УДК 004.6::574.5

ИНФОРМАЦИОННАЯ ИНФРАСТРУКТУРА ДЛЯ ПОДДЕРЖКИ ИССЛЕДОВАНИЙ МИКРОБИОМА БАЙКАЛА

Черкашин Евгений Александрович
Институт динамики систем и теории управления
Сибирского отделения Российской Академии наук
email: eugeneai@icc.ru,
Шигаров Алексей Олегович
Иркутский научный центр Сибирского отделения Российской Академии наук
e-mail: shigarov@icc.ru,
664033, Иркутск, ул. Лермонтова, 134.
Христюк Василий Владимирович
Национальный исследовательский Иркутский государственный технический университет
e-mail: khr@icc.ru
664074, Иркутск, Лермонтова, 83.

Аннотация. Рассмотрена проблема построения исследовательской среды для обработки данных секвенирования нового поколения (NGS – Next Generation Sequencing). Среда включает облачное хранилище данных (DaaS) и вычислительные службы (SaaS и PaaS), а также службы визуализации и интеграции данных. Осуществляется интеграция технологий с открытым исходным кодом для поддержки MiSeq SOP (стандартная операционная процедура), которая позволяет специалистам в предметной области – биологам независимо от программистов самостоятельно обрабатывать данные. Для реализации интеграции конструируются формальные модели SOP, позволяющие автоматически порождать исходный код компонентов среды. Технология преобразования основана на принципах архитектуры, управляемой моделями (Model driven architecture), и логическом выводе структур производных моделей и модулей. Представлены текущие результаты и задачи на ближайшую перспективу.

Ключевые слова: секвенирование нового поколения, большие данные, архитектура, управляемая моделями, открытые связанные данные, планирование действий

Цитирование: Черкашин Е.А., Шигаров А., Христюк В. Информационная инфраструктура для поддержки исследований микробиома Байкала //Информационные и математические технологии в науке и управлении. 2020. № 4 (20). С. 108-123. DOI: 10.38028/ESI.2020.20.4.010

Введение. В последнее десятилетие, после изобретения методов секвенирования нового поколения (NGS) и внедрения их в практику исследования биологических систем, формируется новое направление молекулярной генетики, получившее название *метагеномики*. Его основной объект изучения выходит за рамки отдельных микроскопических культивируемых организмов и перемещается к их сообществам, *микробиомам*. Из образца извлекается ДНК, в результате чего накапливается информация, представляющая данные для всего образца микробиома. Этот метод позволяет описать значительное количество новых групп организмов

на всех таксономических уровнях. Подробный обзор существующих проблем и современных подходов к секвенированию представлен в [22].

Один из видов метагеномных исследований – анализ ампликонов, применяющийся для изучения микробиоты различных сред озера Байкал [4]. Для выполнения анализа и интерпретации результатов требуются значительные вычислительные ресурсы, а также навыки в области биоинформатики. Исследователь проектирует вычислительный процесс, комбинируя различные модули биоинформатического программного обеспечения, преобразования данных, анализа данных и визуализации. Для проведения исследований специалисты-предметники должны обладать навыками программирования сценариев командной оболочки операционной системы (Linux, Windows), запуска распределенных вычислительных сред, кластерных вычислительных системы, а также программирования на языках общего и специального назначения, например, Python и R.

Целью исследования является разработка математического и программного обеспечения процессов анализа результатов NGS. Перспективная задача авторов – разработать и внедрить методы и программное обеспечение для визуального представления вычислительного процесса анализа ампликонов, чтобы специалисты предметной области могли составлять вычислительные конвейеры, выполняющиеся в распределенных гетерогенных вычислительных ресурсах (облаках). Конечная цель – создание облачной инфраструктуры, построенной на моделях вычислительного процесса в виде набора операций, структур и функций вычислительных ресурсов, а также алгоритмов управления вычислительными ресурсами.

В [11] проблема использования облачных хранилищ и вычислений сформулирована как основная проблема NGS, так как экспоненциальный рост емкости хранилищ генетических данных происходит медленнее по сравнению с ростом объема данных, генерируемых NGS. Передача данных между хранилищами и компьютерными системами для их обработки приводит к исчерпанию емкости сети. Для обработки данных, в общем случае реконструкции всего генома, требуются терабайты оперативной памяти, а в случае использования кластерных вычислений – специальные высокопроизводительные параллельные алгоритмы. Выявлены два класса пользователей: *опытный пользователь (power user)*, который производит самостоятельно анализ генома, и *предметный исследователь* с *невысокой квалификацией пользователя в биоинформатике (casual user)*, осуществляющий обработку результатов NGS, интегрируя их с результатами других исследований.

Область исследований авторов в области информационных технологий (ИТ) связана с обработкой данных в рамках естественнонаучных исследований, которые характеризуются разнообразием задач, методов и мультидисциплинарных целей. Новые данные сравниваются со всеми данными, полученными в предыдущие годы. Увеличивается и количество научных задач. В рамках исследования предлагается построить облако PaaS и DaaS, состоящее из независимых сетевых SaaS-сервисов, адаптированных для стандартной операционной процедуры MiSeq (MiSeq SOP, стандартная процедура обработки данных), используемой в исследовании микробиома озера Байкал. PaaS позволит биологам самим анализировать и исследовать данные, DaaS позволит специалистам по биоинформатике работать с данными по запросу при разработке новых методов обработки данных, а сервис SaaS будет поддерживать отдельные операции для PaaS. К настоящему моменту уже существует программное обеспечение SaaS для обработки данных NGS, статья [11] содержит его подробный обзор.

1. Стандартная операционная процедура MiSeq. В качестве введения в проблемную область опишем пример реализации MiSeq SOP, выполненный при помощи прикладного пакета Mothur [15].

Процесс анализа данных NGS состоит из отдельных операций с генетическими данными, хранящихся в файлах. Для ознакомления с методикой авторами вручную проведена обработка лимнологических данных из исследования [18] в соответствии с процедурой, представленной на сайте Mothur. Исходные данные прочтения генов получены на секвенаторе GS FLX454.[1]

Все начинается с объединения левых и правых прочтений исходного fasta-файла, содержащего 44934 контига (смежные последовательности генов). Следующая операция – это обрезка праймеров олигонуклеотидов, т.е. частей последовательностей, идентифицирующих образцы для каждого контига. После операции обрезки, фильтрации нужно изучить сводную статистику по fasta-данным «склеенных и обрезанных» прочтений. Свод (summary) представлен в виде статистических критериев минимальной, максимальной, средней и четырехквартильной оценки. На основе полученной информации задаются значения параметров для следующих операций.

После обрезки олигонуклеотидов необходимо удалить последовательности, которые короче или длиннее некоторого среднего значения. Степень отклонения обычно определяется процентом прочтений, которые разрешено удалить. В нашем случае удалено 11052 (25%) коротких прочтений. Следующим шаг – распознавание уникальных последовательностей и подсчет их реплик (точных копий), что позволяет далее выполнять следующие шаги по одному разу для каждой уникальной последовательности, а это, в свою очередь, сокращает время вычислений. Получено 26744 уникальных последовательности, из них удаляются прочтения с характеристикой гомополимера более 8, т.к. более длинные последовательности одних и тех же пар-оснований в ряду гомополимера в природе не существуют.

Основным вычислительно сложным этапом анализа является сопоставление (выравнивание) прочтений с данными, хранящимися в базе данных референсных генов, например, SILVA [23]. Применение выравнивания к имеющимся 21170 последовательностям привело к изменению длины прочтений со средней 407 пар-оснований до 871. Пропуски символов заполняются пробелами. После обрезки «висящих» символов и фильтрации невыровненных прочтений, надо опять выбирать уникальные последовательности, в результате получилось 21052 уникальных прочтения. Для дополнительного сужения количества последовательностей применено «мягкое» сравнение (фильтрация), допускающее одну мутацию на 100 пар-оснований, количество уникальных прочтений уменьшилось до 14234.

Следующий шаг – обнаружение и удаление химер. Химерные последовательности представляют собой контиги, построенные из частичных считываний рРНК (справа налево, слева направо) разных видов микроорганизмов, но имеющих общие части. Этот этап основан на построении таксономии и отнесении прочтений к таксонам (классам). Очень редкие случаи считаются химерными и удаляются.

Следующий этап является обязательным - необходимо оценить количество отклоненных последовательностей после распознавания химер. Имея на входе 11119

1Исследование поддержано грантом Иркутского научного центра СО РАН № 4.2.
https://www.mothur.org/wiki/MiSeq_SOP

уникальных последовательностей, удалено химер 1.3% от общего количества, что считается разумным количеством. В целом, осуществлено сокращение объема обрабатываемой информации до 25% от исходного объема 44939 прочтений.

После этого этапа выполняется одна заключительная фильтрация для удаления митохондриальных и хлоропластных рРНК. Для этого необходимо классифицировать последовательности. Классификация проводится при помощи предобученной байесовской сети. В нашем примере удалено 1828 последовательностей, и уже окончательный набор содержал 9291 уникальное прочтение.

1.1. Анализ микробного сообщества. Для обработанных наборами последовательностей (групп, образцов) выполняются, в основном, четыре вида анализов. Последовательности организуются в операционные таксономические единицы (OTU), которые можно рассматривать как абстракцию понятия «вид». OTU содержит последовательности, имеющих заранее определенный уровень мутации, обычно не превышающий 3%. По нашим данным получено около тысячи OTU.

Первый вид анализа микробного сообщества это определение численности распознанных бактерий по таксономическим уровням в каждой OTU. Второй филогенетический анализ, направленный на определение расстояний между OTU, выражаемый в количестве мутаций. Третий, называемый «Альфа-разнообразие, измерение сходства групп на основе OTU или индивидуальных последовательностей, а также сходства OTU на основе состава групп. В результате получаются тепловые карты, пример которых изображен на рисунке 1.

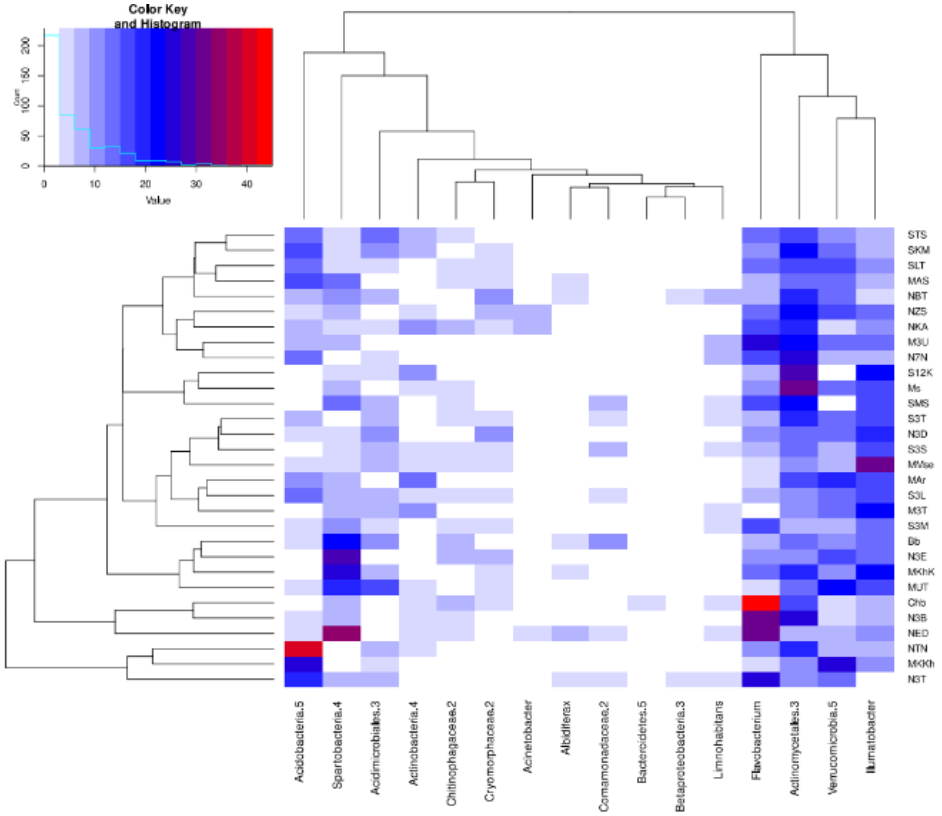

Рис.1. Пример тепловой карты

Четвертый анализ называется «Бета-разнообразие» и основывается на методе главных компонент и его аналогах, в результате строятся диаграммы групп OTU по сходству в пространстве осей главных компонент. Количество осей определяется последовательно, добавляя одну за другой и анализируя изменение R- и stress-критериев.

1.2. Особенности ручной реализации технологии. После применения технологии вручную выявился ряд проблем. Структура параметров и вывод команды Mothur достаточно сложные: необходимо отслеживать изменение имен файлов от команды к команде. После каждого выполнения команда Mothur добавляет суффиксы к именам входных файлов. Именование файлов зависит от входных параметров команды, например, от метода обработки данных. В частности, после применения команды align.seq к файлу с именем HXH779K01.shhh.trim.good.unique.fasta получаем
HXH779K01.shhh.trim.good.unique.align, HXH779K01 ... trim.good.unique.align.report,
и HXH779K01 ... im.good.unique.flip.accnos[2] .

Повторное применение операции к данным приводит к повторяющимся суффиксам. В Mothur включены инструменты отслеживания и передачи правильных имен файлов между командами, который поддерживает стандартные процедуры, реализуемые в виде сценариев. Вернувшись к предыдущему шагу для уточнения коэффициента, исследователь уже должен вручную вводить исходные имена файлов. После получения результатов в виде таблиц и диаграмм биологи обычно повторяют некоторые этапы фильтрации, исключая дополнительные OTU, например, которые похожи на митохондрии и хлоропласты. Иногда пользователи хотят заменить команду на аналогичную из другого пакета, например QUIIME2 или Usearch, где реализованы другие алгоритмы для некоторой процедуры. В этом случае необходимо преобразовывать данные из форматов Mothur в формат QUIIME2, Usearch и обратно.

Другое возможное, но менее частое отклонение от методики – это использование ранее обработанных OTU из ранних исследований, например, из образцов прошлых лет, отобранных на тех же локациях в рамках экологического мониторинга. Здесь требуется реализация процедуры для объединения содержимого OTU из разных исследований, т. е. программирование новых сценариев.

Визуализация результатов частично представлена в наборе команд Mothur, можно создавать векторные изображения SVG, но вид изображений, как правило, не настраивается. Приходится использовать внешнее программное обеспечение, такое, как R, для построения диаграмм желаемого качества. Наш опыт показывает, что, несмотря на время, потраченное на изучение методов построения диаграмм, большая часть времени уходит на преобразование и фильтрацию входных данных и уточнение параметров команд, генерирующих диаграммы.

1.3. Исследования-аналоги. Основная деятельность НИОКР в области NGS разделена на три основных направления: а) разработка новых эффективных алгоритмов, реализующих операции обработки данных и построения диаграмм, б) организация стандартизированных конвейеров облачных вычислений на кластерах класса HPC (High performance comuting), в) представление конвейеров в виде рабочих процессов, а также пользовательских интерфейсов для поддержки интерактивной обработки и оценки данных (моделирование вычислительного процесса).

[2] Имена файлов в начале усечены, чтобы соответствовать размерам текста.

Рассмотрению свойств вычислительных сред, как инструментов реализации методов биоинформатики, посвящена работа [8], где языки программирования Go, C++ и Java оценены с точки зрения простоты реализации алгоритмов NGS, потребления памяти и общей производительности вычислений; в результате выбран Go. В [9] разработан конвейер NGS для анализа ДНК вирусов в организме человека. Этот анализ позволил медицинским инженерам сосредоточить внимание на разработке вакцины. В статье [10] рассматривается реализация эвристического алгоритма для построения структур данных для упорядочения, перемещения и ориентации контигов с использованием дополнительной информации, что на следующих этапах позволяет создавать более длинные последовательности. В [21] рассматривается разработка пользовательского интерфейса для визуализации вычислительного процесса, управляемого данными. Результаты находят применение в системе поддержки принятия клинических решений. Пользовательский интерфейс предоставляет врачу возможность принятия решений и выдачи пояснений для пациентов. Пояснение представляется в виде списка структур различного типа (медицинские записи, результаты NGS, аутопсии ткани и т. д.), представленных в виде портлетов HTML5. Подробный обзор контроля качества, обнаружения исправления ошибок при обработке данных NGS представлен в [1].

Обзор методов HPC начнем с применения технологии BOINC для выравнивания последовательностей, представленной в [25], где отмасштабирован алгоритм Novoalign. В справочнике [11] рассмотрена проблемная область, описаны существующие подходы и уже реализованные методики, но нет упоминаются действительно реализованных сред облачных вычислений. В статье [16] содержится превосходный обзор текущих достижений в области NGS и смежных областях. Авторы сомневаются в возможности организации лабораторного HPC-центра на базе кластерных вычислений пользователями программного обеспечения NGS и предлагают заняться облачными вычислениями IaaS. В статье содержится также обзор существующих коммерческих платформ и платформ с открытым исходным кодом, позволяющих создавать конвейеры вычислительных процессов. Коммерческое программное обеспечение, в основном, реализует предопределенные конвейеры и является негибким, в то время как программное обеспечение с открытым исходным кодом позволяет реализовывать как стандартизованные конвейеры, так и предоставлять модули для отдельных операций и реализации облачных сервисов, т. е. инструментарий разработчика.

В [26] рассматривается программа Rainbow, поддерживающая облачную обработку данных NGS. По сути, Rainbow – это сценарий Perl, реализующий операцию map (разделение) над входными данными и reduce (соединение) для получения выходных агрегированных данных; разделенные части распределяются между узлами облачного сервиса Amazon EC2. Облачные узлы выполняют исключительно операцию выравнивания. В статье представлен также хороший обзор дистрибутивов виртуальных машин Linux для организации облачных сервисов и биоинформатических пакетов. Другой интересный обзор применения облачных вычислений представлен в [17]. Авторы обращают внимание на облачную технологию с открытым исходным кодом Open Stack и её инструментарий – Common Workflow Language (CWL) [2], используемый для представления вычислительного процесса в облаке. Подробный обзор технологий есть и в [3].

Существуют визуальные инструменты для генетического анализа, например, Galaxy [5], реализующие популярный подход «интерактивной веб-страницы», где данные импортируются и обрабатываются модулями. В Galaxy реализованы инструменты анализа существующих

скриптов (сценариев) и визуального представления модели потока данных (dataflow). Его основная цель – научить биологов проведению анализов данных NGS. Набор функций является расширяемым, что позволяет проводить специализированные исследования NGS. Проект находится в стадии активной разработки, авторы считают необходимым провести его интеграцию в разрабатываемую платформу. Другой полезный инструмент UGENE [24] – приложение с открытым исходным кодом, основанное на платформе QT5, также находится в активной разработке. UGENE визуализирует вычислительный процесс и генетические данные.

Основная критика существующих инструментов представлена в [19], где справедливо утверждается, что утилиты командной строки поддерживают больше функций и обладают большей гибкостью, чем визуальные инструменты. Авторы предлагают собственный визуальный инструмент VisPro, подключенный к облаку. Инструмент реализует гибкий подход к проектированию вычислений, предполагающий принципиальное участие пользователя в процессе построения, настройки и выполнения процесса анализа данных.

Подводя итоги этого краткого обзора, приходим к выводу, что для построения инфраструктуры анализа NGS есть хорошие открытые технологические наработки, и методы, используемые в исследованиях микробиома Байкала в Лимнологическом институте СО РАН, необходимо адаптировать к этим наработкам. Вышеупомянутые технологии позволяют действовать в рамках наших задач, решение которых требуют большей гибкости процесса вычислений и адаптации опыта пользователя-предметника.

2. Подход к автоматизации MiSeq SOP. Лимнологи выполняют действия как опытных биоинформатиков, так и пользователей-предметников [11], т. е. обрабатывают как исходные данные секвенирования, так и результаты обработки последовательностей, визуализируя, сравнивая и обобщая результаты. Высокопроизводительные вычисления (HPC) обычно основаны на двух популярных моделях программирования [11]: Map-Reduce и программировании сети задач. Первая подразумевает, что данные разделяемы на подмножества, обрабатываемые независимо, затем результаты параллельной обработки объединяются в агрегированный объект. Вторая модель, сеть задач, запускает модули на узлах кластера при готовности их водных данных, при этом модули также выполняются независимо.

Простые команды фильтрации Mothur запускаются параллельно, используя отдельные ядра процессора, но вычислительная сложность не кажется достаточно высокой, чтобы было рационально тратить время на их разделение на отдельные узлы кластера и последующее объединение. Основной причиной использования облачных сред здесь является объединение ресурсов оперативной памяти, если данные не помещаются в памяти одной рабочей станции. Некоторые алгоритмы фильтрации основаны на классификации, которая обрабатывает все генетические данные, используя случайные выборки (bagging). Переход соответствующих алгоритмов на SaaS потребует их замены на распределенную версию.

В целом, чтобы упростить создаваемую архитектуру облачных вычислений, на первом этапе НИОКР решено использовать модель выполнения очереди задач (dataflow), в которой вычислительные ресурсы выполняют отдельные задачи из сети модулей, представляющих вариант MiSeq SOP. В этой сети каждый узел является модулем пакета Mothur. Конкретный вычислительный процесс проектируется при помощи приложения Rapidminer studio, являющегося визуальным редактором сети dataflow-модулей. На рис. 2 представлена начальная часть процесса MiSeq SOP. Объединение с облачной инфраструктурой потребует передачи

данных между DaaS и SaaS, хранения объектов с метаданными, что также учитывается в нашей модели.

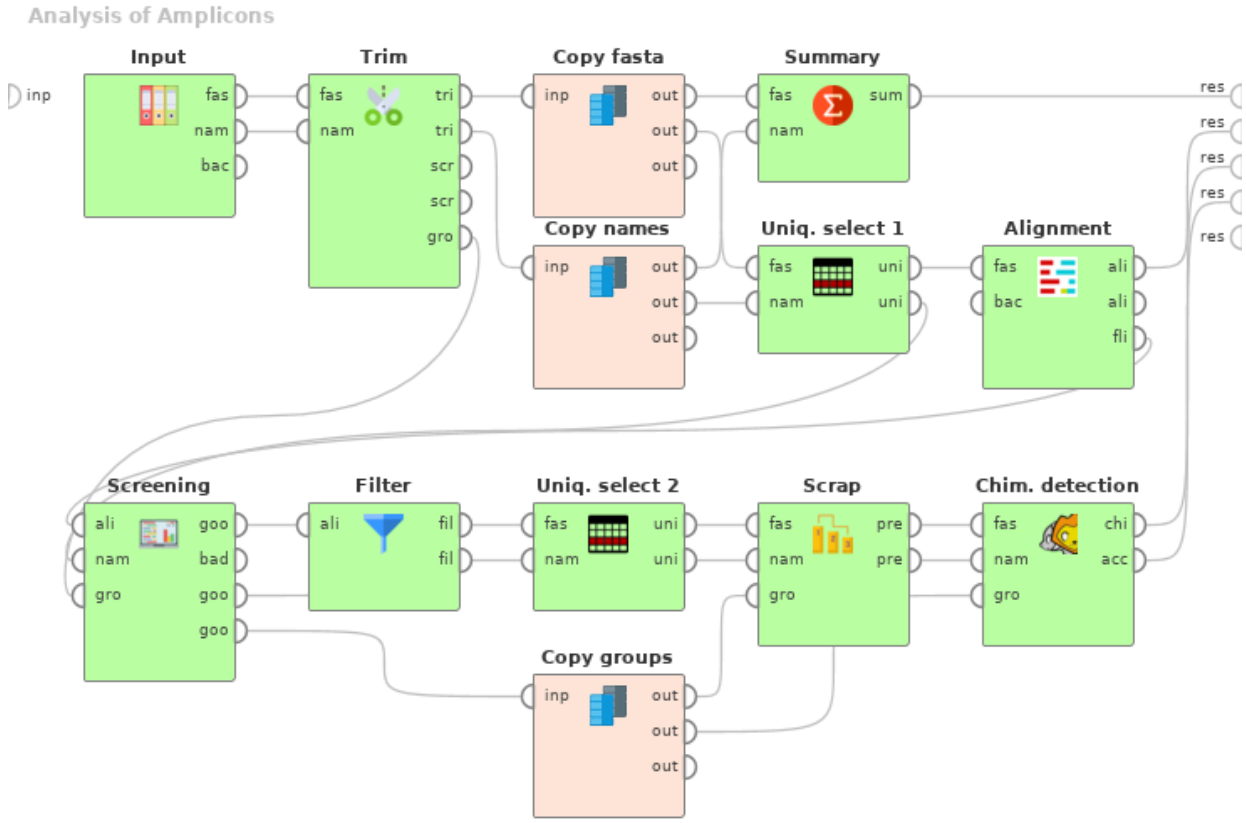

Рис. 2. Представление первых этапов MiSeq в виде модулей потока данных [7]

3. Концепция реализации. Обзор литературы показывает, что в настоящее время в активной разработке находятся два open-source проекта, связанных с автоматизацией анализа данных NGS: Galaxy и UGENE. Если взять Galaxy в качестве основного метода визуализации обработки данных, то необходимо реализовать модули Galaxy, адаптирующие команды Mothur, а также адаптировать визуализацию Galaxy к MiSeq SOP. С другой стороны, то же самое реализуемо и для UGENE, что позволит пользователю работать с более отзывчивым динамическим интерфейсом настольного приложения UGENE.

В [7] и [8] авторами предложена и частично реализована методика представления потока данных для всех команд Mothur с использованием Model Driven Architecture (MDA) для создания модулей Rapidminer studio. Согласно MDA, исходный код модулей генерируется на основе платформенно-зависимой модели (PSM), которая представляет разрабатываемое программное обеспечение в нотации, позволяющей прямую генерацию программного кода с помощью шаблонов и других императивных процедур. В нашем случае PSM представляет собой исходный код модулей потока данных на языке программирования Java.

Модель PSM порождается на основе платформенно-независимой модели (PIM), представляющей программное обеспечение на более абстрактном уровне, чем PSM. Она описывает отношения между сущностями, их объектные структуры, метаинформацию и т.д. Переход от PIM к PSM (трансформация модели) осуществляется при помощи логического вывода свойств PSM на основе фактов, представляющих PIM, и свойств платформы реализации, называемой моделью платформы (PM), в нашем случае это свойства среды программирования Java.

Некоторые свойства PIM, например, список полей объекта, создаются путем преобразования вычислительно независимой модели (CIM), еще более абстрактной модели, чем PIM. Модель CIM представляет программное обеспечение в виде объектов – команд Mothur. Преобразование – логический вывод, реализующий распознавание образов над данными CIM. Модель CIM также получается автоматически в результате анализа исходного кода C++ пакета Mothur. Процедура реализована в среде Python, где происходит сканирование исходного кода и выявление заданных структур. Каждая реализация команды анализируется с помощью набора регулярных выражений, организованных в сценарии.

3.1. Представление модельных данных в RDF. Данные исходной модели представлены на основе технологий Семантической паутины (Semantic Web). Модель и составляющие ее структуры идентифицируются глобально как ресурсы. Отношения между ресурсами и литералами выражаются с помощью предикатов стандартных и специально разработанных онтологий. Использование онтологий позволило стандартизировать структуры метаданных и, используя опыт разработчиков онтологий, сузить пространство поиска решений.

В облачном DaaS предполагается хранить файлы и их содержимое как объект с его метаданными. Современные стандарты OMG описывают спецификации преобразования реляционных метаданных UML, SysML в представление RDF. Таким образом, данные хранятся в обычных реляционных базах данных или базах данных «ключ-значение» совместно с их моделями. При проектировании облачного хранилища используется декларативный язык JHipster Domain Language [12] и его инструменты для создания структур баз данных, преобразования метаданных, формальных представлений онтологий для хранимых данных.

Метаданные хранимых объектов базы данных, в основном, описывают отношения между ресурсом и его атрибутами. Некоторые атрибуты, а именно внешние ключи, являются ссылками на другие ресурсы, которые также отражаются в метаданных. Между ресурсами встречаются редкие отношения, которые не хранятся в обычных базах данных. Эти отношения отражают, например, происхождение данных, дополнительные специальные атрибуты для конкретного объекта. Редкие связи встречается нечасто и появляются в отдельных исследованиях, поэтому изменение структуры реляционной базы данных для каждого такого случая не имеет смысла. Для хранения таких отношений будем использовать RDF-хранилище ClioPatria.

В формализации данных адаптированы следующие стандартизированные онтологии: онтология Fiend-of-a-friend (foaf), которая используется для представления информации об агентах: физических, юридических лицах, программ; Provenance (prov), используемая для ссылок между документами; Dublin Core (dc), которая используется для разметки метаданных опубликованных ресурсов; ресурсы DBPedia (dbr) относят данные к внешним глобально используемым классам и объектам-экземплярам; Open annotation (oa), которая используется в качестве онтологии представления содержания опубликованного документа; Bibliographic Ontology (bibo), используемая для разметки литературы. Для представления Mothur CIM и PIM разработаны две онтологии mothur и uml. Онтологии CIM и PIM используются для представления отношений между хранимыми объектами, как субъектами входных и выходных данных команд Mothur.

Используемые инструменты трансформации позволили авторам решить множество технических проблем, в том числе укомплектовать систему dataflow-визуализации актуальным набором команд Mothur, реализовать абстрактный механизм отображения свойств команд Mothur в программную среду. В рамках разработки получен набор сценариев преобразования, выраженных в виде объектов на языке программирования Logtalk [20]. Эти результаты можно использовать для генерации PSM и исходных кодов для внедрения команд Mothur в новые средства вычислений и визуализации.

3.2. Интеграция данных: вывод метаданных. Общепринятая форма представления полученных результатов NGS – это публикация в виде научных статей, отчетов и таблиц в различных форматах, таких, как HTML, Word, Excel, PDF. Дополнительная семантическая разметка RDF/RDFa в этих файлах позволит как исследователю, так и программному агенту использовать ранее полученные результаты в новых задачах. Разметка RDF документов является частью сервисов DaaS, поддерживающих также LOD (Открытые связанные данные), и обеспечивающих интеграцию с другими Интернет-ресурсами NGS. К данному моменту не найдено стандартизованного способа интеграции: есть только прототипы аннотационных ресурсов, такие, как BioSearch [13], реализованные на основе технологий BIO2RDF и LOD.

Служба LOD и необходимая гибкость программного обеспечения для научных исследований требуют, чтобы метаданные были связаны со всеми элементами данных NGS. Метаданные входных данных MiSeq SOP преобразуются при каждом применении команд в метаданные, описывающие объекты выходных данных. Для Mothur синтезируются правила автоматического вывода метаданных на основе анализа его исходного кода C++ и алгоритмов преобразования имен файлов. Для экономии памяти, расходуемой на метаданные, решено реализовать динамическое восстановление метаданных по структурам баз данных. Поскольку тысячи последовательностей организованы в файлы, группы и OTU, метаданные последовательностей дополняются всеми метаданными файла/группы/OTU. Метаданные каждой последовательности генерируются с использованием контекста своего хранилища, а именно имени файла fasta и отношения к своей группе, происхождению файла и т. д. Текущая архитектура сервисов обработки метаданных показана на рисунке 3.

Заключение. Предложен подход к созданию ИТ-инфраструктуры поддержки исследований микробиома озера Байкал на основе секвенирования следующего поколения. Хорошая база алгоритмов и программного обеспечения, уже созданных разработчиками, позволяет авторам реализовать среду, применяя адаптацию существующих методов, используемых биологами, к инструментарию проектирования вычислительных процессов. Для этого реализуется модель потока данных методики MiSeq, поддерживаемая прикладным пакетом Mothur. Модель интерпретируется в виде программных модулей различных визуальных сред и облачных сервисов. Преобразование осуществляется с использованием архитектуры, управляемой моделями, где трансформация модели реализована как система логического вывода, что позволяет быстро переходить от одной платформы к другой, сохраняя полученный формализованный опыт. На данном этапе существующая реализация ограничена спецификой программного обеспечения Mothur, также пока не было необходимости автоматизировать преобразование форматов данных.

Следующий этап НИРОКР связан с реализацией преобразования данных в форматы, поддерживаемые внешними по отношению к пакету Mothur программами, например,

открытым QIIME2 или коммерческим программным обеспечением Usearch. Эти пакеты дополняют средства Mothur в части функций визуализации и модулями обработки данных на отдельных стадиях MiSeq.

Набор задач, требующих решения, включает оптимизацию использования вычислительных ресурсов кластера, используемого биологами в настоящее время; планирование выполнения параллельных вычислений на основе анализа структуры процесса и свойств алгоритмов Mothur, а также реализацию контрольных точек для служб.

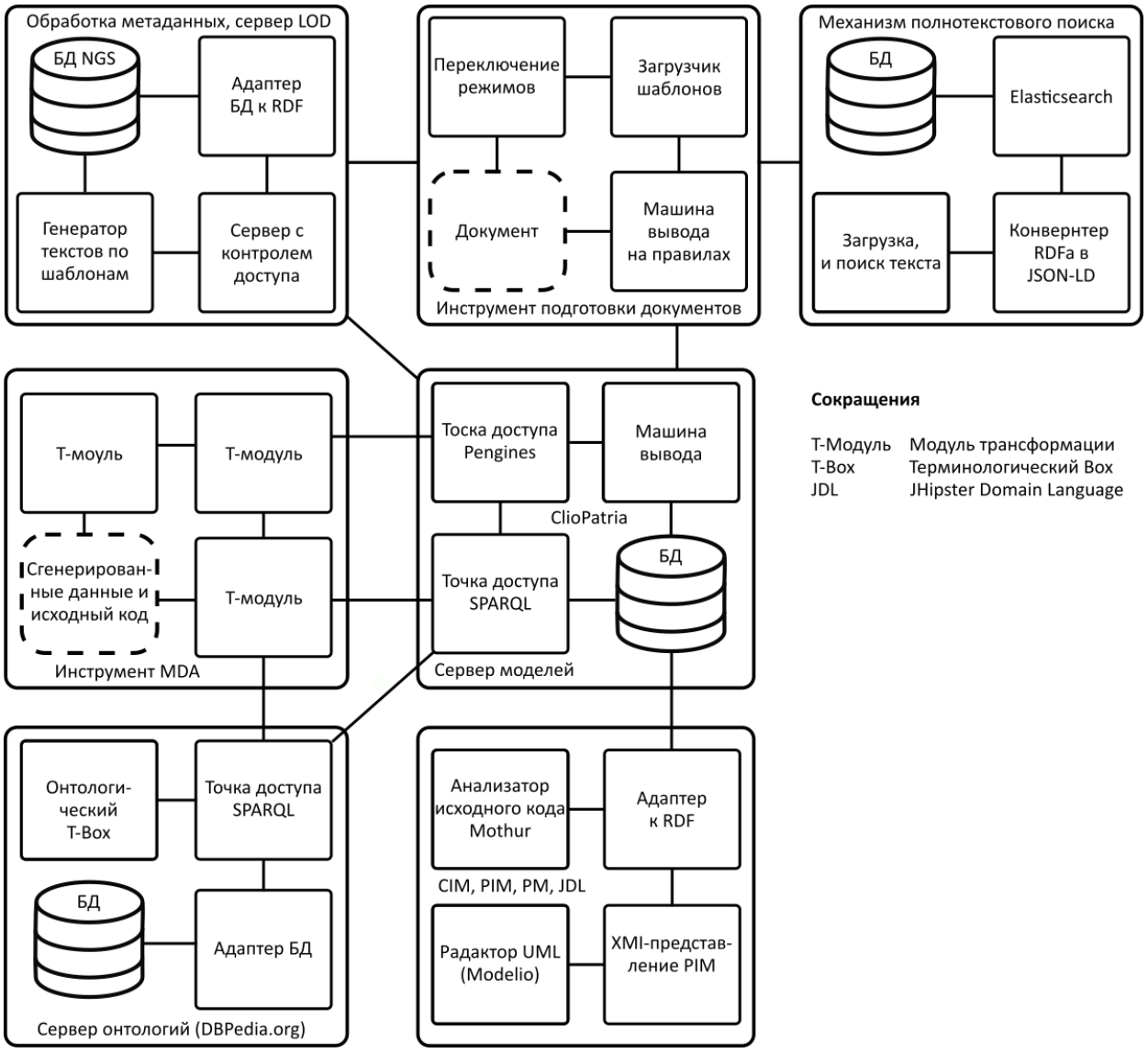

Рис.3. Архитектура облачных сервисов обработки метаданных

СПИСОК ЛИТЕРАТУРЫ

1. Боекхорст Р, Науменко Ф. М., Орлова Н. Г., Галиева Э. Р., Спицина А. М., Чадаева И. В., Орлов Ю. Л., Абнизова И. И. Вычислительные проблемы анализа ошибок коротких прочтений ДНК при секвенировании следующего поколения // Вавиловский журнал генетики и селекции. 2016. №6 (20). С. 746-755.

2. Amstutz P., Crusoe M. R., Tijanic N., Chapman B. Common workflow language. V. 1.0. 2016. DOI:10.6084/m9.figshare.3115156.v2

3. Baker Q. B., Al-Rashdan W., Jararweh Y. Cloud-based tools for next-generation sequencing data analysis //Procs. of Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS). Valencia. 2018. Pp. 99–105. DOI:10.1109/SNAMS.2018.8554515

4. Bashenkhaeva M. V., Zakharova Y. R., Petrova D. P. Sub-ice microalgal and bacterial communitie in freshwater lake Baikal //Environmental Microbiology. 2015. Vol.70. № 3. Pp. 751–765. DOI:10.1007/s00248-015-0619-2

5. Batut B., Hiltemann S., Bagnacani A., Baker D., Bhardwaj V. Community-driven data analysis training for biology cell systems. 2018. DOI:10.1016/j.cels.2018.05.012

6. Cherkashin E., Shigarov A., Paramonov V. Representation of MDA transformation with logical objects //International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON). Novosibirsk. Russia. 2019. Pp. 0913–0918. DOI:10.1109/SIBIRCON48586.2019.8958008/

7. Cherkashin E., Shigarov A., Malkov F., Morozov A. An instrumental environment for metagenomic analysis // Information Technologies in the Research of Biodiversity. Springer Proceedings in Earth and Environmental Sciences. 2019. Pp. 151–158. DOI:10.1007/978-3-030-11720-7.

8. Costanza P., Herzeel C., Verachtert W. A comparison of three programming languages for a full-fledged next generation sequencing tool // BMC Bioinformatics. 2019. Vol. 20. № 1. 301p. DOI:10.1186/s12859-019-2903-5

9. Gong Y.-N., Chen G.-W., Yang S.-L., Lee C.-J. A next-generation sequencing data analysis pipeline for detecting unknown pathogens from mixed clinical samples and revealing their genetic diversity. 2016. DOI:10.1371/journal.pone.0151495

10. Gritsenko A. A., Nijkamp J. F., Reinders M.J.T., D. de Ridder. GRASS: a generic algorithm for scaffolding next-generation sequencing assemblies // Bioinformatics. 2012. Vol.28. № 11. Pp. 1429–1437. DOI:10.1093/bioinformatics/bts175

11. Guo X., Yu N., Li B., Pan Y. Cloud computing for next-generation sequencing data analysis. in Computational Methods for Next Generation Sequencing Data Analysis. John Wiley & Sons. 2016. Pp. 3-24.

12. Halin A., Nuttinck A., Acher M., Devroey X., Perrouin G., Heymans P. Yo. JHipster: a playground for web-apps analyses // Procs. of the Eleventh international workshop on variability modelling of software-intensive systems. VAMOS'17. ACM. New York. 2017. Pp. 44–51. DOI:10.1145/3023956.3023963.

13. Hu W., Qiu H., Huang J., Dumontier M. BioSearch: a semantic search engine for Bio2RDF. Database. Vol. 2017. bax059. DOI:10.1093/database/bax059.

14. Johnston W. M., Hanna J. R. P., Millar R. J. Advances in dataflow programming languages //ACM Computing Surveys. 2004. Vol. 36. Pp 1–34. DOI:10.1145/1013208.1013209

15. Kozich J. J., Westcott S. L., Baxter N. T., Highlander S. K., Schloss P. D. Development of a dual-index sequencing strategy and curation pipeline for analyzing amplicon sequence data on the MiSeq Illumina sequencing platform. Applied and Environmental Microbiology. 2013. Vol. 79. № 17. Pp. 5112–5120. DOI:10.1128/AEM.01043-13

16. Kwon T., Yoo W. G., Lee W. Next-generation sequencing data analysis on cloud computing // Genes Genom. 2015. Vol. 37. Pp. 489–501. DOI:10.1007/s13258-015-0280-7

17. Langmead B., Nellore A. Cloud computing as a platform for genomic data analysis and collaboration // Nat Rev Genet. 2018. Vol. 19. № 4. Pp. 208–219. DOI:10.1038/nrg.2017.113

18. Mikhailov I. S., Zakharova Y. R., Bukin Y. S. Co-occurrence networks among bacteria and microbial eukaryotes of lake Baikal during a spring phytoplankton bloom. Microbial Ecology. 2019. Vol. 77. Pp. 96–109. DOI:10.1007/s00248-018-1212-2.

19. Milicchio F., Rose R., Bian J. Visual programming for next-generation sequencing data analytics // BioData Mining. 2016. Vol. 9. № 16. DOI:10.1186/s13040-016-0095-3.

20. Moura P. .Programming patterns for Logtalk parametric objects //Applications of Declarative Programming and Knowledge Management. A. Abreu. D. Seipel eds. INAP 2009. Lecture Notes in Computer Science. Vol. 6547. Berlin. Heidelberg. Pp. 52–69. 2009. DOI:10.1007/978-3-642-20589-7_4

21. Müller H., Reihs R., Posch A. E., Kremer A., Ulrich D., Zatloukal K. Data driven GUI design and visualization for a NGS based clinical decision support system. //Procs. of 20th International Conference Information Visualisation. 2016. Lisbon. Portugal. Pp. 355–360. DOI:10.1109/IV.2016.79

22. Pereira R., Oliveira J., Sousa M. Bioinformatics and computational tools for next-generation sequencing analysis in clinical genetics. J. Clin. Med.. Vol. 9. № 1. Pp. 1–30. 2020. DOI:10.3390/jcm9010132

23. Quast Ch., Pruesse E., Yilmaz P., Gerken J. The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. Nucleic Acids Research. 2013. Vol. 41. D590–D596. DOI:10.1093/nar/gks1219.

24. Rose R., Golosova O., Sukhomlinov D., Tiunov A., Prosperi M. Flexible design of multiple metagenomics classification pipelines with UGENE //Bioinformatics. 2019. Vol. 35. № 11. Pp. 1963–1965. DOI:10.1186/s13040-016-0095-3

25. Srimani J. K., Wu P., Phan J. H., Wang M. D. A distributed system for fast alignment of next-generation sequencing data //2010 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW). Hong Kong. 2010. Pp. 579-584. DOI:10.1109/BIBMW.2010.5703865

26. Zhao S., Watrous K., Zhang Ch., Zhang B. Cloud computing for next-generation sequencing data analysis // Cloud Computing–Architecture and Applications. IntechOpen Limited. 2017. Pp. 29–51. DOI:10.5772/66732

UDK 004.6::574.5

INFORMATION INFRASTRUCTURE FOR SUPPORTING BAIKAL MICROBIOME RESEARCH

Evgeny A. Cherkashin
Matrosov Institute for System Dynamics and Control theory of
Siberian Branch of Russian Academy of Sciences
e-mail: eugeneai@icc.ru
Alexey O. Shigarov
Irkutsk Scientific Center of Siberian Branch of Russian Academy of Sciences
Irkutsk, Russia
e-mail: shigarov@icc.ru
664033, Irkutsk, st. Lermontov, 134.
Vasily V. Khristyuk
National Research Irkutsk State Technical University
e-mail: khr@icc.ru
664074, Irkutsk, st. Lermontov, 83.

Abstract. A problem of construction of a research environment for Next Generation Sequencing data processing is considered. The environment comprises cloud data storage (DaaS) and computational services (SaaS and PaaS), as well as visualization, and data integration services. We are integrating existing open-source technologies to support MiSeq SOP (standard operational procedure), which is to allow domain specialists, biologists, to process data independently. For the realization of the integration, formal models of the SOP are constructed, automatically processed (transformed) into source code of new components. The technique of the transformation is based on Model Driven Architecture principles and logical inference of the derived models and the code. The current results are presented and discussed.

Keywords: next generation sequencing, big data, model driven architecture, linked open data, problem solving

Referens

1. Boekhorst R., Naumenko F. M., Orlova N. G., Galieva E. R., Spitsina A. M. Computational problems of analysis of short next generation sequencing reads // Vavilovskii Zhurnal Genetiki i Selektsii = Vavilov Journal of Genetics and Breeding. 2016. Vol. 20. № 6. Pp. 746–755. DOI:10.18699/VJ16.191

2. Amstutz P., Crusoe M. R., Tijanic N., Chapman B. Common workflow language. V. 1.0. 2016. DOI:10.6084/m9.figshare.3115156.v2

3. Baker Q. B., Al-Rashdan W., Jararweh Y. Cloud-based tools for next-generation sequencing data analysis //Procs. of Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS). Valencia. 2018. Pp. 99–105. DOI:10.1109/SNAMS.2018.8554515

4. Bashenkhaeva M. V., Zakharova Y. R., Petrova D. P. Sub-ice microalgal and bacterial communitie in freshwater lake Baikal, Russia. Environmental Microbiology. Vol.70. № 3. 2015. Pp. 751–765. DOI:10.1007/s00248-015-0619-2

5. Batut B., Hiltemann S., Bagnacani A., Baker D., Bhardwaj V. Community-driven data analysis training for biology cell systems. 2018. DOI:10.1016/j.cels.2018.05.012

6. Cherkashin E., Shigarov A., Paramonov V. Representation of MDA transformation with logical objects //International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON). Novosibirsk. Russia. 2019. Pp. 0913–0918. DOI:10.1109/SIBIRCON48586.2019.8958008/

7. Cherkashin E., Shigarov A., Malkov F., Morozov A. An instrumental environment for metagenomic analysis // Information Technologies in the Research of Biodiversity. Springer Proceedings in Earth and Environmental Sciences. 2019. Pp. 151–158. DOI:10.1007/978-3-030-11720-7.

8. Costanza P., Herzeel C., Verachtert W. A comparison of three programming languages for a full-fledged next generation sequencing tool // BMC Bioinformatics. 2019. Vol. 20. № 1. 301p. DOI:10.1186/s12859-019-2903-5

9. Gong Y.-N., Chen G.-W., Yang S.-L., Lee C.-J. A next-generation sequencing data analysis pipeline for detecting unknown pathogens from mixed clinical samples and revealing their genetic diversity. 2016. DOI:10.1371/journal.pone.0151495

10. Gritsenko A. A., Nijkamp J. F., Reinders M.J.T., D. de Ridder. GRASS: a generic algorithm for scaffolding next-generation sequencing assemblies // Bioinformatics. 2012. Vol.28. № 11. Pp. 1429–1437. DOI:10.1093/bioinformatics/bts175

11. Guo X., Yu N., Li B., Pan Y. Cloud computing for next-generation sequencing data analysis. in Computational Methods for Next Generation Sequencing Data Analysis. John Wiley & Sons. 2016. Pp. 3-24.

12. Halin A., Nuttinck A., Acher M., Devroey X., Perrouin G., Heymans P. Yo. JHipster: a playground for web-apps analyses // Procs. of the Eleventh international workshop on variability modelling of software-intensive systems. VAMOS'17. ACM. New York. 2017. Pp. 44–51. DOI:10.1145/3023956.3023963.

13. Hu W., Qiu H., Huang J., Dumontier M. BioSearch: a semantic search engine for Bio2RDF. Database. Vol. 2017. bax059. DOI:10.1093/database/bax059.

14. Johnston W. M., Hanna J. R. P., Millar R. J. Advances in dataflow programming languages //ACM Computing Surveys. 2004. Vol. 36. Pp. 1–34. DOI:10.1145/1013208.1013209

15. Kozich J. J., Westcott S. L., Baxter N. T., Highlander S. K., Schloss P. D. Development of a dual-index sequencing strategy and curation pipeline for analyzing amplicon sequence data on the MiSeq Illumina sequencing platform. Applied and Environmental Microbiology. 2013. Vol. 79. № 17. Pp. 5112–5120. DOI:10.1128/AEM.01043-13

16. Kwon T., Yoo W. G., Lee W. Next-generation sequencing data analysis on cloud computing // Genes Genom. 2015. Vol. 37. Pp. 489–501. DOI:10.1007/s13258-015-0280-7

17. Langmead B., Nellore A. Cloud computing as a platform for genomic data analysis and collaboration // Nat Rev Genet. 2018. Vol. 19. № 4. Pp. 208–219. DOI:10.1038/nrg.2017.113

18. Mikhailov I. S., Zakharova Y. R., Bukin Y. S. Co-occurrence networks among bacteria and microbial eukaryotes of lake Baikal during a spring phytoplankton bloom. Microbial Ecology. 2019. Vol. 77. Pp. 96–109. DOI:10.1007/s00248-018-1212-2.

19. Milicchio F., Rose R., Bian J. Visual programming for next-generation sequencing data analytics // BioData Mining. 2016. Vol. 9. № 16. DOI:10.1186/s13040-016-0095-3.

20. Moura P. .Programming patterns for Logtalk parametric objects //Applications of Declarative Programming and Knowledge Management. A. Abreu. D. Seipel eds. INAP 2009. Lecture Notes in Computer Science. Vol. 6547. Berlin. Heidelberg. Pp. 52–69. 2009. DOI:10.1007/978-3-642-20589-7_4

21. Müller H., Reihs R., Posch A. E., Kremer A., Ulrich D., Zatloukal K. Data driven GUI design and visualization for a NGS based clinical decision support system. //Procs. of 20th International Conference Information Visualisation. 2016. Lisbon. Portugal. Pp. 355–360. DOI:10.1109/IV.2016.79

22. Pereira R., Oliveira J., Sousa M. Bioinformatics and computational tools for next-generation sequencing analysis in clinical genetics. J. Clin. Med.. Vol. 9. № 1. Pp. 1–30. 2020. DOI:10.3390/jcm9010132

23. Quast Ch., Pruesse E., Yilmaz P., Gerken J. The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. Nucleic Acids Research. 2013. Vol. 41. D590–D596. DOI:10.1093/nar/gks1219.

24. Rose R., Golosova O., Sukhomlinov D., Tiunov A., Prosperi M. Flexible design of multiple metagenomics classification pipelines with UGENE //Bioinformatics. 2019. Vol. 35. № 11. Pp. 1963–1965. DOI:10.1186/s13040-016-0095-3

25. Srimani J. K., Wu P., Phan J. H., Wang M. D. A distributed system for fast alignment of next-generation sequencing data //2010 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW). Hong Kong. 2010. Pp. 579-584. DOI:10.1109/BIBMW.2010.5703865

26. Zhao S., Watrous K., Zhang Ch., Zhang B. Cloud computing for next-generation sequencing data analysis // Cloud Computing–Architecture and Applications. IntechOpen Limited. 2017. Pp. 29–51. DOI:10.5772/66732