Finanziato dall'Unione europea
NextGenerationEU

Ministero dell'Università e della Ricerca

Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA

ICSC
Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

IRCSS Sant'Orsola Computational Genomic Platform

Jacopo Gasparetto

Workshop sul Calcolo nell'INFN – Palau, 20 - 24 Maggio 2024

Contesto

Porting della piattaforma di genomica computazione dell'IRCSS Policlinico Sant'Orsola di Bologna su EPIC Cloud

Accordo di ricerca tra INFN e Sant'Orsola per l'evoluzione della piattaforma di calcolo e superamento delle attuali limitazioni

Secondo il GDPR, dati clinici e medici sono dati personali e i dati **genomici** sono per definizioni impossibili da anonimizzare

Sviluppo della piattaforma genomica sulla cloud certificata ISO/IEC 27001, 27017, 27018 EPIC Cloud

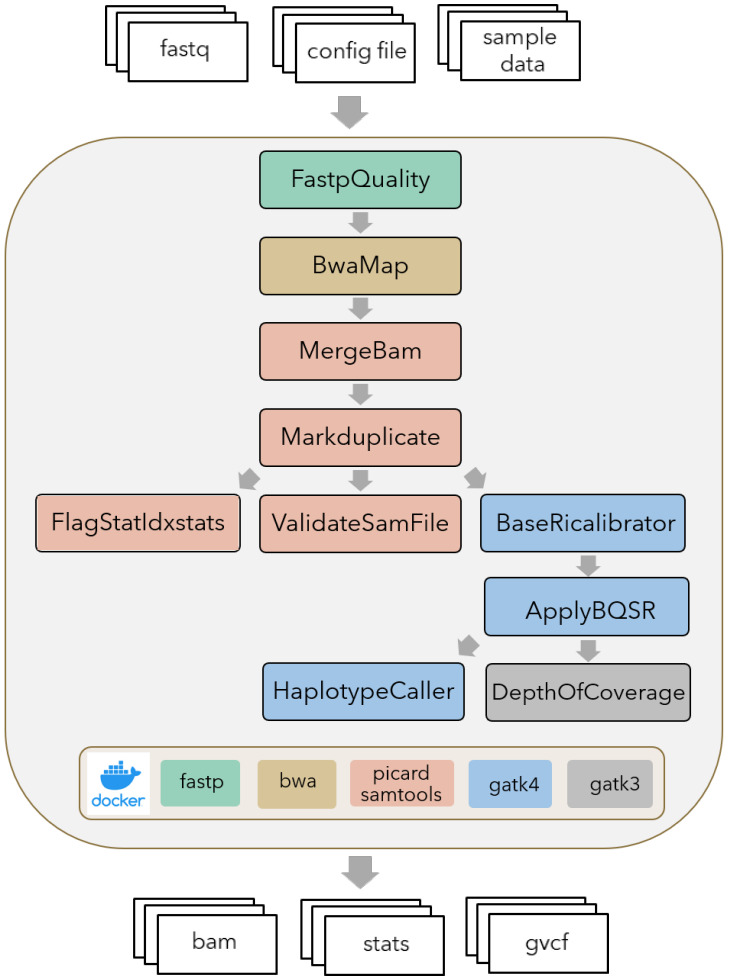
Collaborazione INFN – Sant'Orsola

- Fase 1: tenant OpenStack «Baseline»
  - IaaS gestita da INFN
  - Software installato e gestito da Sant'Orsola
  - Assenza di dati personali
- Fase 2: tenant OpenStack «Baseline» e «Integration»
  - Iaas gestita da INFN
  - Baseline gestito da Sant'Orsola con presenza di dati personali
  - Integration gestito congiuntamente da Sant'Orsola e INFN in assenza di dati personali
- Fase 3: integrazione delle nuove funzionalità sviluppate in Integration verso Baseline

Workflow di benchmark

Workflow per il rivelamento di varianti della linea germinale su dati di sequenziamento dell'intero genoma

- File types: .bam, .fastq, .gvcf
- Software
  - Burrow-Wheeler Aligner
  - Samtools
  - Genome Analysis Toolkit
- ~100GB di input per campione
- ~100GB di output per campione
  - BAM file (Binary Alignment Map format)
  - Quality metrics
  - GVCF file (Genomic Variant Call Format)
- ~ 0.5 TB di file temporanei (scratch)



Stato dell'arte: architettura Baseline

Architettura

- Architettura iper-convergente basata **macchine virtuali** (fino a 40 core e 160 GB RAM)
- **Snakemake** (workflow manager)
- **SLURM** (batch system)
- **Conda** (package manager)
- **Storage** montato sulle VM come **NAS**
- **Utenti Linux** creati manualmente sulle VM

Limitazioni

- Bassa **scalabilità**
- Bassa **affidabilità** e **disponibilità**
- Difficoltà nella gestione della **sicurezza**

Nuova architettura: Integration

- IaaS **OpenStack** (EPIC Cloud)
- Approccio a micro-servizi (**container**)
- Cluster **Kubernetes**/RKE2 per l'orchestrazione
- **Nextflow** (workflow manager)
- Monitoring tramite **Prometheus**/**Grafana**
- Pipeline di Continuous Integration/Continuous Delivery **(CI/CD)** per il build automatizzato delle immagini
- Tool di automazione (**Puppet**, **Ansible**, ecc.) per la gestione dell'infrastruttura

Cluster Kubernetes – RKE2

kubernetes RKE2

Feature

- Ambiente distribuito
- Alta Affidabilità (HA) e Resilienza
- Elasticità
- Ampia community
- Continuamente mantenuto e supportato
- Focus sulla sicurezza (RKE2)
- Hardening presets (RKE2)
- CVEs e interventi pubblicati con regolarità

Cluster Demo

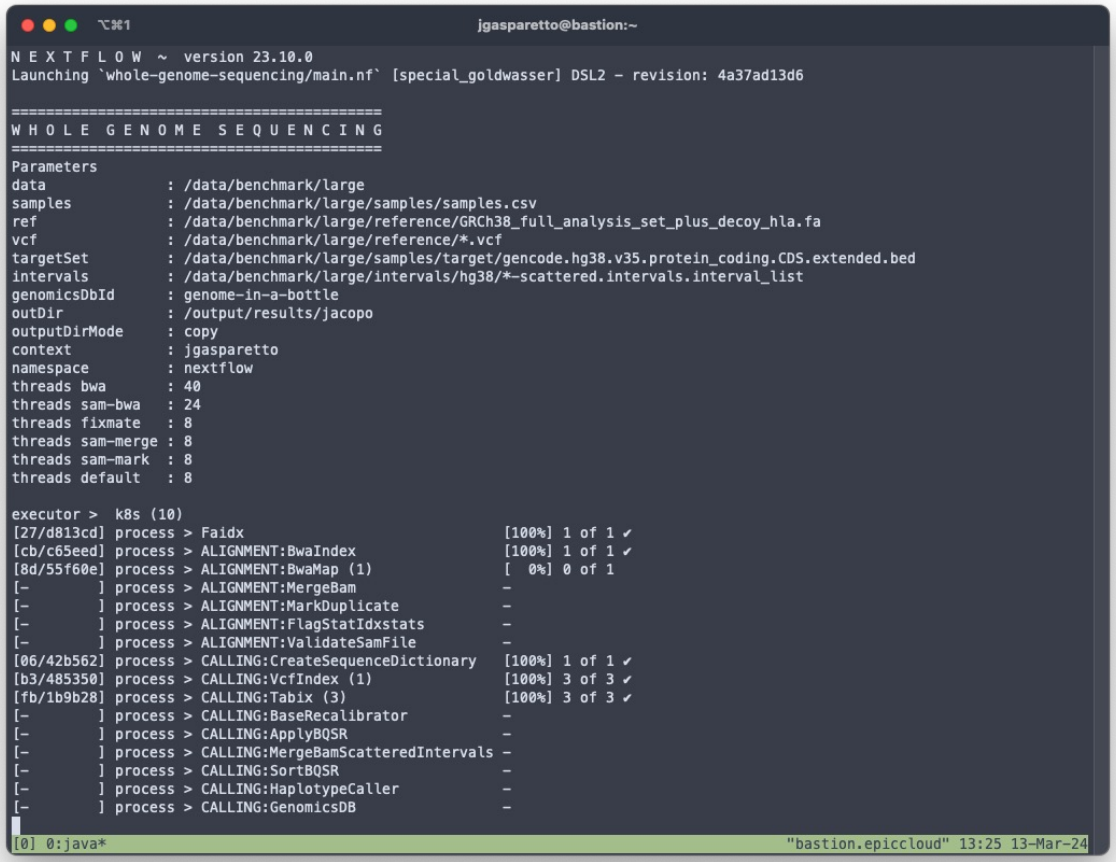
- 3 Master Node (4 CPU, 8GB RAM)
- 3 Worker Node (8 CPU, 16GB RAM)
- 1 Worker Node (40CPU, 80GB RAM)
- 3 Persistent Volume Claims (PVC) (1 TB each)
- 1 Bastione (8 CPU, 16GB RAM)

Nextflow e Kubernetes

Feature

- General purpose Workflow Manager
- Molto supportato e largamente adottato dalla comunità di bioinformatica
- Supporto nativo per i container
- Supporto nativo di K8s (anche Snakemake)
- Supporto nativo K8s Persistent Volume Claim (PVC)
- Node selector (CPU, GPU, "small", "medium", "large")

nextflow



Nextflow vs Snakemake

snakemake

- Pipeline scritte in Python
- Incentrato su Conda Environment
- Supporto limitato ai container (container-in-container)
- Cache delle immagini per K8s non supportata
- Storage and I/O vincolati a trasferimenti S3/SFTP
- K8s PVC non supportati
- Ogni pod deve scaricare/caricare centinaia di GB di dati per ogni step del workflow

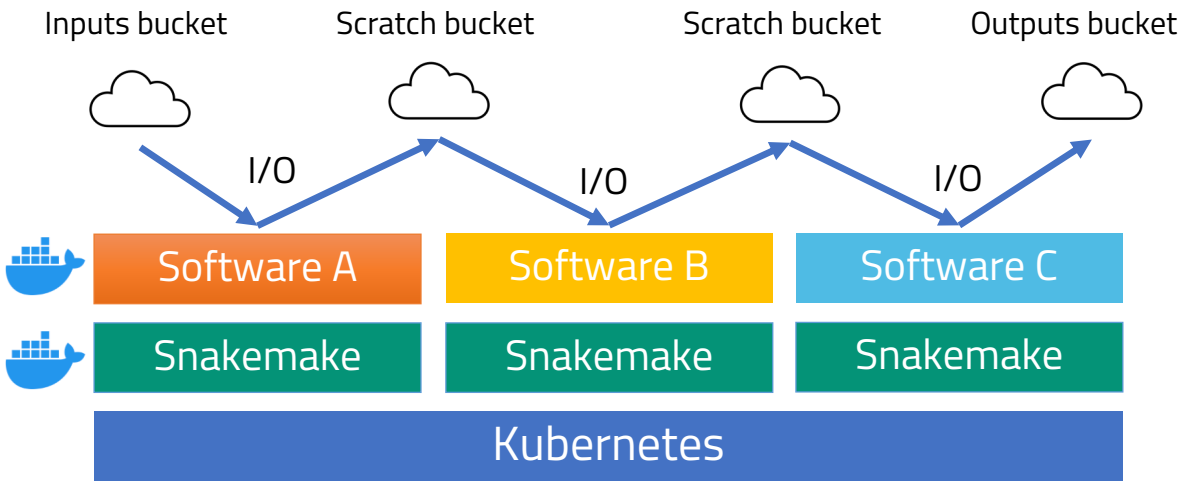
nextflow

- Pipeline scritte in Groovy
- Container pienamente supportati
- Supporto nativo per i container
- Possibilità di eseguire il workflow in locale o sul cluster senza alcuna modifica
- K8s «cache friendly»
- K8s PVC supportati
- Dati di I/O immediatamente visibili ai pod/step successivi senza la necessità di trasferimenti aggiuntivi

Nextflow vs Snakemake

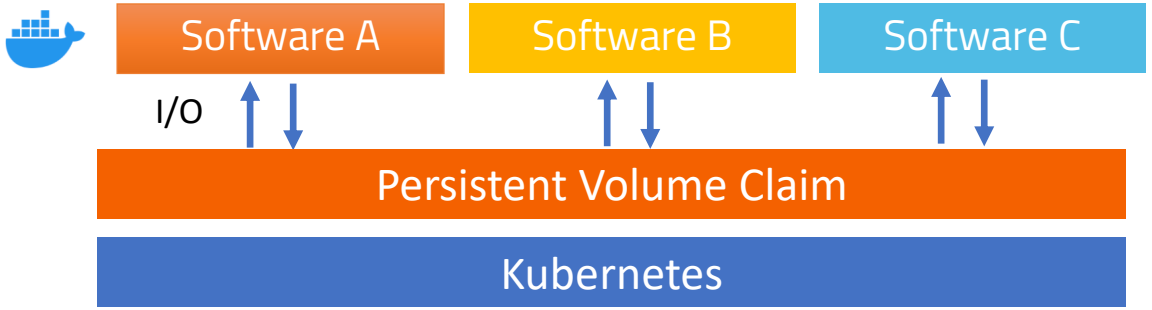
snakemake

- File scaricati/caricati verso storage S3-like
- Tutti i pod creati a partire dalla stessa immagine contenente Snakemake
- Il software finale viene eseguito come container-in-a-container (no cache da parte di K8s)



nextflow

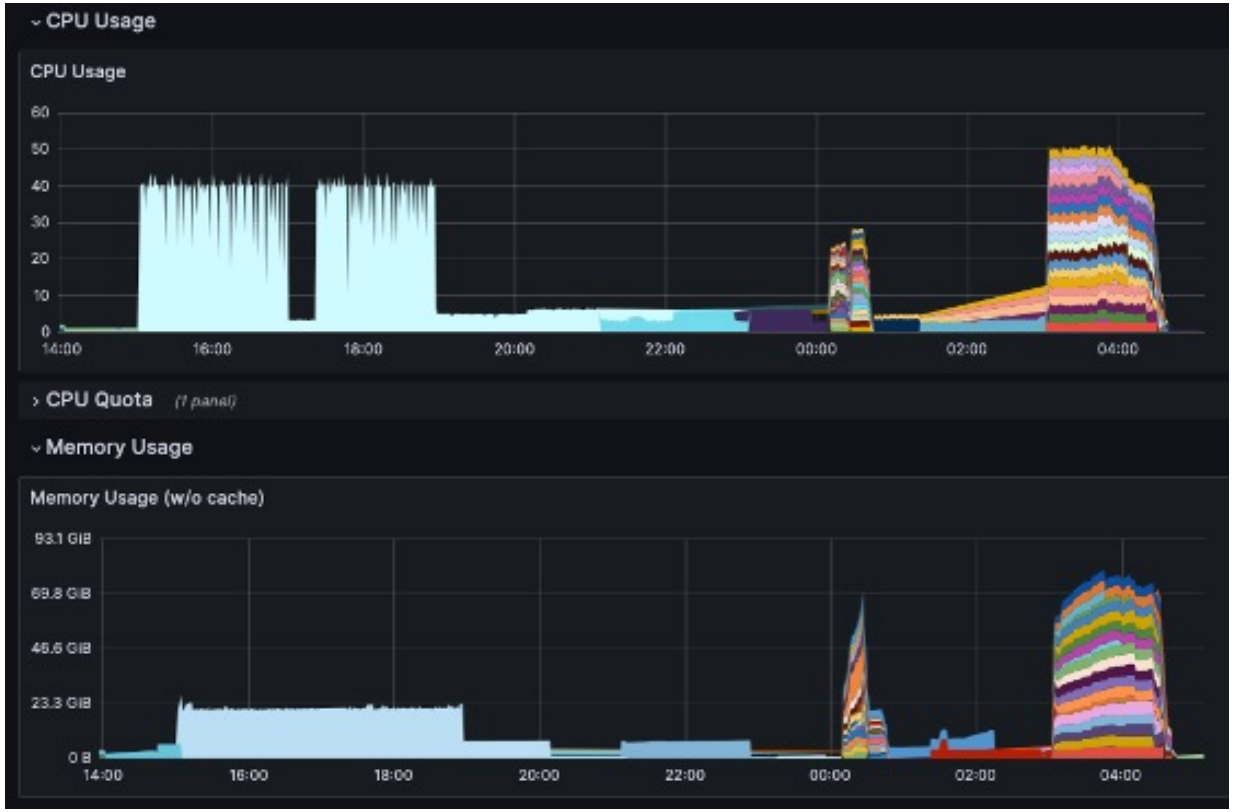
- PVC configurati come volumi NFS montati su tutti i worker node e visibili ai pod
- File di I/O direttamente disponibili ai pod senza trasferimenti
- Pod creati a partire da immagini «native» e altamente ottimizzate (cache K8s)



Monitoring (Prometheus & Grafana)

Feature

- Analisi approfondita dell'intero workflow e dei singoli step/pod
- Debugging accurato per ogni singolo processo
- Ottimizzazione dell'allocazione delle risorse in base ai requisiti dei singoli step (node selector)
- Colli di bottiglia facilmente individuabili
- Risultati e benchmark consistenti e comparabili



Conclusioni

- I primi test della migrazione da un'architettura «tradizionale» verso un'architettura cloud hanno dato risultati promettenti in termini di performance e di esperienza utente
- Overhead iniziale nella realizzazione dell'infrastruttura
- Architettura complessa che offre però alta affidabilità, scalabilità ed elasticità
- L'adozione dell'approccio a micro-servizi offre portabilità, permettendo di eseguire lo stesso workflow sul proprio laptop in fase di sviluppo e successivamente su grande cluster per la computazione vera e propria
- «User friendly» per gli operatori e ricercatori che sottomettono i job
- Sistema di monitoraggio avanzato per lo sviluppo, debugging e ottimizzazione delle performance e delle risorse
- Possibilità di eseguire scansione automatiche di vulnerabilità delle immagini «tailormade» per una migliore sicurezza del software impiegato

Futuro

- Integrazione di un Identity Provider (IdP) come Keycloak, FreeIPA o Indigo IAM per la gestione centralizzata degli utenti e Single Sign-On (SSO)
- Integrazione del software Galaxy per la sottomissione dei jobs attraverso dashboard web, oltre che da riga di comando
- Integrazione con soluzioni di object storage per lo stoccaggio dei risultati di output
- Stress test
- Migrazione dell'architettura definitiva sul tenant Baseline

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

Ringraziamenti

Alessandro Costantini, Andrea Chierici, Daniele Cesini, Francesco Sinisi, Diego Michelotto, Giusy Sergi, Letizia Magenta, Lorenzo Chiarelli, Luca dell'Agnello, Stefano Zani, Tania Giangregorio, Federica Isidori, Emanuela Iovino, Tommaso Pippucci, Vincenzo Ciaschini, Barbara Martelli

jacopo.gasparetto@cnaf.infn.it


ICSC
Centro Nazionale di Ricerca in HPC,
Big Data and Quantum Computing
Supercomputing
shaping the future