CI Working Group
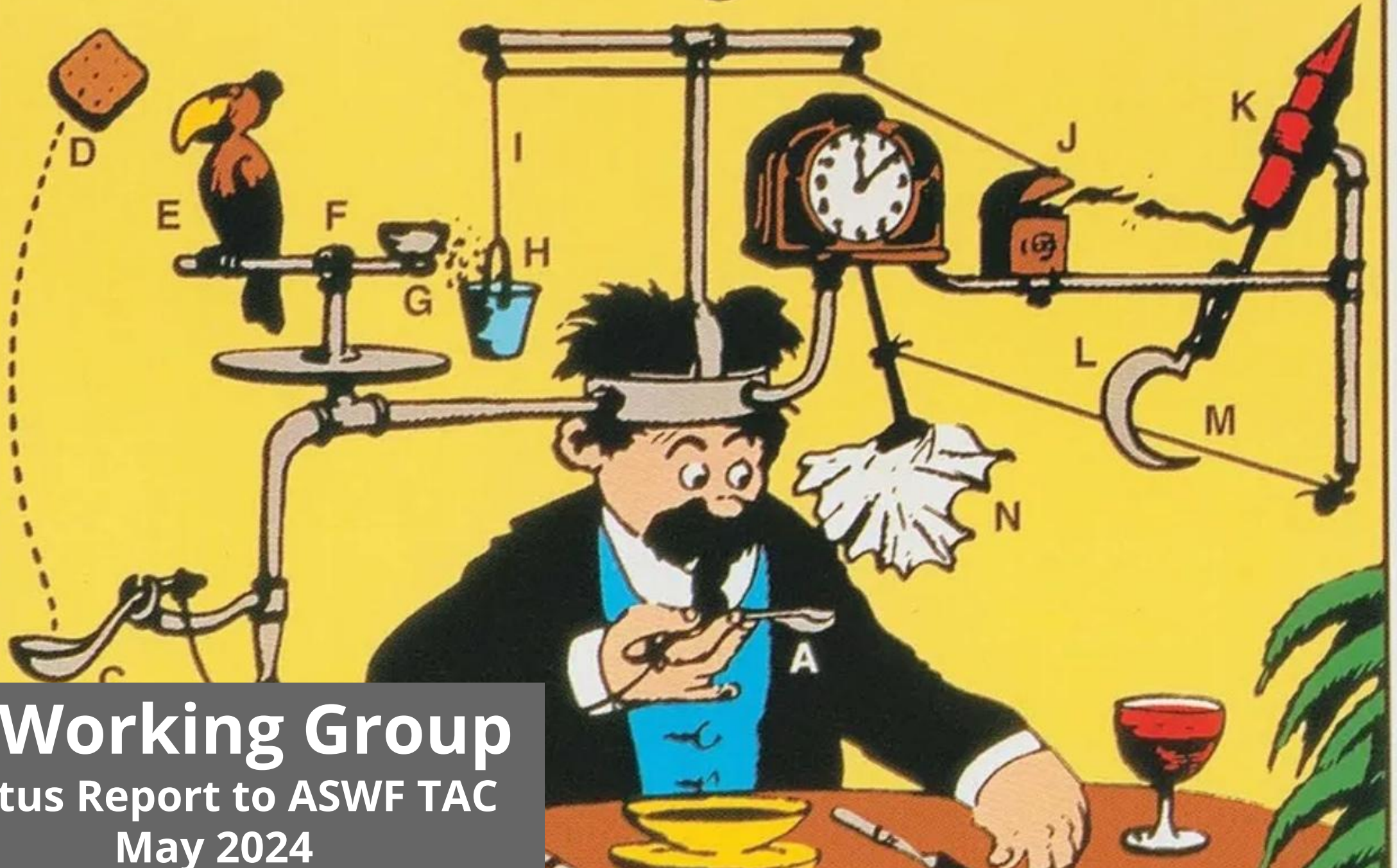Status Report to ASWF TAC
May 2024

Continuous Integration WG Mission Statement

The ASWF Continuous Integration (CI) Working Group is tasked to investigate and deliver tools, procedures and services used by ASWF projects in their development process, including:

- Revision control
- Build toolchains
- CI build environments
- Testing
- Packaging and distribution

Continuous Integration WG Mission Statement (cont)

- Provide a forum for the exchange of ideas between individuals interested in all aspects of the tooling and processes involved in software development, building and distribution
- Nothing we do is prescriptive, projects are free to adopt or not, and are encouraged to share infrastructure with other ASWF projects.

Our Meetings

- One meeting a month, off cadence from TAC (next is June 19th at 13:00 Pacific Time)
- On average 6-7 attendees over last 12 months, core group of frequent attendees
- A good place to interact with LF Release Engineering
- Jan 2024 had guests from GitHub Actions team discussing upcoming features
  - Hoping to make that a recurring event
- Meeting minutes in wg-ci repo: https://github.com/AcademySoftwareFoundation/wg-ci/tree/main/meetings

Slack: #wg-ci

- Slack Channel: #wg-ci 330 members (up from 240 last year)
  - reasonable traffic volume, enough to help fill meeting agenda, a good place to share an interest in tooling

Where to Get Help with Infrastructure

- #wg-ci for general discussion
- Linux Foundation Release Engineering Helpdesk:
  https://jira.linuxfoundation.org/plugins/servlet/desk
  - GitHub Permissions
  - Secrets Management
  - External Integrations
- Drop by our monthly call!

CI WG Wiki

https://wiki.aswf.io/display/CIWG

GitHub Free Runners

- Free runners adequate for many jobs
  - Recently upgraded: 4 core, 16GB RAM (was 2 cores, 7GB)
  - But still only 14GB disk, can be challenging
- Apple Silicon runners finally available (Jan 2024)
- No Windows/Linux on ARM yet
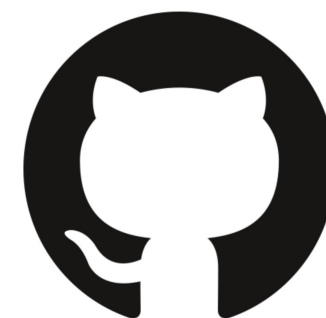
Virtual Machine	Processor (CPU)	Memory (RAM)	Storage (SSD)	OS (YAML workflow label)	Notes
Linux	4	16 GB	14 GB	`ubuntu-latest`, `ubuntu-24.04` [Beta], `ubuntu-22.04`, `ubuntu-20.04`	The `ubuntu-latest` label currently uses the Ubuntu 22.04 runner image.
Windows	4	16 GB	14 GB	`windows-latest`, `windows-2022`, `windows-2019`	The `windows-latest` label currently uses the Windows 2022 runner image.
macOS	3	14 GB	14 GB	`macos-12` or `macos-11`	The `macos-11` label has been deprecated and will no longer be available after 6/28/2024.
macOS	4	14 GB	14 GB	`macos-13`	N/A
macOS	3 (M1)	7 GB	14 GB	`macos-latest` or `macos-14`	The `macos-latest` label currently uses the macOS 14 runner image.

ASWF Enterprise GitHub Organization

- Higher limits on free GHA minutes, concurrent jobs
- Access to larger, for pay runners:
  - $1,500/month pre-authorized
  - Used by OpenVDB, aswf-docker for builds that wouldn't complete otherwise
  - Informally managed
  - Available to all projects, but to be used judiciously
- LF RelEng working on extending Enterprise org benefits to projects not under ASWF GitHub org

Access to Larger Runners

Access to Larger Runners
OpenVDB Weekly Workflow
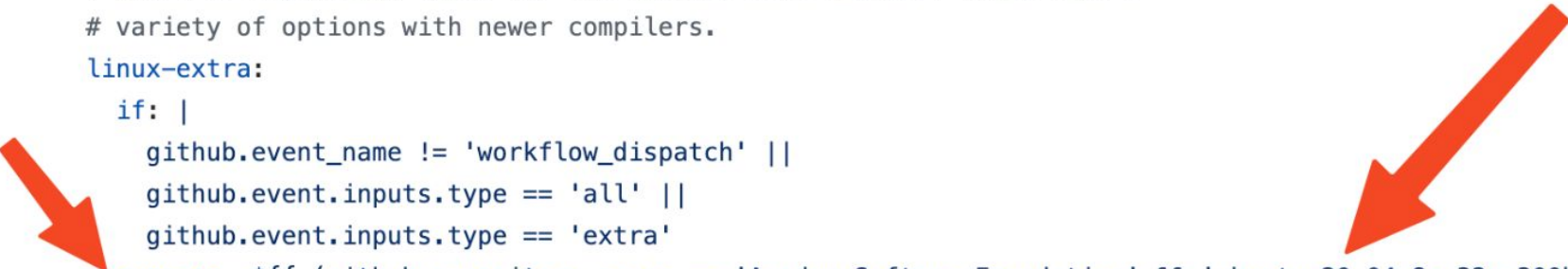
github.com/AcademySoftwareFoundation/openvdb/blob/master/.github/workflows/weekly.yml

ᛘ master ▾ **openvdb** / .github / workflows / **weekly.yml**
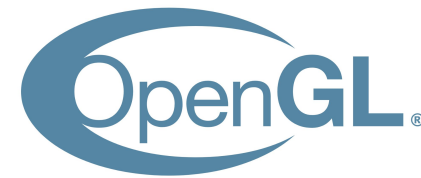
Code Blame 486 lines (467 loc) · 21.4 KB · 🔒

```
137      ###############################################################
138      ######################### Core Library Extras #################
139      ###############################################################
140
141      # Extra configuration tests for the OpenVDB Core library. These test a
142      # variety of options with newer compilers.
143      linux-extra:
144        if: |
145          github.event_name != 'workflow_dispatch' ||
146          github.event.inputs.type == 'all' ||
147          github.event.inputs.type == 'extra'
148        runs-on: ${{ (github.repository_owner == 'AcademySoftwareFoundation' && 'ubuntu-20.04-8c-32g-300h') || 'ubuntu-latest' }}
149        name: linux-extra:${{ matrix.config.name }}
150        container:
151          # @note  we specifically use clang15.0 (not clang15) here as the newest
152          #  versions of the clang15.X containers have some issues with the GLFW
153          #  installation
154          image: aswf/ci-openvdb:2023-clang15.0
155        env:
```

Native GPU GitHub Actions Runners

- Windows / Linux T4 GPU runners
  - 4 core / 16 GB RAM / 176GB disk
  - 16GB VRAM
  - runs-on: ubuntu-20.04-gpu-t4-4c-16g-176h
  - runs-on: windows-ds-2019-gpu-t4-4c-16g-176h
  - for pay, so typically to run GPU test suite
- Runner provides NVIDIA driver 535.54.03
- aswf-docker container provides OpenGL dev and runtime environment
  - but no X server: see OpenColorIO for how to use EGL to get OpenGL context
  - would like to research how to provide GLX support

This actually works!

Candidate Projects for GPU Runners

- OpenVDB needs GPU for nanoVDB (?)
- OpenColorIO could transition from Amazon CodeBuild (which requires custom setup per project)
- OpenShadingLanguage
- Others? We want to help
- Running out of monthly funds for GPU accelerated testing would be a good problem to have
- T4 GPUs sufficient?

Static Analysis, Dynamic Analysis and Testing

- ASWF projects use a variety of tools for:
  - Code coverage
  - Static analysis
  - Dynamic analysis
  - Test harness
- Tools required to meet OpenSSF badging requirements
- Projects can help each other
  - OpenEXR has done great work there
- WG-CI Wiki a central place to document (TODO)
- External tools can be set up with an ASWF org and org-level secrets / credentials (SonarCloud)

LLVM

pytest

Sample ASFW Projects on SonarCloud

Packaging and Distribution

- Python and PyPI
  - Wheels packaging expertise in several projects, ask in #wg-ci
- Paid Docker Hub account for Docker containers
  - No throttling on downloads of aswf-docker containers
  - Available to all projects (GitHub org level secrets)
- JFrog Artifactory
  - https://linuxfoundation.jfrog.io/artifactory/aswf-conan/
  - Conan packages from aswf-docker, supports other formats
  - Working towards making Conan packages usable outside of aswf-docker for project dependencies

The aswf-docker Project

https://github.com/AcademySoftwareFoundation/aswf-docker
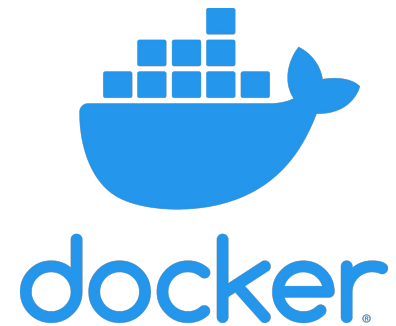
- Created by Aloys Baillet
- Yearly implementation of the VFX Reference Platform
- Hierarchy of layered containers:
  - aswf/ci-base has tools and most prerequisites
  - aswf/ci-openexr is used to build OpenEXR
  - aswf/ci-vfxall has everything pre-built
- Used by most ASWF projects to build in a controlled environment with all required dependencies
- VFX2024 based on Rocky Linux 8, leverages RedHat Developer Toolset 11 for newer dev tools (gcc 11.2.1)
- Clang also included (16 and 17 in 2024)
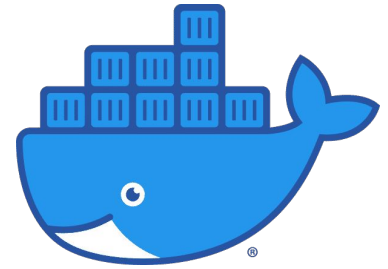
The aswf-docker Project (cont.)

- Includes CUDA, OpenGL
  - Can be used to run GPU accelerated test suites
  - Can be used to run entire applications
- Can be used to build "difficult to build" open source projects on CentOS 7 by providing updated dependencies and tools
- Pushed to paid for, unthrottled account:
  - https://hub.docker.com/u/aswf
  - but 4+ GB vfx-all container can still take a while to pull...

The aswf-docker project: work in progress

- Convert more dependencies and packages to Conan
  - Create Conan packages from build results
  - Push to JFrog Artifactory repository
  - More granularity for clients consuming packages
  - More applicable to Windows and macOS
  - `aswf-docker/packages/conan/recipes/`

- VFX Platform 2024/2025 support
  - Qt6 was a big step for 2024
  - Hoping to release a 2024.1 with updated versions
  - Main transition in 2025 is oneTBB / openMKL from Intel oneAPI

The aswf-docker project: what's next

- Support for new ASWF projects: ORI, OIIO, OpenFX...
  - OIIO hopefully for 2024.1
- Complete transition to Conan for every dependency and package
- Support use of Conan packages without having to build inside container (lighter weight dependencies)
- WINDOWS! (strong feedback from FMX presentation)
  - Leverage CMake / Conan integration to provider simpler Windows builds?
  - Many ASWF projects already have public Conan recipes

Documentation

- ASWF Confluence instance, available to all projects and WGs that want a Wiki
  - wg-ci just started our own
- Paid ReadTheDocs instance (gets rid of small ad / supports the project)
  - current effort to standardize on docs.<project>.org

Challenges

- Leadership
  - How to bring in new participants
  - Where to find external guest speakers
  - No leadership transition planning
- Could use more sustained resource commitment
  - aswf-docker project is "load bearing" for ASWF yet doesn't have any dedicated resources
  - So many opportunities to gather and reuse infrastructure work done in individual projects
  - Not exactly a unique problem...
- Breaks the model
  - Not a project but has deliverables
  - Doesn't have a clear "end" like a WG is supposed to
  - But doesn't really fit being a full project

For Your Consideration

- CI WG provides a unique forum for exchange of development infrastructure information
- Doesn't really compete with other projects for resources (except this annual review...)
- Provides value to ASWF projects in general
- So... renew us?