HOKKAIDO UNIVERSITY

DATAIA-JST International Symposium
on Data Science and AI
Paris, France

Optimal Resource Selection in Application-Centric Overlay Cloud Utilizing Inter-Cloud

July 12th, 2018

JST CREST

大学共同利用機関法人 情報・システム研究機構
国立情報学研究所
National Institute of Informatics

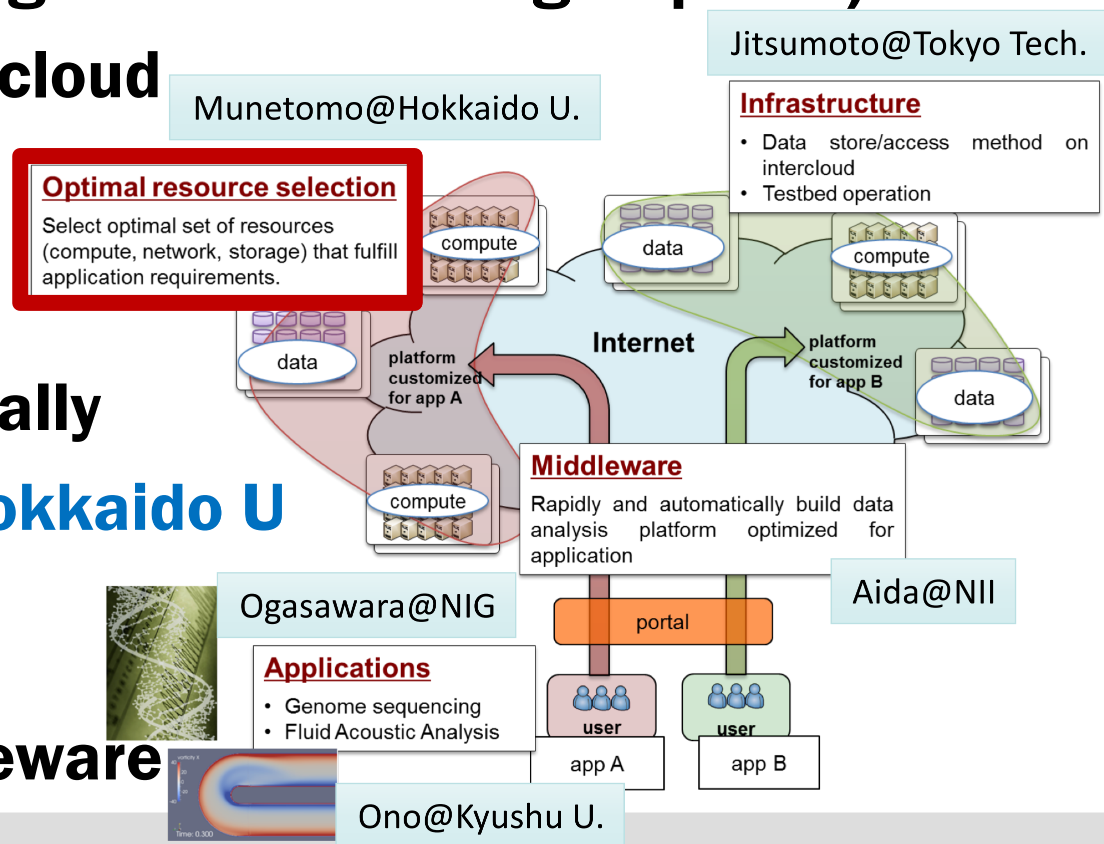
Masaharu Munetomo

Information Initiative Center,
Hokkaido University

(JST CREST Prof. Aida group@NII)

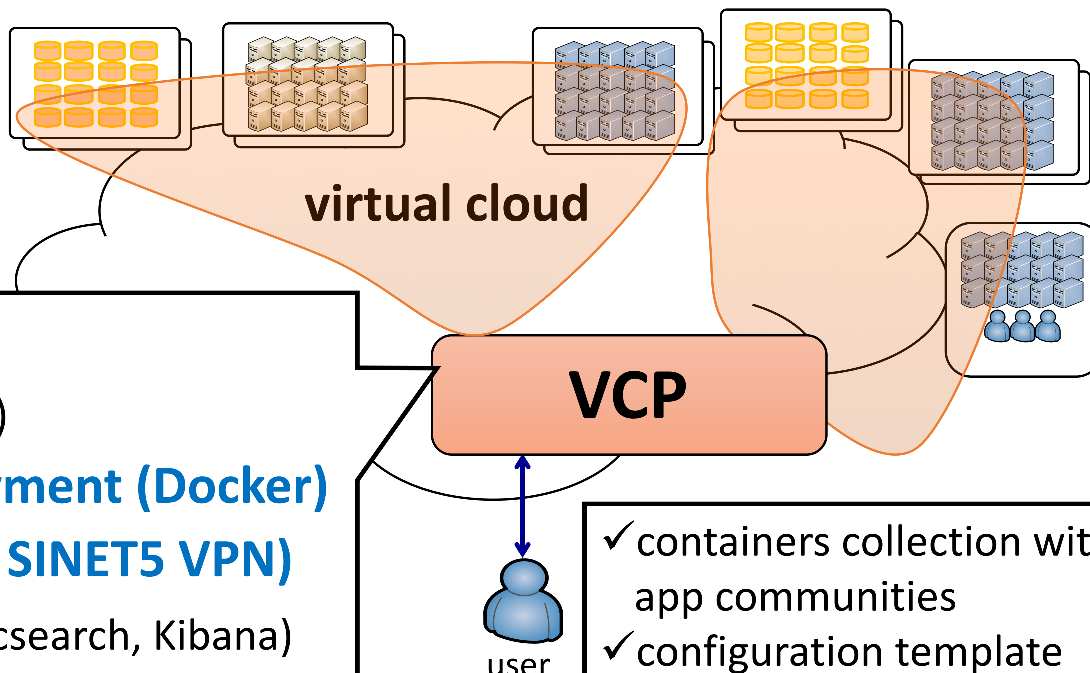
Project Overview

- We develop optimal resource selection scheme based for **"Application-Centric Overlay Cloud Utilizing Inter-Cloud"** (JST CREST Bigdata: Prof. Aida group@NII)
- Development of intercloud middleware to build bigdata analytic platform in the intercloud automatically
- **Munetomo group@Hokkaido U** develops an optimal resource selection scheme of the middleware



Virtual Cloud Provider (VCP) Middleware (Prof. Aida group)

- **Automatic and quick creation of virtual clouds (= data analysis platforms) configured by resources over multiple cloud platforms**



1. resource selection
2. VM/BM control (Terraform)
3. **base container deployment (Docker)**
4. **network setup (IPSec, SINET5 VPN)**
5. monitoring (Fluentd, Elasticsearch, Kibana)
6. dynamic resource reconfiguration

✓ containers collection with app communities
✓ configuration template (Jupyter Notebook)

Demo: Building Galaxy Virtual Cloud with VCP

jupyter SC17_Galaxy_infra Last Checkpoint: 14 minutes ago (autosaved)

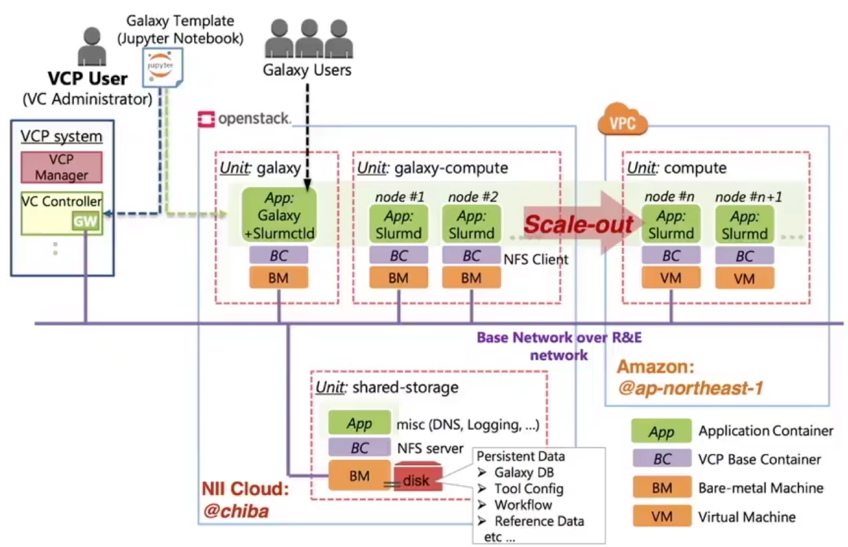
File Edit View Insert Cell Kernel Navigate Help

Trusted Python 3

Contents [-]

1 Construct a virtual cloud for Galaxy + SLURM cluster

- NIIクラウドでGalaxyクラスタ環境の VC (Virtual Cloud) を作成する。
- Galaxyサーバと、そのジョブ実行先となる計算ノード群をSLURMクラスタとして構成する。



Optimal resource selection (Munetomo group)

- **We discuss optimal resource selection in scientific bigdata processing in the intercloud environment.**
  => Integration to VCP middleware

1. **Multi-objective optimization to minimize deployment cost, make span, and maximize availability, etc. to deploy scientific workflows in the intercloud.**
2. **Development of cloud service broker satisfying a number of constraints related to the target workflow and cloud infrastructure (services).**
3. **Multi-objective optimization of parameters configuration in bigdata processing (Hadoop) and scientific workflows.**

Difficulty in optimal cloud service selection

- **It is difficult to select proper services and their options for instances of workflow components.**
  - **Instance types (cores, memory, storages)**
  - **Location (region)**
  - **Providers**
  - **Purchase option (spot, reserved)**
- **The users are only interested in overall cost & performance.**

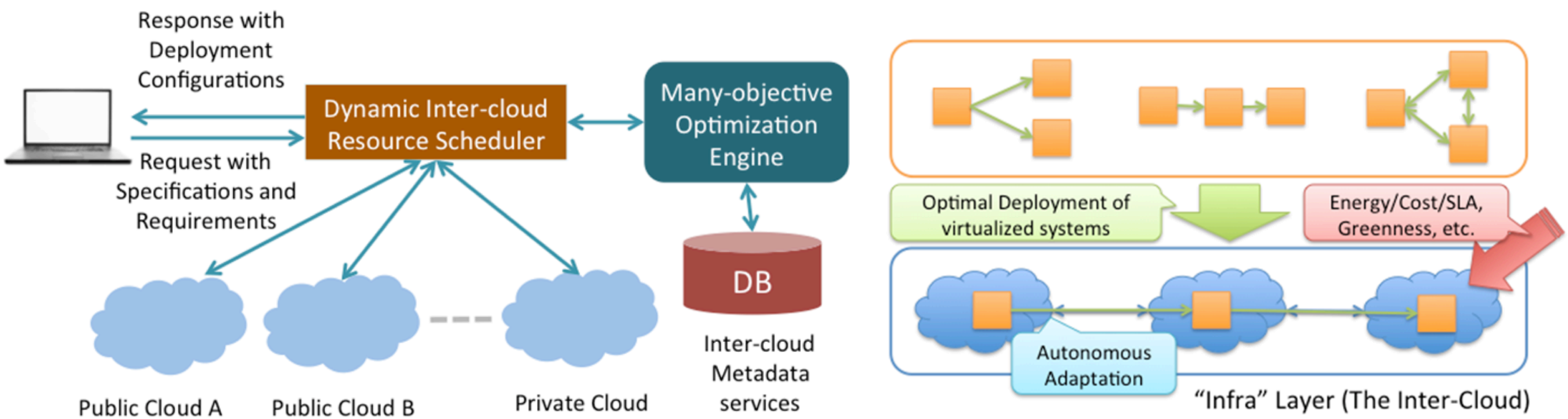
Amazon EC2 instance types

Instance Family	Current Generation Instance Types
General purpose	t2.nano | t2.micro | t2.small | t2.medium | t2.large | t2.xlarge | t2.2xlarge | m4.large | m4.xlarge | m4.2xlarge | m4.4xlarge | m4.10xlarge | m4.16xlarge | m5.large | m5.xlarge | m5.2xlarge | m5.4xlarge | m5.12xlarge | m5.24xlarge
Compute optimized	c4.large | c4.xlarge | c4.2xlarge | c4.4xlarge | c4.8xlarge | c5.large | c5.xlarge | c5.2xlarge | c5.4xlarge | c5.9xlarge | c5.18xlarge | c5d.xlarge | c5d.2xlarge | c5d.4xlarge | c5d.9xlarge | c5d.18xlarge
Memory optimized	r4.large | r4.xlarge | r4.2xlarge | r4.4xlarge | r4.8xlarge | r4.16xlarge | x1.16xlarge | x1.32xlarge | x1e.xlarge | x1e.2xlarge | x1e.4xlarge | x1e.8xlarge | x1e.16xlarge | x1e.32xlarge
Storage optimized	d2.xlarge | d2.2xlarge | d2.4xlarge | d2.8xlarge | h1.2xlarge | h1.4xlarge | h1.8xlarge | h1.16xlarge | i3.large | i3.xlarge | i3.2xlarge | i3.4xlarge | i3.8xlarge | i3.16xlarge | i3.metal
Accelerated computing	f1.2xlarge | f1.16xlarge | g3.4xlarge | g3.8xlarge | g3.16xlarge | p2.xlarge | p2.8xlarge | p2.16xlarge | p3.2xlarge | p3.8xlarge | p3.16xlarge

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/instance-types.html

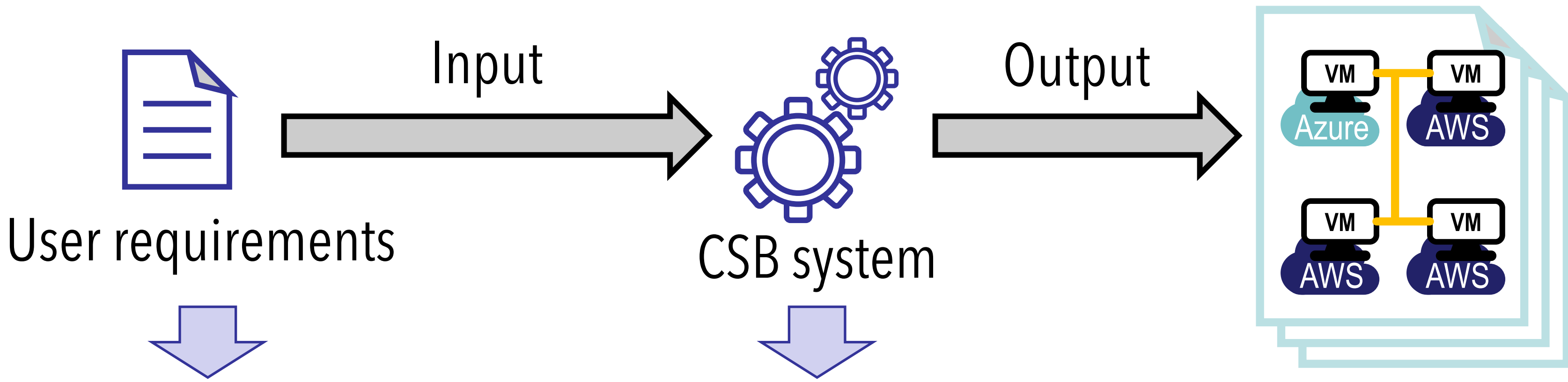
Optimal resource selection in the intercloud

- **Multi-objective optimization problem: we want to minimize cost, execution time, and maximizes availability etc. simultaneously.**
- **Scheduler should be developed to select optimal resources such as virtual machine instances and their configurations from a huge variety of services options.**



Cloud Service Broker (CSB) based on Equivalent Transformation (ET) with Predicate Logic Spec. (PLS)

- **We develop a cloud service broker (CSB) by ET to find feasible infrastructure deployment patterns based on user's requirements.**



Predicate
Logic-defined
Specification

Replacement of definite clause sets by **ET** rules

Feasible Infrastructure deployment patterns

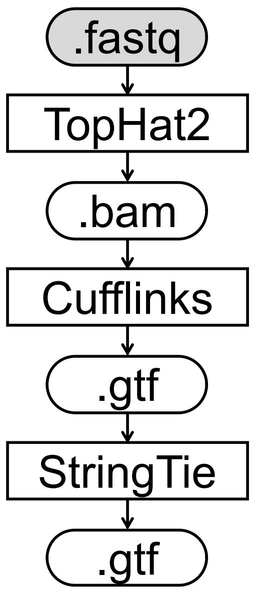
Predicate Logic-defined Specification (PLS)

- **We describe system requirements with predicate logic formula.**

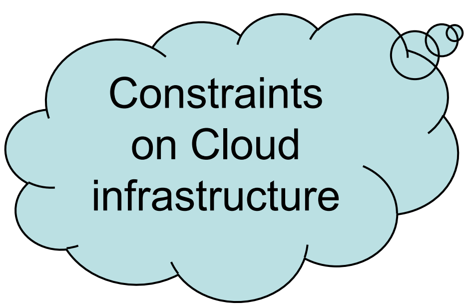
$$\forall_{\bar{v}}\{E_1 \wedge E_2 \wedge \cdots E_n\}$$

Variables

Atomic formula (predicates)



Workflow Spec.



$$\forall_{\{*structure\}}\{Envrionment(*structure, [[step1, \text{“}tophat2\text{”}, [2.0]] \; [step2, \text{“}cufflinks\text{”}, [2.0]] \; [step3, \text{“}stringtie\text{”}, [2.0]]]) \wedge$$

$$Cost(*structure, 10.0) \wedge$$

$$Location(*structure, [\text{“}Virginia\text{”}, \text{“}Tokyo\text{”}]) \wedge$$

$$Policy(*structure, \text{“}public\text{”}, \text{“}single\text{”})\}$$

**structure* Cloud service/type (e.g. instance type)
Environment Genomic analytics workflow to deploy
Cost Maximum price payable
Location Deployment location or region, zone
Policy Selction policy of cloud service providers

Equivalent Transformation (ET)

- **ET program: Set of rewriting rules that replaces clause sets, while preserving the declarative meaning of original clause. (Akama, 2006)**

S_1 = {(answer *x *y) <-- (append *x *y (1 2))}

Replacement

(append *a *b *c)
  ==> {(= *a ()), (= *b *c)};
  ==> {(= *a (*d|*e)), (= *c (*d|*f))}, (append *e *b *f).

Rewriting rule

(*a|*b) = (1 2 3)
*a = 1, *b = (2 3)

Preserving the declarative meaning

S_2 = {(answer () (1 2)) <--
    (answer (1|*z) *y) <-- (append *z *y (2))}

Cloud Resource Selection Based On PLS Method

Step 1: Construct a PLS reflecting system requirements

A PLS has the following form:

$$\forall_{\bar{v}}\{E_1 \wedge E_2 \wedge \cdots E_n\},$$

where E is an atomic formula (atom), $\bar{v}$ is a set of variables appearing in the formula.

Step 2: Generate a definite clause cl from the PLS

A definite clause cl has the following form:

$$ans(v_1, v_2, \cdots v_i) \leftarrow E_1, E_2, \cdots E_n,$$

where the term of an ans atom is all variables in $\bar{v}$.

Step 3: Transform clause set $\mathbb{D}$

Clause set $\mathbb{D}$ is transformed while preserving the declarative meaning of the initial state in accordance with the following procedure, where the initial state is $\{cl\}$.

Cloud Service Broker (CSB) based on ET and GA

- **We combine ET and GA to find optimal solutions that also satisfy constraints (requirments).**

Input: System/App requirement

- Genomic workflow
- Maximum price payable
- Deployment location of VMs
- Service provider type

CSB system

Output: Pattern set

Infrastructure pattern 1

$ans([\text{"aws"}, \text{"r4.16xlarge"}, \text{"Virginia"}, [s1, \text{"tophat2"}, [2.0], 892.49]],$
$[\text{"aws"}, \text{"m4.xlarge"}, \text{"Virginia"}, [s2, \text{"cufflinks"}, [2.0], 1.0]],$
$[\text{"aws"}, \text{"m4.xlarge"}, \text{"Virginia"}, [s3, \text{"stringtie"}, [2.0], 1.0]]) \leftarrow$

Describe

Compute

Describe

Logical formula

$\forall_{\overline{v}}\{E_1 \wedge E_2 \wedge \cdots E_n\}$

Transformation of clause set

Set of unit clauses

$$\{(H\theta \leftarrow) \mid (H = ans(v_1, v_2 \cdots v_m))\& (B = \{E_1, E_2, \cdots E_n\})\& (\theta \in \mathbb{S})\&(B\theta \subseteq \mathbb{G})\& (H\theta \in \mathcal{G})\}$$

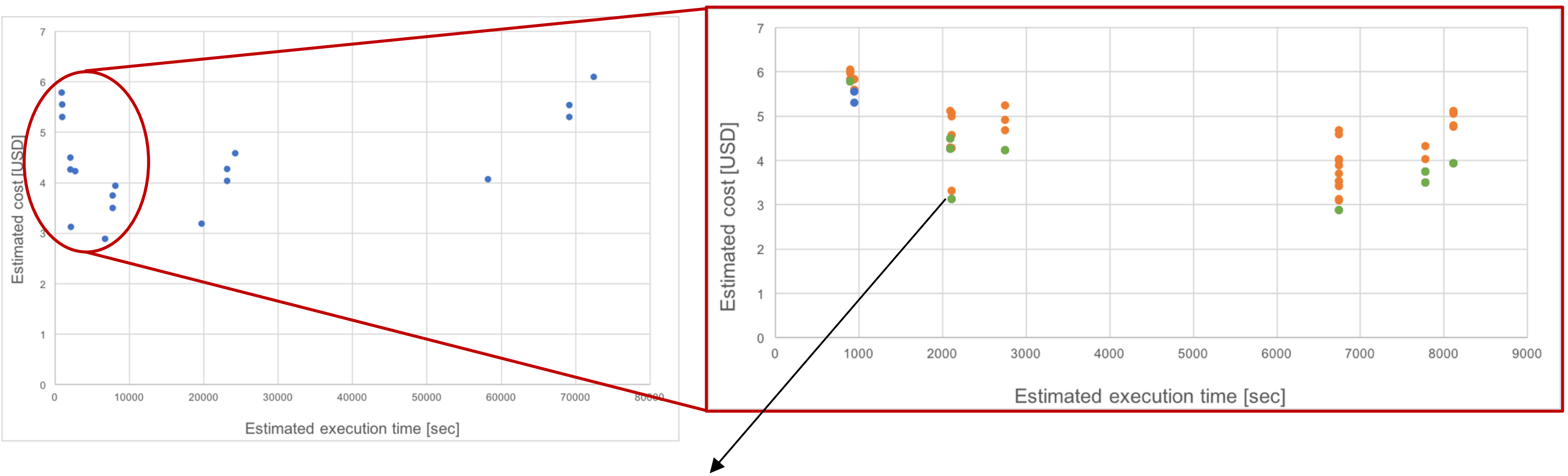
Clause transformation by ET while evaluating unit clause by GA

For optimality

For satisfiability

Results

- **60 definite clauses and 19 unit clauses are obtained from the initial state, and the unit clauses are evaluated by GA. Green points represent optimal solutions.**



$$ans(\,[\text{“}aws\text{”}, \text{“}r4.16xlarge\text{”}, \text{“}Virginia\text{”}, [step1, \text{“}tophat2\text{”}, [2.0], 892.49]],\\ [\text{“}aws\text{”}, \text{“}m4.xlarge\text{”}, \text{“}Virginia\text{”}, [step2, \text{“}cufflinks\text{”}, [2.0], 1.0]],\\ [\text{“}aws\text{”}, \text{“}m4.xlarge\text{”}, \text{“}Virginia\text{”}, [step3, \text{“}stringtie\text{”}, [2.0], 1.0]]\,) \leftarrow$$

Many-objective optimization engine with NSGA-III

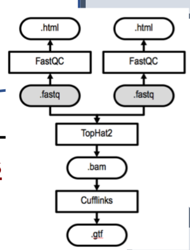
- **We develop a many-objective optimization engine for scientific workflows considering tradeoff among cost, make span, and availability by employing NSGA-III.**

Input from Constraint Solver

Set of resource configurations satisfying constraints *(feasible configs.)*

Historical Data

- Site uptime/downtime
- Hourly usage ratio
- N/w conditions



Input from Users

- Genomic workflow
- Objectives (cost, makespan, availability)
- Reference values *(optional)*

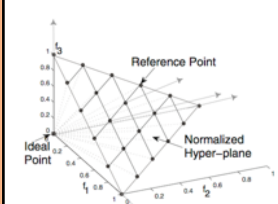
Modified NSGA-III Optimizer

(Search feasible set for optimal configurations)

Feasible initial set of resource configs

Nondominated sort

Reference values association

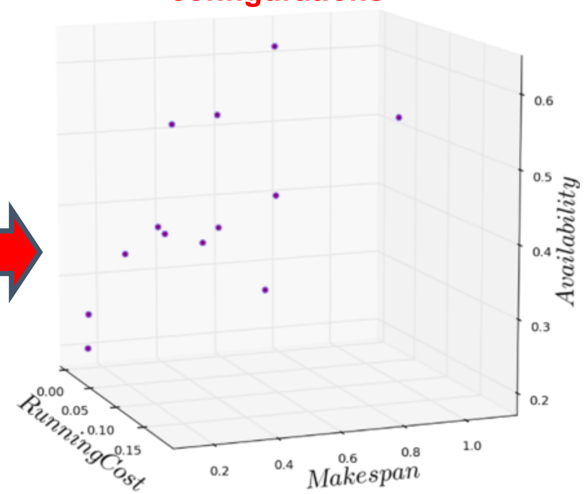


Niching

Discard worst configurations

Insert new feasible resource configurations

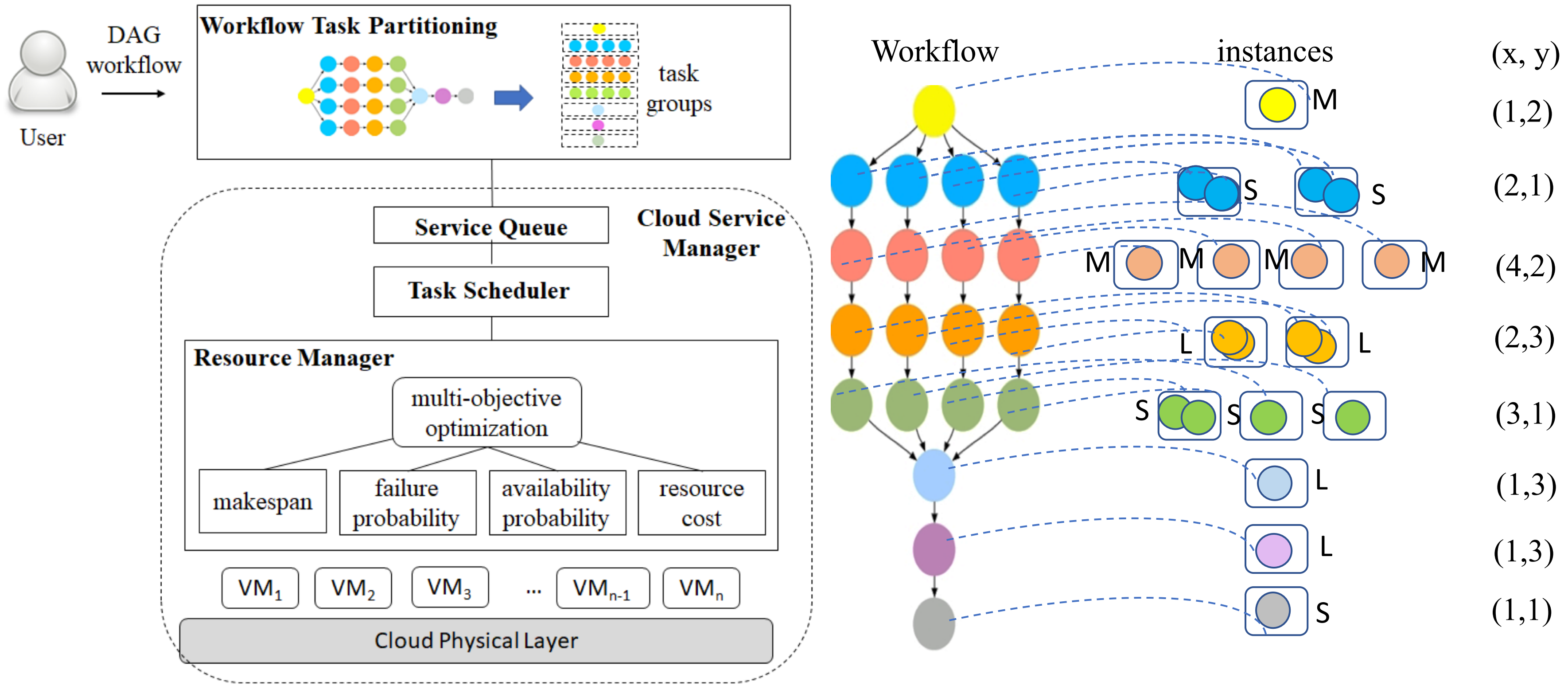
Distribution of optimal tradeoff configurations



Note: *Optimal tradeoff solutions are evenly distributed*

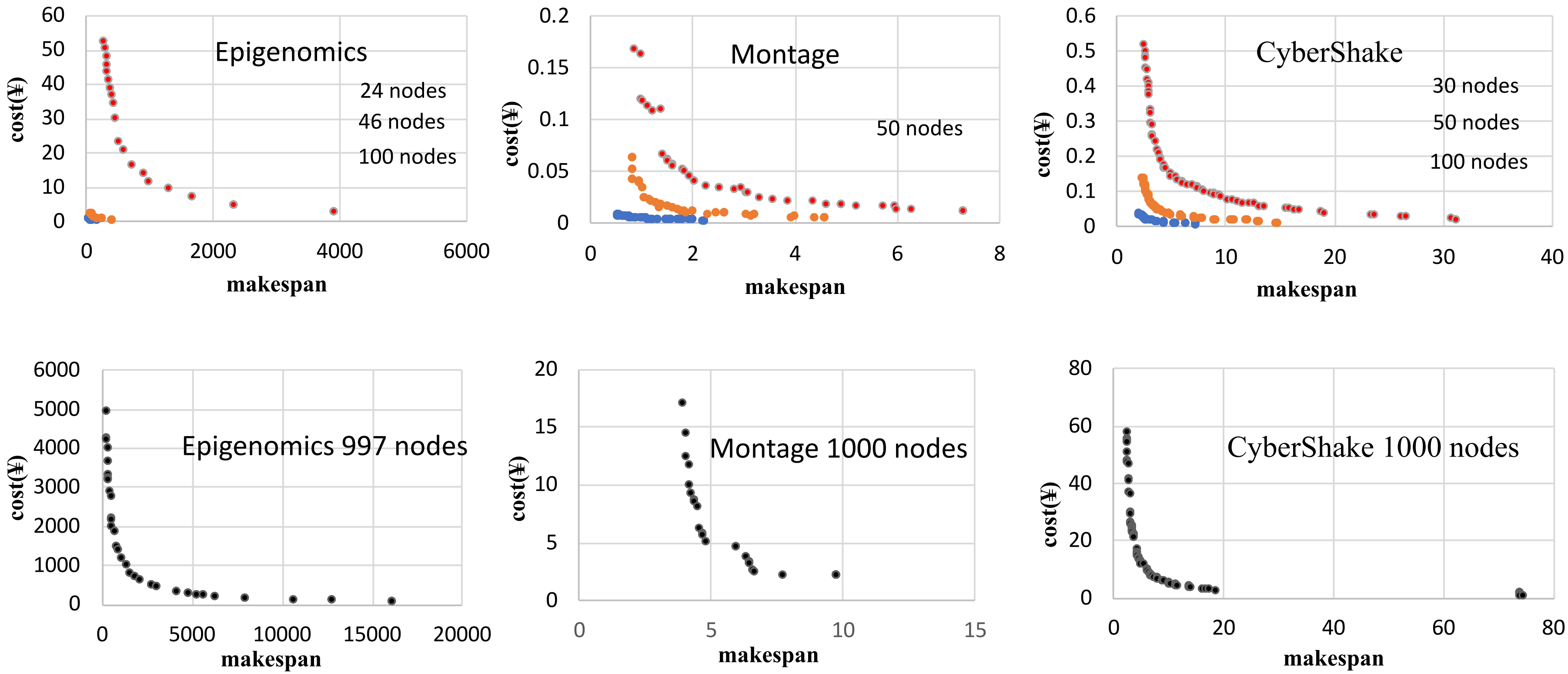
Multi-objective scientific workflow optimization

- **We employ NSGA-II/III to solve multi-objective optimization (cost, makespan, availability) of level-wise scientific workflows.**



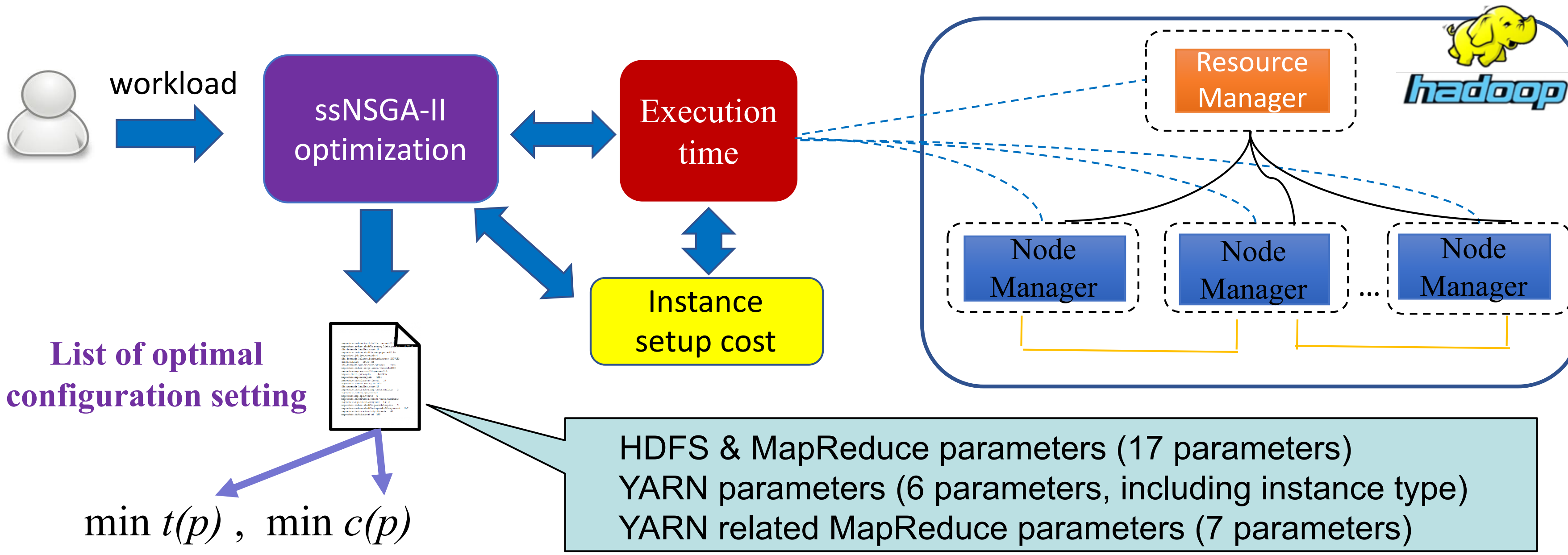
Experimental results (Scientific workflows)

- **We perform experiments on Epigenomics, Montage, and CyberShake workflow benchmarks, and obtain Pareto optimal solutions.**



Parameter optimization for Hadoop bigdata processing

- **Hadoop configuration parameter optimization using steady-state NSGA-II.**



$p = [p_1, p_2, \ldots, p_m]$, configuration parameter list and instance type

$t(p)$ = execution time of MR job per second

$c(p)$ = machine instance usage cost in yen

Experiment results (Hadoop)

■ Conditions

Population Size, P= 30

Number of Evaluations, E = 150

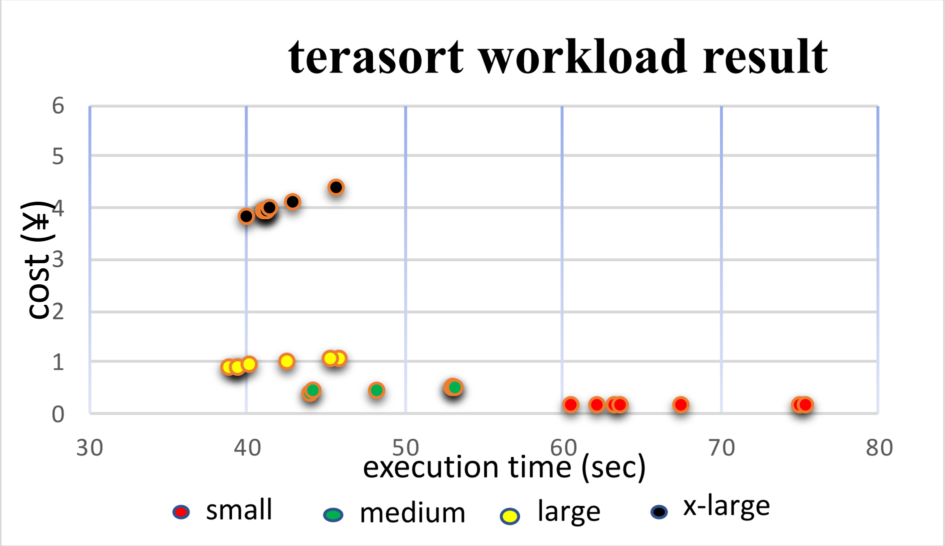
Number of Objectives, O = 2

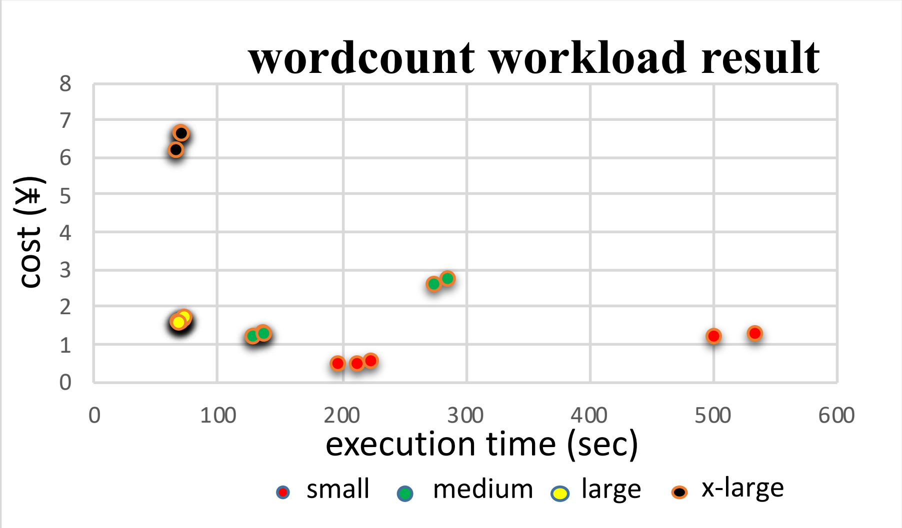
Mutation Probability, MP = 0.1

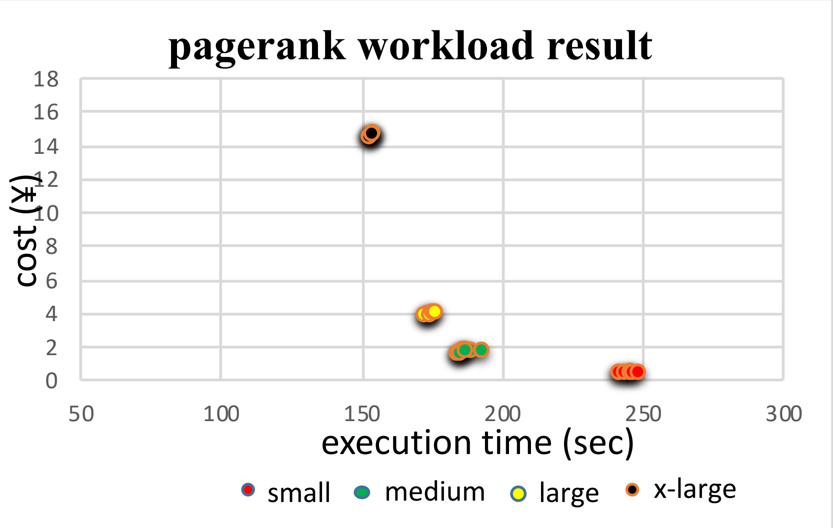
Crossover Probability, CP = 1

Hokkaido univ. academic cloud

(for cost & exec. time conditions)







Current achievements and Future works

- **We employ multi-objective genetic algorithms and equivalent transformation to solve multi-objective resource optimization problems with complex constraints in the inter-cloud environment.**
  - **Scientific (genomic) bigdata workflows**
  - **Hadoop parameters configuration**
- **We are integrating our methodologies into the Application-Centric Overlay Cloud (VCP).**
  - **As a Cloud Broker/Scheduler to satisfy various needs in wide-spectrum of bigdata applications.**

Publications

Katsunori Miura, Courtney Powell, Masaharu Munetomo: Cloud Resource Selection Based on PLS for Deploying Optimal Infrastructures for Genomic Analytics Application, ACM/IEEE SC17 Poster (2017.11.14)

Phyo Thandar Thant, Courtney Powell, Martin Schlueter, Masaharu Munetomo: Multi-Objective Level-Wise Scientific Workflow Optimization in IaaS Public Cloud Environment, Scientific Programming, Hindawi, Vol. 2017, Article ID 5342727, 17 pages (2017)

Phyo Thandar Thant, Courtney Powell, Martin Schlueter, Masaharu Munetomo: Constrained Multi-Objective Scientific Workflow Execution Optimization with NSGA-III in the Cloud, International Journal of Computer Science and Information Security, Vol. 15, No.10, pp. 92-102 (2017)

Phyo Thandar Thant, Courtney Powell, Martin Schlueter, Masaharu Munetomo: A Level-Wise Load Balanced Scientific Workflow Execution Optimization using NSGA-II, Proceedings of 17th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (IEEE/ACM CCGRID 2017) (the 2nd International Workshop on Distributed Big Data Management (DBDM 2017)), pp. 882-889 (2017)

Katsunori Miura, Tazro Ohta, Courtney Powell, Masaharu Munetomo: Intercloud Brokerages based on PLS Method for deploying Infrastructure for Big Data Analytics, Workshop of Big Data for Cloud Operation Management, 2016 IEEE International Conference on Big Data (IEEE Big Data 2016) (2016)

Courtney Powell, Masaharu Munetomo, Phyo Thunder Thant: Evaluation of Three Steady-State NSGA-III Offspring Selection Schemes for Many-Objective Optimization, Proceedings of the 2016 Joint 8th International Conference on Soft Computing and Intelligent Systems and 2016 17th International Symposium on Advanced Intelligent Systems (SCIS&ISIS 2016), pp. 166-171 (2016)

Phyo Thandar Thant, Courtney Powell, Akiyoshi Sugiki, Masaharu Munetomo: Multi-Objective Hadoop Configuration Optimization using Steady-State NSGA-II, Proceedings of the 2016 Joint 8th International Conference on Soft Computing and Intelligent Systems and 2016 17th International Symposium on Advanced Intelligent Systems (SCIS&ISIS 2016), pp.293-298 (2016)

Katsunori Miura, Masaharu Munetomo: A Predicate Logic-defined Specification Method for Systems Deployed by Intercloud Brokerages, Proceedings of the 2016 IEEE International Conference on Cloud Engineering Workshops (Intercloud 2016), pp.172-177 (2016)

Hokkaido University High-Performance Intercloud

- Supercomputer + Intercloud system to support research projects related to HPC, Bigdata, IoT, AI, etc.
- Supercomputer system: 3.96PFLOPS (Intel Xeon + Xeon Phi)
- Intercloud system: Virtual & Baremetal Servers with multiple sites (Hokkaido, Tokyo, Osaka, Kyushu) across Japan to support distributed systems development (IoT, etc.).