Salvus

Opis

Salvus je set alata za analizu seizmičkih podatka.

Verzije

Na računalnom klasteru dostupna je verzija 0.12.14.

Licenciranje

Salvus je komercijalan program te je za pokretanje istok potrebna licenca. Srce nema vlastitih licenci za pokretanje programa, već je na korisnicima da sami priskrbe svoje licence.

Korištenje

Salvus je zamišljen da se koristi na lokalnom računalu, ali da se računalno najzahtjevniji dio izvodi na računalnim klasterima. Kako bi korisnici mogli koristiti sve alate moraju zadovoljiti nekoliko stvari:

- Posjedovati licencu za korištenje Salvus programa (.toml datoteka)
- Imati spremnu python virtualnu okolinu na osobnom računalu
- Imati ssh pristup računalnom klasteru na kojem planiraju pokretati SalvusCompute
- Imati odgovarajuću konfiguraciju za klaster na kojem planiraju koristiti Salvus
- Inicijalizirati klaster kao lokaciju na kojoj će se izvoditi kalkulacije

Python okolina

Python okolinu je najjednostavnije pripremiti prema uputama na stranicama programa.

Kad se uspješno preuzmu svi potrebni alati potrebno je inicijalizirati lokaciju.

Inicijalizacija lokacije

Prvo je potrebno locirati konfiguracijske datoteke. S aktiviranom salvus virtualnom okolinom potrebno je izvršiti naredbe:

lociranje config datoteka

```
salvus-cli print-config-paths
```

Primjer ispisa

```
salvus-cli print-config-paths

(salvus_12) [korisnik@salvus_stroj ~]$ salvus-cli print-config-paths
  Config file: /home/korisnik/.config/SalvusFlow/1.0/salvus-flow-config.toml
  DB file: /home/korisnik/.local/share/SalvusFlow/1.0/salvus-flow-db.sqlite
```

Lokaciju je moguće incijalizirati na tri načina:

1. koristeći interaktivnu opciju koja se pokreće sa:
   salvus-cli add-site
2. koristeći interaktivni builder na stranicama programa
3. Kopirati konfiguraciju s wiki stranice i urediti je prema uputama

Opcija jedan je jednostavna, ali ne omogućava izravno potpuno točnu definiciju, već je potrebno ručno doraditi konfiguracijsku datoteku.

S opcijom dva je moguće izravno dobiti bolju definicijsku datoteku nego opcijom jedan, ali i dalje nisu unesene sve opcije te je potrebno ručno uređivati datoteku *salvus-flow-config.toml*

Opcija tri traži najmanje posla od korisnika. Potrebno je urediti login podatke, i adrese direktorija u koje će se spremati podaci i iz kojih će se pokretati poslovi.

Primjer definicijske datoteke

Primjer konfiguracijske datoteke za dodavanje lokacije *supek_cpu*. Sadržaj ove datoteke korisnik mora kopirati u svoju *salvus-flow-config.toml* definicijsku datoteku. Nakon toga potrebno je urediti točnost određenih podataka:

Obavezne izmjene:

1. Podesiti korisničke direktorije
   a. run_directory
   b. tmp_directory
2. Podesiti korisničko ime računa kojim se spaja na Supeka

Opcionalne izmjene:

1. Red na koji se šalje
2. max_ranks - ako licenca dozvoljava više od 64 jezgre
3. parametre okoline, ali tada korisnik mora osigurati da se ta okolina priprema na Supeku prilikom pokretanja posla

Upozorenje

Važno je napomenuti kako izuzev pristupnih čvorova, drugi čvorovi nemaju pristup internetu, tako da nije moguće preuzimanje podataka kroz dijelove poslova koji se pokreću na Supeku. Također, salvus mora biti u "offline" modu, koristeći tokene umjesto izravnog kontakta sa serverom za licenciranje. Ako se inicijalizira lokacija na Supeku, a da za nju nije postavljeno *use_license_tokens = true* salvus je neće moći koristiti.

Primjer konfiguracije za Supeka

```
(salvus_12) [korisnik@osobno_racunalo 1.0]$ cat salvus-flow-config.toml
    [sites.supek_cpu]
        # ############################################################################
        # > The site type must be given.
        site_type = "pbs"
        # ############################################################################
        # > The default number of processors Salvus will be run with on this
        # > site, if not explicitly specified.
        default_ranks = 1
        # ############################################################################
        # > The maximum number of processors Salvus will be run with on this
        # > site. Serves to protect against accidental user errors.
        max_ranks = 512
        # ############################################################################
        # > Absolute path to the Salvus binary on the site.
        salvus_binary = "/apps/scientific/salvus/0.12.14/bin/salvus"
        # ############################################################################
        # > Run directory on the site. Salvus will copy all files it requires to
        # > run into this directory. Must be read-and writeable.
        run_directory = "/lustre/home/korisnicko_ime/salvus_data/run_directory"
        # ############################################################################
        # > Run directory on the site. Use for very large and/or temporary
        # > output. Must be read-and writeable. If site's systems has a scratch
        # > file system use it here.
        tmp_directory = "/lustre/home/korisnicko_ime/salvus_data/tmp_directory"
        # ############################################################################
        # > Update interval in seconds for commands that have to periodically
        # > ping the remote site. Don't choose this value too small for shared
        # > machines to conserve resources. We recommend 30.0 seconds for sites
        # > with a job queuing system and around 1.0 to 5.0 seconds for direct
        # > execution sites depending on the expected average job size.
        # update_interval_in_seconds = 30.0
        # ############################################################################
        # > Only use this on dark sites (e.g. sites with no internet access). If
        # > this is set to True, Salvus will negotiate a license token with
        # > Mondaic's license server and send along the token.
        use_license_tokens = true
        # ############################################################################
        # > Set this to true if each compute node on the target site has at
        # > least one CUDA capable GPU accelerator attached and a CUDA driver
        # > version that is greater than or equal to 410.48 (CUDA 10.0).
        use_cuda_capable_gpus = false
        # ############################################################################
        # > Force usage of a login bash shell. A bit slower but sometimes
        # > necessary/easier if a system performs some essential setup in a
        # > login shell.
        # use_login_shell = false
        # ############################################################################
        # > Optionally specify environment variables to be set before every call
        # > on the site. Repeat this group as often as needed.
        # [[sites.supek_cpu.environment_variable]]
        #     name = "LD_LIBRARY_PATH"
        #     value = "/path/to/some/lib/dir"
        [sites.supek_cpu.ssh_settings]
            # ############################################################################
            # > Hostname of the site.
            hostname = "login-cpu.hpc.srce.hr"
            # ############################################################################
            # > SSH username of the site.
            username = "korisnicko_ime"
            # ############################################################################
            # > Manually enter the password every time an SSH connection to this
            # > site is established.
            # enter_password_manually = true
            # ############################################################################
```

```
    # > Fully interactive login for ever SSH connection to this site.
    # > This setting overwrites any manually entered password. This is
    # > necessary for example for interactive two-factor authentication
    # > schemes.
    # interactive_login = true
    # ######################################################################
    # > Any extra arguments that should be passed to
    # > paramiko.SSHClient.connect() when Salvus initially establishes
    # > the SSH connection. This will also be applied to any potential
    # > proxyjump hosts. This should rarely be necessary.
    # [sites.supek_cpu.ssh_settings.extra_paramiko_connect_arguments]
      #  banner_timeout = 20
[sites.supek_cpu.site_specific]
    # ######################################################################
    # > The number of processes used on each compute node. Best set this
    # > to the number of physical cores of a single node.
    tasks_per_node = 128
    # ######################################################################
    # > The memory per rank for a simulation. The total memory usage
    # > will then be calculated based on the number of ranks and is
    # > passed as 'mem=XXXXmb' to PBS' select statement. If not given it
    # > will not be set.
    # memory_per_rank_in_mb = 1024
    # ######################################################################
    # > The PBS queue in which to execute jobs.
    queue = "cpu"
    # ######################################################################
    # > Optionally specify the PBS queue in which to execute debug and
    # > info jobs.
    # debug_queue = "debug"
    # ######################################################################
    # > The variety of PBS implementations means that sometimes one must
    # > pass a custom string defining the compute resources to the
    # > scheduler. This settings allows you to overwrite the default
    # > (nodes={NODES}:ppn={RANKS}) with something else that will be
    # > passed in its stead. Available variables are `NODES`, `RANKS`,
    # > and `TASKS_PER_NODE`.
     compute_resources_template = "select={RANKS}:mem=10000mb"
    # ######################################################################
    # > Path to the folder with the PBS binaries, e.g. where qsub, qdel,
    # > and friends are located.
    path_to_pbs_binaries = "/opt/pbs/bin/"
    # ######################################################################
    # > Jobs run on PBS sites will be launched with `pbsrun` by default.
    # > A different launcher can be specified here. Available variables
    # > are `{NODES}`, `{RANKS}`, and `{TASKS_PER_NODE}` which will be
    # > filled automatically.
     replace_pbsrun_with = "mpiexec -np {RANKS}"
    # ######################################################################
    # > Environment variables are per-default set in the environment and
    # > then passed on to qsub. If this does not work, try setting this
    # > flag to True in which case the environment variables will be set
    # > directly in the job script as well.
    # set_environment_variables_in_submit_script = true
    # ######################################################################
    # > Run `module unload MODULE_NAME` before starting Salvus. Run
    # > after switching and before loading modules. Might be necessary
    # > for some slurm configurations.
    # modules_to_unload = ['MODULE_A', 'MODULE_B']
    # ######################################################################
    # > Run `module load MODULE_NAME` before starting Salvus. Run after
    # > switching and unloading modules. Might be necessary for some
    # > slurm configurations.
    modules_to_load = ['libs/salvus-mpi/salvus-mpi']
    # ######################################################################
    # > Run `module switch OLD_MODULE NEW_MODULE` before starting
    # > Salvus. Might be necessary for some slurm configurations. Repeat
    # > this group as often as needed.
    # [[sites.supek_cpu.site_specific.modules_to_switch]]
    #     old = "OLD_MODULE"
    #     new = "NEW_MODULE"
```

```
# ########################################################################
# > Additional arguments that will be added to the top of the
# > generated sbatch file in the format '-ARGNAME VALUE'. Might be
# > necessary for some pbs configurations. Repeat this group as
# > often as needed. Available variables are `{NODES}`, `{RANKS}`,
# > and `{TASKS_PER_NODE}` which will be filled automatically.
# [[sites.supek_cpu.site_specific.additional_qsub_arguments]]
#     name = "ARGNAME"
#     value = "ARGVALUE"
# ########################################################################
# > Additional arguments that will be added to the pbs/alternative
# > binary call in the form '--ARGNAME=VALUE' (or `-a v` in the case
# > of a single letter name). Might be necessary for some PBS
# > configurations. Repeat this group as often as needed. Available
# > variables are `{NODES}`, `{RANKS}`, and `{TASKS_PER_NODE}` which
# > will be filled automatically.
# [[sites.supek_cpu.site_specific.additional_pbsrun_arguments]]
#     name = "ARGNAME"
#     value = "ARGVALUE"
```

Nakon što se urede definicijske datoteke potrebno je izvršiti naredbu

inicijalizacija

```
salvus-cli init-site supek_cpu
```

Tom naredbom testirat će se ispravnost definicijskih datoteka, te će se poslati mali posao na dvije jezgre na Supeku. Ako se posao ispravno izvrši lokacija je uspješno inicijalizirana i može se početi koristiti.

U skriptama kojima se pokreče salvus, bitno je da se kao lokacija koristi ona koja se uspješno inicijalizirala.