically regardless of the system it is run on.
- Allow a developer or administrator to modify the environment (configuration options, packages, etc.) being built.
- Be easily kept up to date by the developers and allow for developers to perform automatic testing.



### **The Deployment Tool - Portability**

- Written in POSIX Shell, it is designed to be highly portable
  - it has been tested to run on various versions several major \*NIX-like operating systems, including several GNU/Unix distributions (RHEL, Ubuntu, and others) and BSDderivatives (FreeBSD and MacOS).

```
Function to collect system information and determine packaging type
information_collection() {
 if [[ -f /etc/os-release ]]; then
    . /etc/os-release
    case "$ID" in
   ubuntu | pop | neon | zorin | tuxedo)
      OS="ubuntu"
     if [[ "${UBUNTU_CODENAME:-}" != "" ]]; then
        VERSION="$UBUNTU_CODENAME"
      else
        VERSION="$VERSION_CODENAME"
      fi
     PACKAGE_TYPE="apt"
     ;;
    debian)
      OS="$ID"
      VERSION="$VERSION_CODENAME"
      PACKAGE_TYPE="apt"
     ;;
    centos)
      OS="$ID"
      VERSION="$VERSION_ID"
      PACKAGE_TYPE="dnf"
     if [[ "$VERSION" = "7" ]]; then
        PACKAGE_TYPE="yum"
      fi
      ;;
    ol)
      OS="oracle"
     VERSION="$(echo "$VERSION_ID" | cut -f1 -d.)"
      PACKAGE_TYPE="dnf"
     if [[ "$VERSION" = "7" ]]; then
       PACKAGE_TYPE="yum"
      fi
      ;;
```



### The Deployment Tool – Ease of Use

- By leveraging flags, a single command can be copied from documentation or as created by an administrator to install a desired installation without any further user-involved other than entering their password:
- -u | --unattended : whether to handle the installation automatically or not
- -r | --release : whether to use a stable, rolling, or 'developer' release
- -e | --environment : whether to create a user or developer environment



# **The Companion File**

 Set what packages to install, what version to install (or several by leveraging the

\$RELEASE\_TYPE variable)

[chapel] versions = { stable = "1.31.0", rolling = "1.33.0", fallback = "1.32.0" } download\_url = "https://github.com/chapel-lang/chapel/releases/download/{version}/chapel-{version}.tar.gz" env\_vars = { CHPL\_HOME = "{INSTALL\_LOC}/chapel", CHPL\_GMP = "bundled", CHPL\_LLVM = "system", CHPL\_RE2 = "bundled", CHPL\_COMM = "none", ARKOUDA\_QUICK\_COMPILE = "true" [arkouda] versions = { rolling = "2024.04.19", stable = "2023.10.06", fallback = "2023.11.15" } download\_url = "https://github.com/Bears-R-Us/arkouda/archive/refs/tags/v{version}.tar.gz" env\_vars = { AK\_HOME = "{INSTALL\_LOC}/arkouda" } [anaconda] linux\_urls = { x86\_64 = "https://repo.anaconda.com/archive/Anaconda3-2024.02-1-Linux-x86\_64.sh", aarch64 = "https://repo.anaconda.com/archive/Anaconda3-2024.02-1-Linux-aarch64.sh" macos urls = { arm64 = "https://repo.anaconda.com/archive/Anaconda3-2024.02-1-MacOSX-arm64.sh", x86\_64 = "https://repo.anaconda.com/archive/Anaconda3-2024.02-1-MacOSX-x86\_64.sh"

- Determine the location from which to install packages from
- Define what, and the value of, the environment variables to be set
- Override what functions are run, the order of different steps, etc.



## **Real-World Results**

#### Without the deployment tool:

- Several PhD students (both at our lab and collaborating universities) took weeks getting a working Arachne installation, new undergraduates and other interns often didn't work with Arachne directly.
- When collaborating with Ivy league universities, even individuals with technical backgrounds took **nearly a week** and **over a hundred back-and-forth messages** with the developers at our lab to get a working Arachne installation.

#### With the deployment tool:

- While errors are still sometimes encountered, several members of our lab (and even high school interns) have been able to install Arachne from a single command – across three different operating systems and several different desired environment configurations.
- When issues were encountered, they were usually solved in just a few messages, and some were solved by the intern themselves while waiting for a response. N I I T

New Jersey Institute

of Technology

## What comes next?

- There are several improvements planned for the deployment tool, especially for the script generator and its integration with GitHub actions.
- New errors and improvements are being discovered as it is being run on more systems, as we have more users, it will become more stable.
- There have been several feature requests already, including allowing it to install other modules for the Arkouda/Arachne framework, which offer additional functionality such as: multiplexed and bidirectional communications with the Arachne server, integrations with Kubernetes/Helm, and integrations with Prometheus for better monitoring the sever.



## Conclusion

- Allows for individuals not experienced in system administration to install and configure Arachne on their machine.
  - In unattended mode, the total interaction time by the user often less than one minute.
- Allows for better testing (CI/CD) of Arachne to ensure compatibility with different packages / dependencies.
- The tool should be incorporated into the Arachne GitHub repository for public use soon
  - Along with new and updated documentation for installing, configuring, and using Arachne (both manually and with the deployment tool).



# **Thank You!**

# **Questions?**



September 2024

## What about containerization?

- Containerization is powerful, however, poses certain restrictions, especially for particularly resource intensive applications, which Arachne absolutely can be.
- Local installation allows for better control over the configuration, direct access to system resources, and better integration with other tools on the system.
- The script generator for the deployment tool should be able to generate the dockerfiles to containerize Arachne, which will also be offered if that is preferred.



# What about Ansible?

- Ansible playbooks and Nix/Nix Flakes were inspirations for the the deployment tool and its companion file
- Ansible must be installed and setup on the host system (managed node), and is often expected to be used in conjunction with a control node.
- While Ansible can be used with a premade playbook rather than a control node, it does not offer the same ability to be customized in an interactive installation or through passing flags for an unattended installation.

