



openEuler 21.09

Технический информационный документ

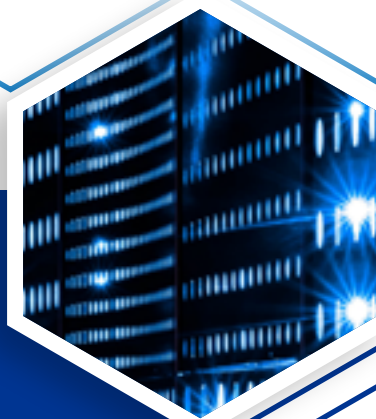


СОДЕРЖАНИЕ

<u>01</u>	<u>Введение</u>	<u>05</u>	<u>07</u>	<u>Расширенные функции</u>	<u>28</u>
<u>02</u>	<u>Архитектура платформы</u>	<u>09</u>	<u>08</u>	<u>Управление сообществом</u>	<u>33</u>
<u>03</u>	<u>Рабочая среда</u>	<u>12</u>	<u>09</u>	<u>Авторские права</u>	<u>41</u>
<u>04</u>	<u>Инновации ядра</u>	<u>14</u>	<u>10</u>	<u>Товарные знаки</u>	<u>41</u>
<u>05</u>	<u>Облачная база</u>	<u>19</u>	<u>11</u>	<u>Приложения</u>	<u>41</u>
<u>06</u>	<u>Инновационные разработки для конкретных сценариев</u>	<u>24</u>			

01/

Введение



Из простой серверной операционной системы (ОС) openEuler эволюционировала в ОС цифровой инфраструктуры, которая подходит для любого сервера, развертывания в качестве встроенного решения, а также для сценариев облачных и периферийных вычислений. Это безопасная, стабильная и простая в использовании ОС с открытым исходным кодом, совместимая с рядом вычислительных архитектур. openEuler подходит для применения в области операционных технологий (ОТ) и позволяет объединить ОТ с информационно-коммуникационными технологиями (ИКТ).

Объединившись в сообществе openEuler, разработчики со всего мира создают открытую, диверсифицированную экосистему разработки программного обеспечения и архитектуры для всех сценариев развертывания цифровой архитектуры. За плечами сообщества обширный опыт участия в разработке программного обеспечения, аппаратных платформ и приложений для предприятий.

Официально созданное 31 декабря 2019 года сообщество openEuler с открытым исходным кодом изначально сконцентрировало свою деятельность на инновациях в области диверсифицированных вычислительных архитектур.

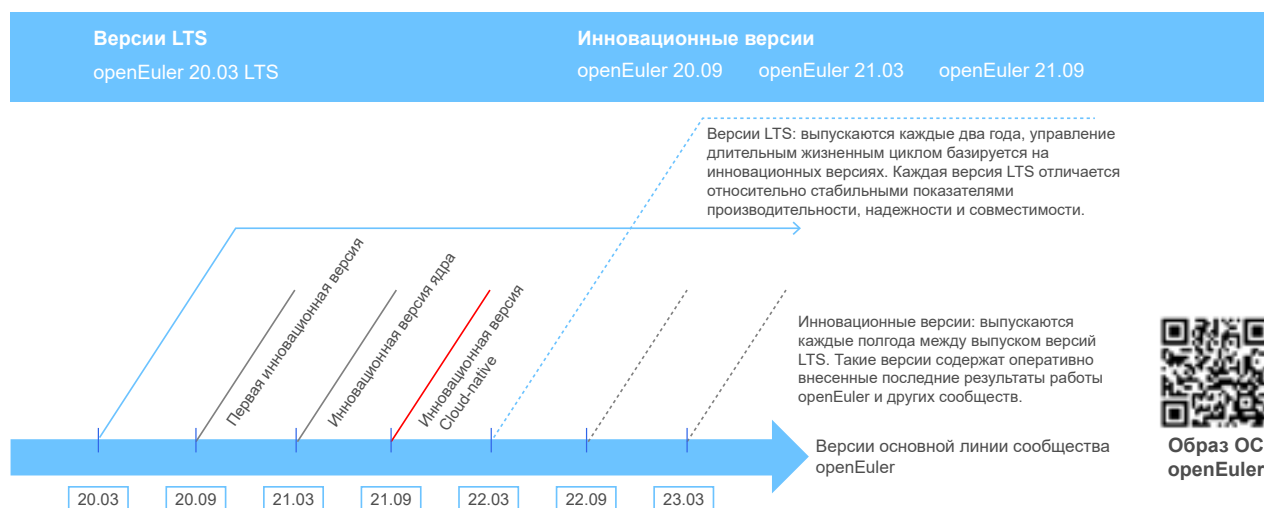
30 марта 2020 г. была выпущена версия openEuler 20.03 с долгосрочной поддержкой программного обеспечения (Long Term Support; LTS), ставшая новым дистрибутивом Linux с возможностью независимого развития технологий.

Позже 30 сентября того же года была выпущена инновационная версия openEuler 20.09 как результат совместной работы ряда компаний, команд и независимых разработчиков в сообществе openEuler. Релиз openEuler 20.09 не только знаменует важный этап в развитии сообщества openEuler, но еще и открывает историю создания программного обеспечения с открытым исходным кодом в Китае.

31 марта 2021 г. выпущена очередная инновационная версия ядра openEuler 21.03. Данная версия, улучшенная в соответствии с версией ядра Linux 5.10, выделяется рядом новых функций, среди которых обновление ядра в реальном времени и расширение памяти с многоуровневой организацией. Эти отличительные особенности улучшают производительность многоядерных систем и позволяют реализовать вычислительную мощность на уровне тысячи ядер.

К 30 сентября 2021 года появился релиз openEuler 21.09. Эта расширенная версия подходит для всех сценариев, включая периферийное и встроенное развертывание. Расширяющая возможности серверов и облачных вычислений версия соединила в себе ряд ключевых технологий, в том числе алгоритмы планирования центрального процессора с использованием подхода cloud-native для гибридного развертывания сервисов и KubeOS для контейнеров.

Управление версиями openEuler

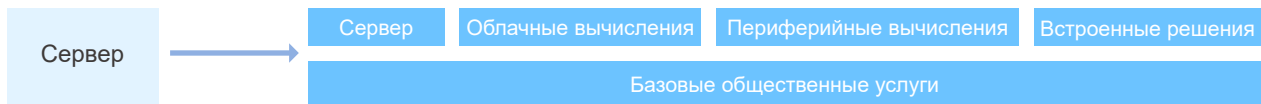


Как платформа разработки ОС сообщество openEuler выпускает обновления версий с долгосрочной поддержкой программного обеспечения (Long Term Support; LTS) каждые два года. Каждая версия LTS содержит расширенные возможности и представляет собой безопасную, стабильную и надежную операционную систему, ориентированную на корпоративных пользователей.

Сообщество openEuler внедряет в релизы только наработанные практикой технологии. В новых выпускаемых каждые 6 месяцев версиях содержатся последние технические достижения openEuler и других сообществ. Любая инновация сначала проверяется в сообществе openEuler в рамках отдельного открытого проекта, затем проверенные функции добавляются в каждый новый релиз, а разработчики сообщества получают исходный код.

Технические функционалы сначала тестируются в сообществе, после чего включаются в каждый выпуск openEuler. Кроме того, при создании очередной версии учитываются отзывы пользователей сообщества, которые дают возможность отточить существующие технологии. Платформа выпуска релизов openEuler является одновременно и инкубатором новых технологий. В симбиозе эти технологии являются движущей силой появления новых версий.

Инновационная платформа для всех сценариев

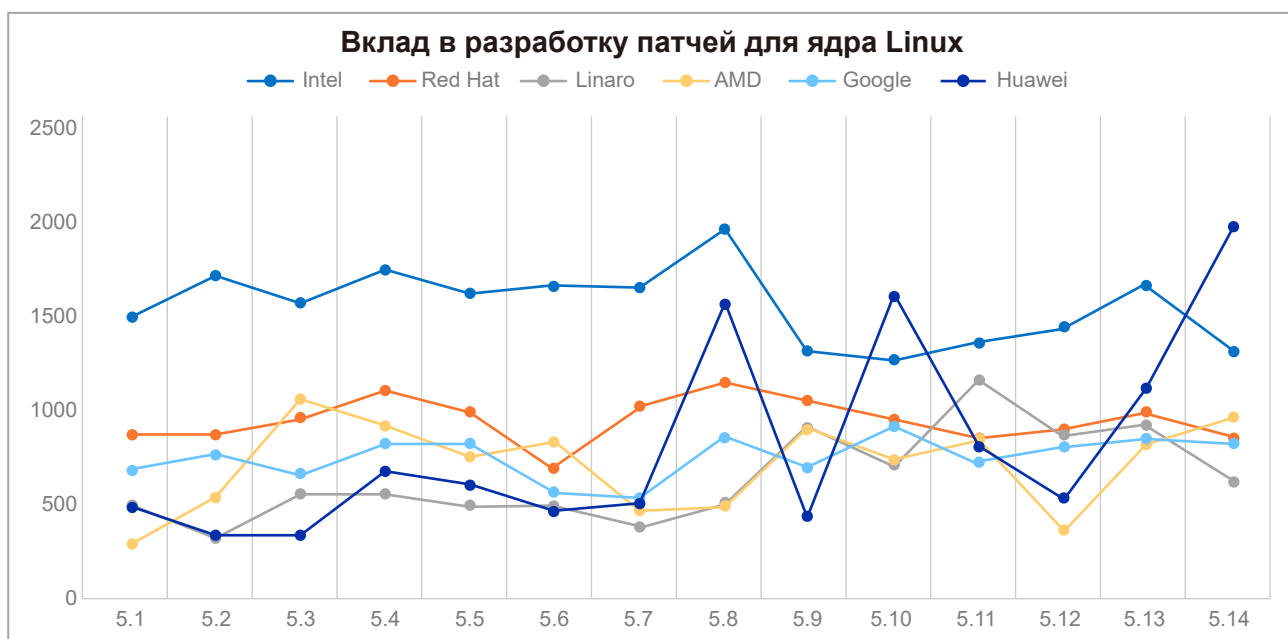


openEuler поддерживает архитектуры процессоров x86, ARM, RISC-V и, следуя своей линии постоянного расширения экосистемы вычислительных ресурсов, в будущем сообщество планирует поддерживать и другие бренды (PowerPC, SW64).

Сообщество openEuler стало пристанищем для растущего числа специальных групп по решению определенных проблем (Special Interest Group; SIG), которые создаются для работы над теми или иными задачами расширения функционалов ОС, включая серверные и встроенные сценарии, облачные и периферийные вычисления. Охватывая все сценарии, операционная система openEuler включает версию 21.09 Edge для периферийных вычислений и 21.09 Embedded для встроенного развертывания.

ОС является идеальным выбором для партнеров экосистемы, пользователей и разработчиков, которые планируют расширить возможности в конкретных сценариях. Создав унифицированную ОС, поддерживающую множество устройств, openEuler организует единый процесс разработки приложений для любых сценариев.

Непрерывный вклад в разработку ядра Linux

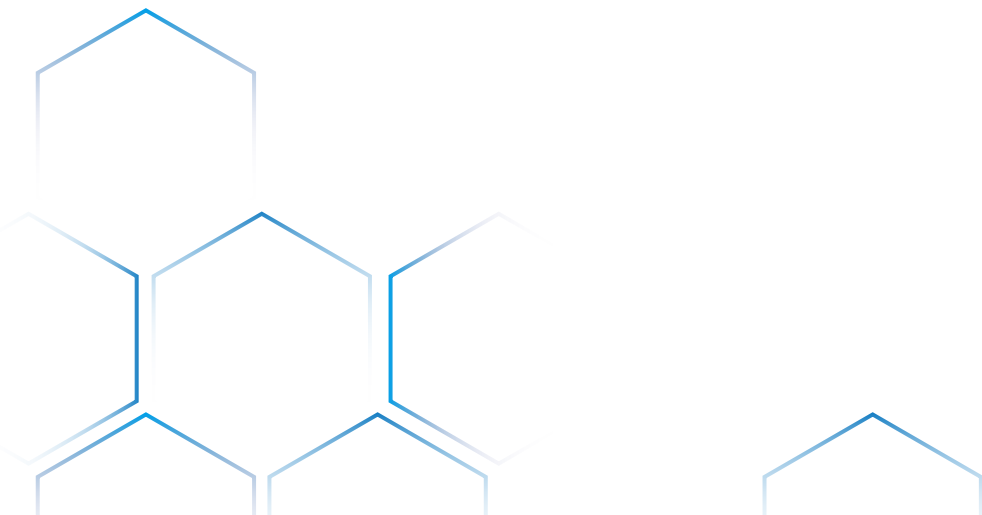


Активно участвуя в разработке ядра Linux, выделенная для этого группа вкладывает усилия в развитие архитектуры процессоров, усовершенствованный интерфейс управления конфигурацией и питанием (Advanced Configuration and Power Interface; ACPI), создание решений для управления памятью, оптимизацию работу файловых систем, разработку носителей и документации на ядро, исправление ошибок и пересборку кода. За последнее десятилетие разработчики openEuler внедрили в ядро Linux более 17,000 патчей.

Сообщество openEuler сделало наибольший вклад в разработку кода для ядер Linux 5.10 и 5.14 среди остальных мировых сообществ. openEuler постоянно участвует в инновационных разработках ядра, организованных исходными сообществами.

Открытость и прозрачность: цепочка поставок программного обеспечения с открытым исходным кодом

В процесс создания ОС с открытым исходным кодом вовлечены этапы консолидации и оптимизации цепи поставок. Чтобы создать надежное открытое программное обеспечение или крупномасштабную коммерческую ОС, openEuler соблюдает все этапы управления жизненным циклом, включая сборку, верификацию и распространение. openEuler регулярно анализирует зависимости своих программных пакетов в реальных условиях применения, упорядочивает коды от исходных сообществ в отношении всех программных пакетов и проверяет свой исходный код, сравнивая его с кодом исходного сообщества. Сборка, проверка зависимостей в среде выполнения и сравнение с кодами исходных сообществ разработчиков свободного программного обеспечения формируют замкнутый цикл полного прозрачного управления цепью поставок ПО.



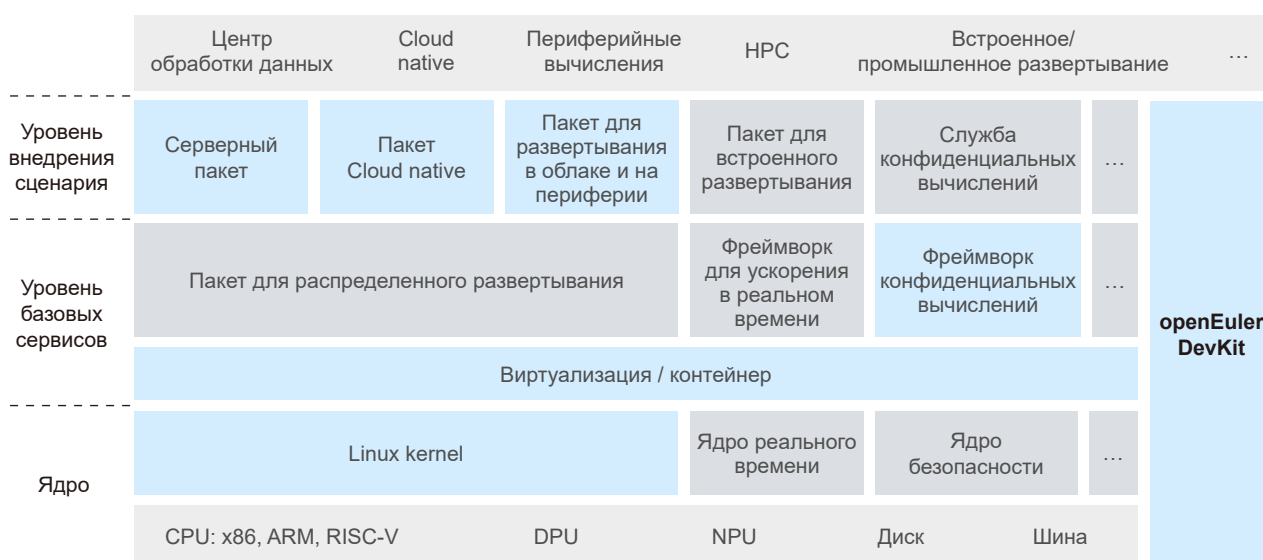
02/

Архитектура платформы



Каркас системы

openEuler — это передовая платформа разработки ОС с открытым исходным кодом, опирающаяся на инновациях ядра и серьезной облачной базе и подходящая для всех сценариев. При ее создании использовались новейшие тенденции в области шин взаимодействия и сред хранения данных. Платформа включает в себя распределенный движок аппаратного ускорения в режиме реального времени и базовые сервисы. Обладая конкурентными преимуществами в сценариях периферийного и встроенного развертывания, платформа станет первым шагом навстречу созданию полноценной цифровой инфраструктуры ОС.



Инновации в области ядра:

- **Усовершенствованное планирование cloud-native:** openEuler подходит для гибридных развертываний облачных онлайн- и офлайн-сервисов. Разработанный инновационный алгоритм планирования центрального процессора обеспечивает приоритетное выделение ресурсов процессора в режиме реального времени и подавление колебаний задержки (jitter) при предоставлении онлайн-сервисов. Кроме того, его инновационный алгоритм освобождения ресурсов памяти, предотвращающий проблему нехватки памяти (Out of Memory; OOM), обеспечивает надежную работу онлайн-сервисов согласно их высокой приоритетности.
- **EulerFS:** это новая файловая система, которая предназначена для модулей NVDIMM (Non-Volatile Dual In-line Memory Module). В файловой системе применяются «мягкие» обновления и каталоги dual-view, которые ускоряют синхронизацию метаданных файлов и, таким образом, повышают производительность чтения и записи файлов.
- **etMem для расширения памяти с многоуровневой организацией:** с помощью функции подкачки памяти в пользовательском режиме отброшенные холодные данные переключаются на хранение в пользовательской памяти в соответствии с заданной политикой. Подкачка памяти в пользовательском режиме обеспечивает более высокую производительность, чем подкачка памяти в режиме ядра, и весь процесс подкачки прозрачен для пользователей.

Облачная база:

- **KubeOS для контейнеров:** в сценариях cloud native операционная система развертывается и поддерживается в контейнерах, что позволяет управлять ОС на основе Kubernetes, так же, как это происходит с контейнерами служб.
- **Решение безопасного контейнера:** по сравнению с традиционным решением Docker+QEMU, безопасный контейнер iSulad+shimv2+StratoVirt сокращает затраты ресурсов памяти на служебные процессы и время загрузки на 40%.
- **Инструмент eggo для развертывания в двух плоскостях:** для гибридных кластеров ARM и x86 операционную систему можно установить одним щелчком кнопкой мыши, а развернуть кластер из 100 узлов всего за 15 минут.

Инновации для конкретных сценариев:

- **Периферийные вычисления:** для данного сценария подготовлена версия openEuler 21.09 Edge. В ней интегрирован фреймворк KubeEdge+, в котором эффективно сочетаются возможности периферии и облака. Версия позволяет управлять и настраивать периферийные и облачные сервисы с использованием унифицированных процессов, а также предоставляет другие функционалы.
- **Встроенное развертывание:** для таких сценариев выпущена версия openEuler 21.09 Embedded, которая позволяет сжимать изображения до 5 МБ и загружать их за 5 секунд.

Процветающая экосистема сообщества:

- **Настольные среды:** UKUI, DDE и Xfce.
- **openEuler DevKit:** поддерживает миграцию ОС, оценку совместимости и различные средства разработки, такие как secPaver, которые упрощают настройку функционалов безопасности.

Платформа как основа

Сообщество разработчиков ПО с открытым исходным кодом openEuler поддерживает партнерские отношения с выше- и нижестоящими организациями, совместно воплощая идею разработки и внедрения версий ОС openEuler.





03/

Рабочая среда

Серверы

Чтобы установить ОС openEuler на физическую машину, необходимо удостовериться, что она отвечает требованиям к совместимости и аппаратному обеспечению. Полный список требований можно найти на странице <https://ru.openeuler.org/ru/>.

Параметр	Требования к конфигурации
Архитектура	ARM64, x86_64
Память	Не менее 4 ГБ
Диски	Не менее 20 ГБ

Виртуальные машины

openEuler поддерживает следующие виртуальные машины (ВМ):

- centos 7.9 qemu 1.5.3-175.el7 libvirt 5.0.0-1.el7
- centos 8 qemu 5.1.0-20.el8 libvirt 6.6.0-7.3.el8
- fedora 29 qemu 3.0.0-1.fc29 libvirt 4.7.0-1.fc29
- fedora 32 qemu 4.2.0-7.fc32 libvirt 6.1.0-2.fc3

Параметр	Требования к конфигурации
Архитектура	ARM64, x86_64
Центральный процессор	2 процессора
Память	Не менее 4 ГБ
Диски	Не менее 20 ГБ

Периферийные устройства

Чтобы установить ОС openEuler на периферийное устройство, необходимо удостовериться, что оно отвечает требованиям к совместимости и аппаратному обеспечению.

Параметр	Требования к конфигурации
Архитектура	ARM64, x86_64
Память	Не менее 4 ГБ
Диски	Не менее 20 ГБ

Встроенные устройства

Чтобы установить ОС openEuler на встроенное устройство, необходимо удостовериться, что оно отвечает требованиям к совместимости и минимальным требованиям к аппаратному обеспечению.

Параметр	Требования к конфигурации
Архитектура	ARM64, ARM32
Память	Не менее 128 МБ
Диски	Не менее 256 МБ



04/

Инновации ядра

Новые функции в ядре openEuler

Версия openEuler 21.09 работает на ядре Linux 5.10, и при ее разработке применены 12 инновационных технологий, которые повышают эффективность планирования ресурсов, использования памяти и обмена по сети. Это следующие инновации:

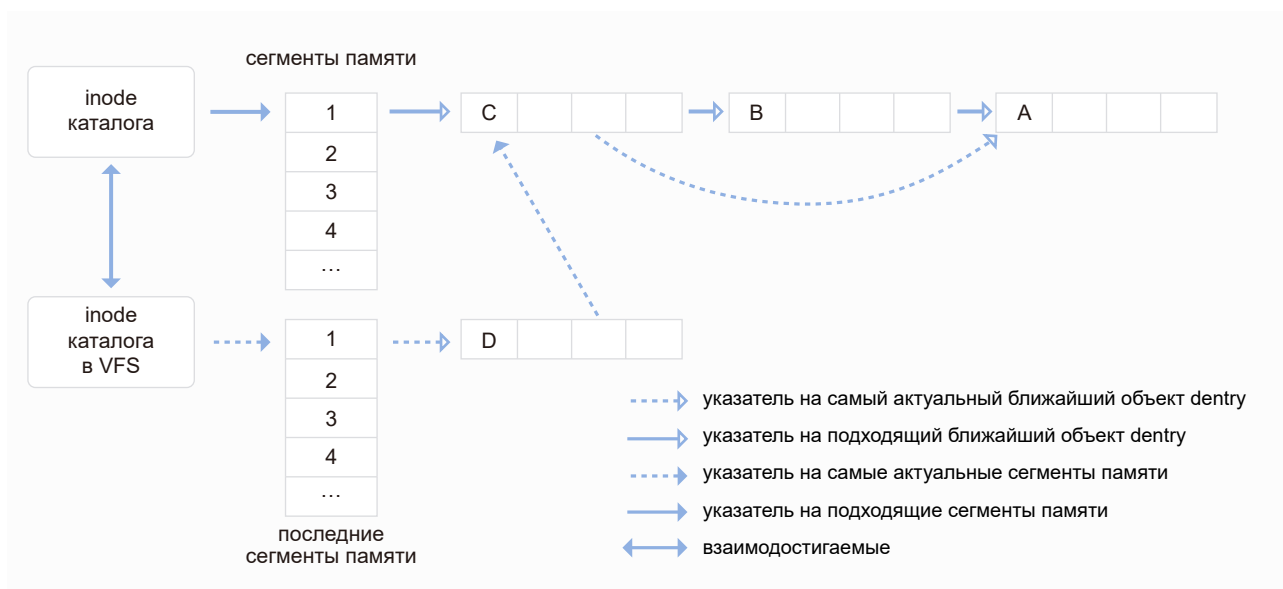
- 01/ Оптимизация планирования процессов:**
Оптимизированный алгоритм балансировки нагрузки уменьшает неэффективное потребление ресурсов во время выравнивания нагрузки между процессами и повышает производительность работы приложений.
- 02/ Динамическое приоритетное выделение ресурсов ядра:**
Благодаря новой опции загрузки **preempt=none/voluntary/full** ядро динамически изменяет режим приоритетного выделения ресурсов.
- 03/ Оптимизация производительности mremap:**
Записи, сделанные в каталогах страниц среднего уровня (PMD) и верхнего уровня (PUD), можно переместить, чтобы быстро отобразить большие блоки памяти.
- 04/ per memcg lru lock:**
Функция устраняет конфликты при блокировках, возникающие между экземплярами контейнеров cloud native, повышая производительность системы.
- 05/ Оптимизация управления памятью огромных страниц:**
В буфере огромных страниц памяти HugeTLB освобождаются хвостовые страницы в режиме совместного отображения и, таким образом, уменьшается объем ресурсов, которые занимают служебными операциями, связанными с управлением памятью огромных страниц.
- 06/ Параллельные обновления для буфера ассоциативной трансляции (TLB):**
Локальный и удаленный буферы TLB (Translation Lookaside Buffer) можно обновлять одновременно, оптимизируя, тем самым, процесс сброса буфера (TLB shootdown), ускоряя обновление TLB и повышая производительность сервисов.
- 07/ Оптимизация производительности огромных страниц vmalloc:**
Во время обращений **vmalloc()**, направленных на выделение пространства памяти, превышающего минимальный размер буфера огромных страниц, для отображения памяти используется не базовая страница, а огромная страница, что повышает эффективность использования TLB и уменьшает частоту промахов попадания в буфер TLB.
- 08/ Освобождение ресурсов памяти для предотвращения OOM:**
Для групп процессов с низким приоритетом в ситуации с нехваткой памяти (OOM) становится эффективной мера освобождение ресурсов, которая обеспечивает нормальную работу онлайн-сервисов.
- 09/ Коды аутентификации указателя (Pointer Authentication Code; PAC):**
Значение регистра перед его использованием в качестве указателя для доступа к данным или коду проходит аутентификацию, чтобы защититься от атак возвратно-ориентированного программирования (Return-Oriented Programming; ROP) и переходо-ориентированного программирования (Jump-Oriented Programming; JOP).
- 10/ Идентификаторы целевых ветвей (Branch Target Identifier; BTI):**
Данные идентификаторы предотвращают выполнение команд, которые не предназначены для целевой ветви. Сочетание BTI и PAC уменьшает число атак на поток управляющих команд.
- 11/ eXpress Data Path (XDP):**
Это высокопроизводительный программируемый сетевой путь к данным на основе eBPF, который обрабатывает данные до того, как сетевые пакеты попадают в стек сетевых протоколов, что повышает производительность сети. XDP работает в различных сценариях, таких как анти-DDoS, межсетевой экран и контроль качества обслуживания сети (QoS).
- 12/ Общая виртуальная адресация памяти (Shared Virtual Addressing; SVA):**
С помощью данного метода процессы и устройства хостов используют один виртуальный адрес, что позволяет избежать необходимости репликации ресурсов между узлами и устройствами. SVA повышает производительность обмена сервисными данными между хостами и устройствами.

Файловая система для новых носителей

Модули памяти NVDIMM, такие как Intel Optane, представляют собой новые высокоскоростные модули хранения данных, которые обеспечивают доступ на уровне байтов. Популярная файловая система ядра EXT4 может работать с функцией Direct Access (DAX) для повышения производительности чтения и записи данных NVDIMM. Однако с точки зрения управления метаданными EXT4 имеет недостаток — низкая производительность механизма синхронизации журналов. Это приводит к высокому расходу ресурсов, которые идут на управление метаданными и увеличение объема записи. Эта общая проблема не позволяет полностью использовать преимущества NVDIMM.

Новая файловая система EulerFS имеет технологию «мягкого» обновления метаданных. Ее механизм расчета dual-view и каталоги с указателями уменьшают расход ресурсов памяти на синхронизацию метаданных, что эффективно повышает производительность вызова функций файловой системы (create, unlink, mkdir, rmdir). EulerFS отличается значительно меньшей задержкой и более высокой пропускной способностью по сравнению с альтернативными вариантами EXT4 и DAX.

Описание функций



- **Каталог хеш-таблиц:** хеш-таблицы используются для управления записями каталогов, повышения эффективности линейного поиска и сокращения псевдо-обмена.
- **Унифицированный распределитель ресурсов:** устраняет барьеры между различными структурами данных и облегчает управление памятью.
- **Мягкое обновление:** облегченная технология, упрощающая реализацию функций и обеспечивающая целостность файловой системы.
- **Механизм расчета dual-view и каталоги с указателями:** этот механизм уменьшает объем ресурсов, потребляемых на синхронизацию метаданных, и эффективно повышает производительность чтения и записи файловой системы.
- **Трассировка зависимостей:** операции с записями каталога (создание, удаление и т.д.) автоматически не сохраняются. После завершения такой операции в inodes отслеживаются только сохраненные зависимости. Эти операции становятся сохраняемыми в асинхронном режиме, чтобы уменьшить задержку выполнения операций с метаданными.

Сценарии применения

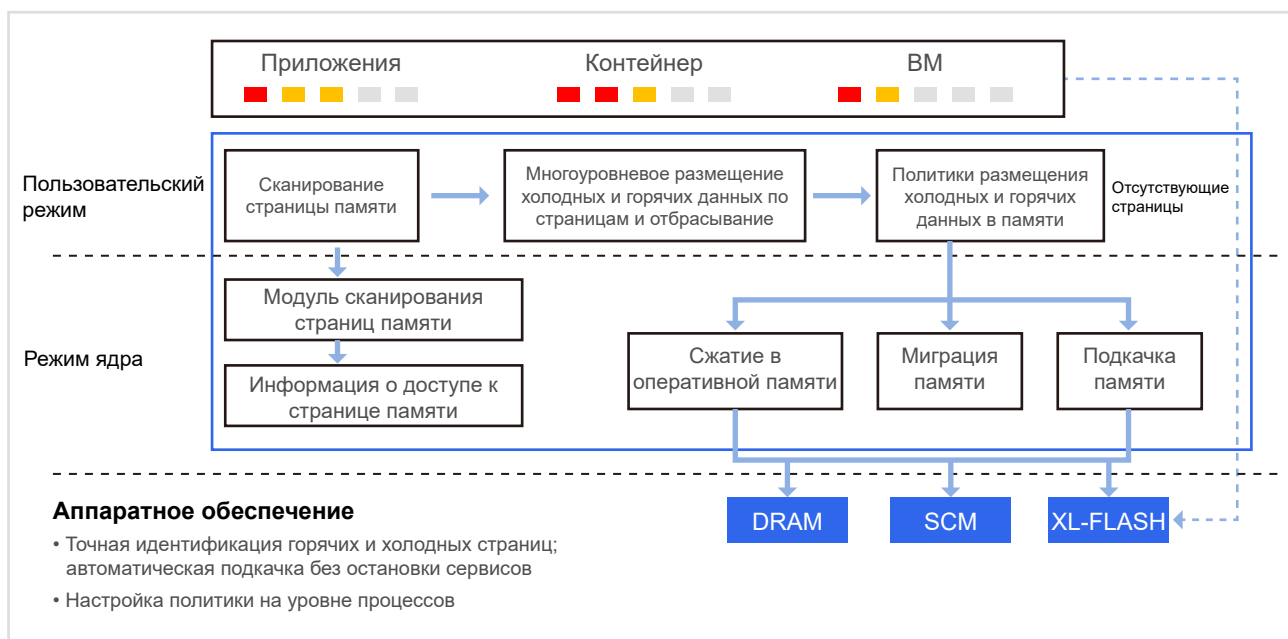
EulerFS идеально подходит для дополнения новых модулей памяти, таких как NVDIMM. Это эффективная альтернатива EXT4 и XFS в одноузловых приложениях и облачных распределенных приложениях.

Расширение памяти с многоуровневой организацией

В развитии современных процессов изготовления модулей памяти сейчас наблюдается стагнация. В то время как зрелая экосистема ARM способствует сокращению затрат на ядро процессора, такие новые технологии, как виртуальные машины, большие данные, искусственный интеллект и глубокое обучение, требуют более высоких вычислительных мощностей и больше ресурсов памяти. Ограниченная в обоих аспектах система становится огромной проблемой для будущего роста.

Одним из решений является etMem для расширения памяти с многоуровневой организацией. Модуль DRAM и такие низкоскоростные модули памяти, как Storage Class Memory (SCM), Apache Pass (AEP) и Remote Direct Memory Access (RDMA), образуют многоуровневую структуру памяти. Механизм автоматического планирования ресурсов памяти перенаправляет горячие данные в зону высокоскоростного доступа к памяти DRAM, а холодные данные — в зону низкоскоростного доступа. Многоуровневая структура увеличивает емкость памяти и обеспечивает эффективную и стабильную работу основных сервисов. etMem идеально подходит для приложений, которые занимают большой объем памяти, но не требуют частого доступа к памяти. Тесты показали, что etMem обеспечивает производительность для MySQL на 40% выше, чем аналоги. По запросам пользователей во фреймворк пользовательской памяти добавлен механизм подкачки памяти в пользовательском режиме.

Описание функций



В версии openEuler 21.09 унаследованы следующие возможности предыдущей версии:

- **Управление на уровне процесса.** Конфигурационный файл etMem можно использовать для расширения памяти. По сравнению с нативным механизмом kswap на базе LRU, etMem гибче и точнее.
- **Многоуровневое размещение холодных и горячих данных.** В пользовательском режиме для определенного процесса можно выполнить сканирование для доступа к памяти. При настройке политики поуровневого размещения данные классифицируются в зависимости от частоты доступа к ним — холодные (менее востребованные) данные помещаются в область памяти с высокой скоростью обработки, а горячие (более востребованные) данные — в область памяти с низкой скоростью обработки, но большой емкостью.
- **Политика отбрасывания.** Холодные данные отбрасываются, если отвечают условиям, указанным в конфигурационном файле etMem и конфигурации системной среды. Процесс отбрасывания использует встроенные функционалы ядра, которые являются безопасными и надежными и не влияют на пользовательский опыт.

В openEuler 21.09 добавлена следующая новая функция:

Подкачка памяти в пользовательском режиме. Хранение отброшенных холодных данных переключается на пользовательскую память в соответствии с заданной предустановленной политикой etMem. Подкачка памяти в пользовательском режиме обеспечивает более высокую производительность, чем подкачка памяти в режиме ядра, и весь процесс подкачки прозрачен для пользователей.

Сценарии применения

Расширение памяти с многоуровневой организацией доступно для работающих на узле сервисных процессов. etMem подходит для приложений, которые занимают большой объем памяти, но доступ требуется нечасто, например MySQL, Redis и Nginx. Все операции с расширением памяти выполняются внутри узла, между узлами операции не выполняются.

В сценарии хранения в пользовательском режиме функция userswap может сделать пользовательскую память устройством подкачки.

05/

Облачная база



Расширенное планирование Cloud-Native

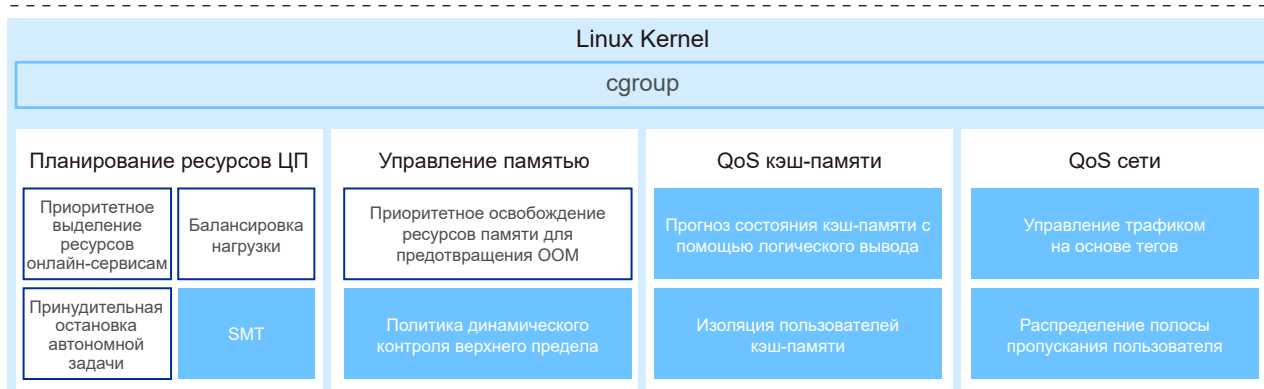
В облаке интерактивные онлайн-сервисы чувствительны к задержкам, и любое воздействие может иметь эффект колебаний нагрузки. В результате средняя загрузка центрального процессора низкая ($\approx 15\%$). Многие предприятия прибегают к гибриднему развертыванию онлайн- и офлайн-сервисов, поскольку этот способ повышает эффективность использования ресурсов. Однако обычного механизма распределения ресурсов и управления ядром недостаточно, он не решает проблему колебаний производительности и QoS для онлайн-сервисов. openEuler имеет собственные алгоритмы планирования ресурсов процессора и высвобождения памяти, которые предназначены для гибридного развертывания облачных сервисов (cloud native). Эти инновационные алгоритмы повышают эффективность использования центрального процессора и обеспечивают QoS онлайн-сервисов на требуемом уровне.

Одним из алгоритмов планирования является планировщик с учетом информации о качестве (quality aware scheduler; QAS), который предназначен для сервисов cloud native в гибридных режимах развертывания. QAS оперативно выделяет ресурсы ЦП под приоритетные онлайн-задачи, планирует задачи, используя детерминированный подход, и принудительно останавливает автономно выполняемые задачи, мешающие выполнению приоритетных задач.

Разработчики openEuler оптимизировали алгоритм высвобождения памяти, устраняющий проблему нехватки памяти (OOM). Для групп процессов с низким приоритетом в ситуации с нехваткой памяти (OOM) становится эффективной мера освобождения ресурсов, которая обеспечивает нормальную работу онлайн-сервисов.

Описание функций

Гибридное развертывание контейнеров	Гибридное развертывание процессов и контейнеров	Гибридное развертывание контейнеров и VM	Гибридное развертывание VM
-------------------------------------	---	--	----------------------------



- **Параметры свойств процесса:** настроить свойства выполнения автономной и онлайн-задач можно, вызвав интерфейс cgroup.
- **Приоритетные задачи:** при одновременной обработке автономных и онлайн-задач процессор приоритетно обрабатывает последние, которые занимают ресурсы на несколько микросекунд раньше автономных процессов.
- **Механизм Kill для автономных задач:** автономные задачи быстро реагируют на полученный сигнал о принудительной остановке и завершают выполнение в короткий срок.
- **Освобождение ресурсов памяти для предотвращения OOM.** Если возникает нехватка памяти, ресурсы памяти, занятые процессами с низким приоритетом, освобождаются. Это обеспечивает нормальную работу онлайн-сервисов.

Сценарии применения

Гибридное развертывание интерактивных сервисов, чувствительных к задержке (MySQL, Redis и Nginx), и сервисов, нечувствительных к задержке (автономные процессы машинного обучения). Сюда включаются гибридные развертывания: контейнер-контейнер, контейнер-процесс, контейнер-виртуальная машина, а также виртуальная машина-виртуальная машина.

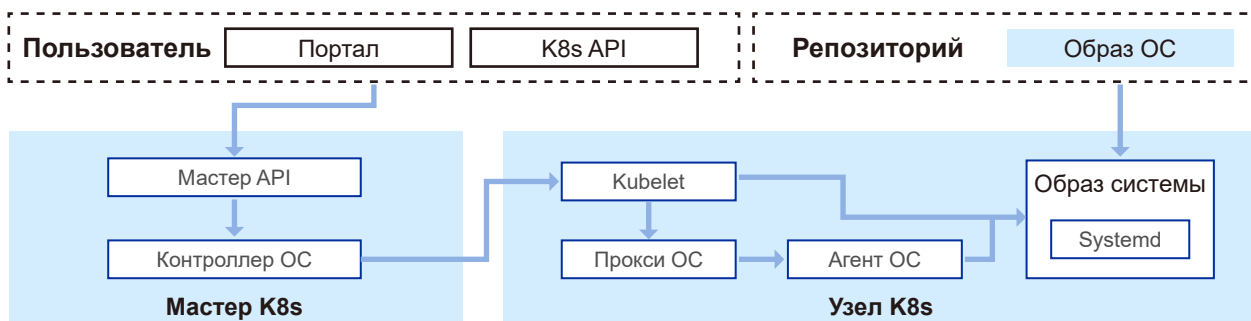
Контейнерная ОС

Следующим шагом в эволюции облачных вычислений является cloud-native. В основе большинства современных программных инфраструктур cloud-native лежит Kubernetes. Свои операционные системы для сценариев cloud-native уже запустили некоторые известные производители ОС: Red Hat Enterprise Linux CoreOS (RHCOS) и AWS Bottlerocket. Эксплуатация операционных систем, которые разворачиваются и управляются в контейнерах, осуществляется аналогично контейнерам.

Чтобы подстроиться под эту тенденцию, сообщество openEuler подготовило KubeOS — операционную систему, которая централизованно управляет кластерными ОС cloud-native в контейнерах. KubeOS обладает следующими особенностями:

- Контейнерная виртуализация на уровне ОС и взаимодействие Kubernetes для высокоточного управления жизненным циклом.
- Легкая «обрезка» ОС, которая уменьшает число ненужных пакетов, ускоряя процессы обновления и замены.

Описание функций



- **Контроллер ОС:** глобальный диспетчер отслеживает экземпляры ОС и собирает информацию о них на всех узлах. Компонент реализует глобальное управление жизненным циклом ОС, включая обновления, перезапуски и настройки конфигураций, а также исключение pod-объектов при завершении работы ОС.
- **Прокси ОС:** прокси узла размещается и работает на рабочем узле. Отслеживая экземпляр ОС, он собирает информацию о нем на одном узле, а затем передает ее в агент ОС.
- **Агент ОС:** получает от прокси ОС команды управления жизненным циклом контейнерной операционной системы, например с целью установки и обновления контейнерной ОС.

Сценарии применения

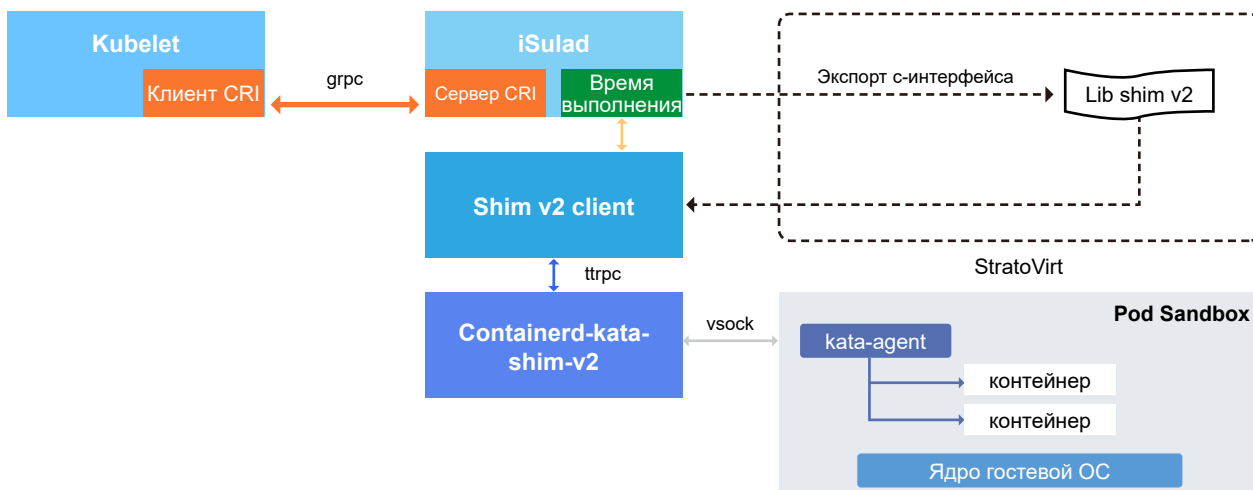
KubeOS предназначена для управления операционными системами хостов с контейнерными облачными сервисами, которые работают на Kubernetes. Эксплуатация и управление жизненным циклом осуществляются аналогично контейнерным сервисам.

Решение безопасного контейнера

Являясь основной технологией облачных вычислений, контейнеры упрощают инкапсуляцию, ускоряют развертывание и уменьшают зависимости среды. Контейнеры могут напрямую получать доступ к ресурсам хоста и совместно использовать ядро хоста. Однако это может стать причиной появления уязвимостей в системе безопасности, которые могут, например, допустить атаку выхода из контейнера (container escape). Таким образом, обычные контейнеры не отвечают большинству требований к изоляции с целью защиты многопользовательских финансовых приложений. Ведущие в этой области игроки Alibaba и Intel активно продвигают проекты с открытым исходным кодом Kata. Обе компании используют контейнеры на изолированном уровне виртуализации, формируя безопасные контейнерные решения. Компания Google также запустила песочницу GVisor для виртуализации на уровне процессов, защищая, таким образом, контейнерные приложения от рисков.

В отличие от этих компаний, разработчики сообщества openEuler, объединив платформу виртуализации StratoVirt и контейнерный движок iSulad, создали безопасное контейнерное решение. По сравнению с решением Docker+QEMU openEuler сокращает непроизводительное потребление памяти и время загрузки более чем на 40%. Платформа обеспечивает легкую и безопасную среду выполнения приложений, устраняя риски безопасности в контейнерных средах и средах ОС хостов.

Описание функций



Платформа виртуализации StratoVirt отличается следующими особенностями:

- **Высокая безопасность.** Обеспечивается безопасность на уровне языка программирования Rust. Его модульная структура сводит к минимуму поверхность атаки и физически изолирует каждого пользователя.
- **Облегченное решение с небольшими затратами ресурсов на служебные процессы.** При использовании упрощенной модели устройства можно запустить за 50 мс и контролировать потребление ресурсов памяти на служебные процессы, не допуская их превышения 4 МБ.
- **Конвергенция программного и аппаратного обеспечения.** Поддерживается виртуализация x86 и Kunpeng-V.
- **Молниеносное масштабирование.** Масштабирование устройства за считанные миллисекунды, гибкие возможности расширения ресурсов для облегченных нагрузок.
- **Поддержка нескольких сценариев.** Развертывание без сервера, безопасные контейнеры и стандартные виртуальные машины — все это доступно с использованием всего одной архитектуры.

Контейнерный движок iSulad включает в себя следующие преимущества:

- **Облегченный движок.** Для перекомпоновки облегченной базы контейнеров с целью адаптации к развертыванию в сценариях периферийных и облачных вычислений используется язык программирования C/C++.
- **Гибридное планирование.** Для взаимодействия с StratoVirt используется containerd-shim-kata-v2. Планирование и развертывание StratoVirt и Kubernetes осуществляется на единой платформе.

Сценарии применения

Решение безопасного контейнера заменит контейнеры Docker благодаря тому, что обеспечивает безопасность и изоляцию контейнерных сервисов. Решение подходит для многопользовательских приложений, для применения в финансовой сфере, телекоммуникационной отрасли и области безопасности.

Инструмент развертывания в двух плоскостях

eggo — проект по развертыванию и управлению кластером Kubernetes, инициированный openEuler SIG **sig-CloudNative**. Он обеспечивает эффективное и стабильное развертывание (в автономном и онлайн-режимах) одного кластера на нескольких архитектурах. В нем применяется методология GitOps, которая определяет изменения в конфигурации кластера и обеспечивает унифицированный и эффективный процесс развертывания кластерных ОС.

Описание функций



- **Управление кластером на основе версий:** для хранения и отслеживания изменений конфигурации кластера используются репозитории Git.
- **Оповещение о конфигурации:** обнаружив изменения в конфигурации кластера в репозиториях Git, GitOps отправляет на движок развертывания запрос на операцию с кластером.
- **Движок развертывания:** выполняет задачи сервисного кластера. Например, он запускает такие задачи, как развертывание сервисного кластера, удаление сервисного кластера, добавление и удаление узлов.

Сценарии применения

eggo применяется для облачных инфраструктур с двумя плоскостями на базе архитектур x86 и ARM. Поскольку eggo работает на базе фреймворка cloud native (Kubernetes), инструмент реализует унифицированное двухплоскостное развертывание кластеров, мониторинг и аудит работы операционных систем.



06/

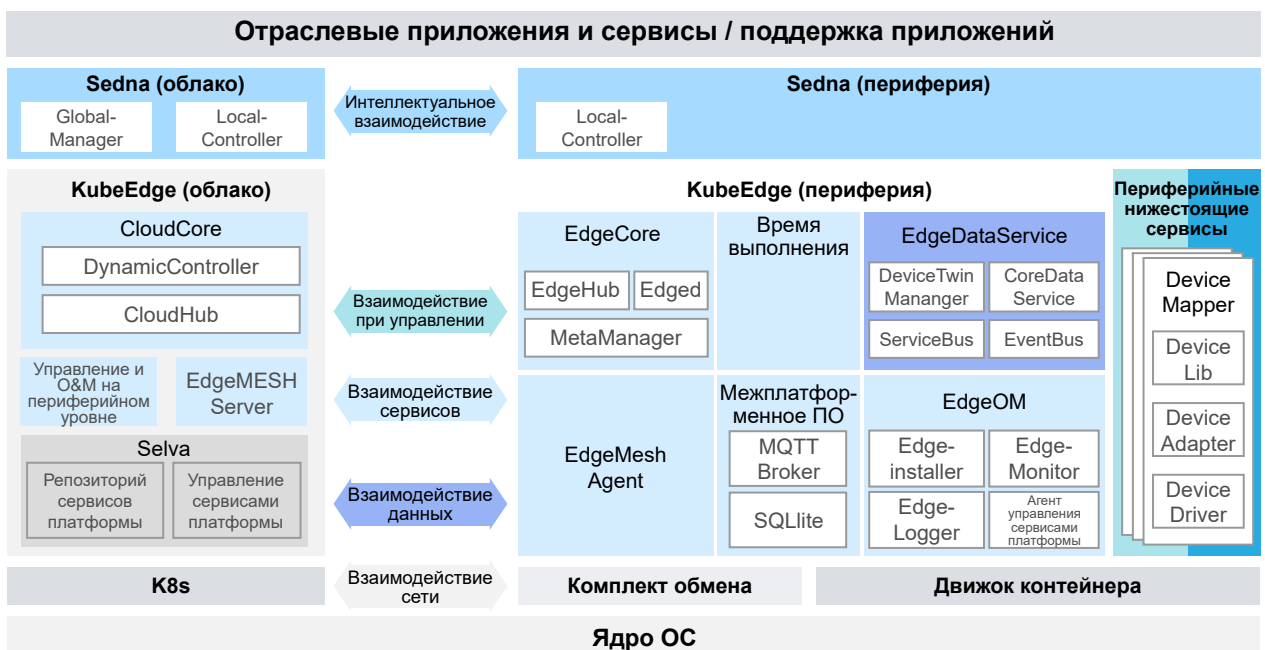
**Инновационные разработки
для конкретных сценариев**

Периферийные вычисления

Периферийные вычисления считаются одним из 10 главных технологических трендов, эта область требует инновационных решений, которые справятся с обработкой огромных объемов данных, генерируемых инфраструктурой умного города, приложениями автономного вождения и сервисами промышленного интернета. По прогнозам IDC к 2025 году в Китае будет генерироваться до 48,6 ЗБ данных, а централизованные платформы облачных вычислений уже не смогут обеспечить необходимую пропускную способность, удерживать на стабильном уровне сетевые задержки и затраты на управление данными. Все свидетельствует о том, что периферийные вычисления крайне важны для многих отраслей.

В версии openEuler 21.09 Edge интегрирован фреймворк KubeEdge+, который обеспечивает унифицированное управление и выделение ресурсов для работы периферийных приложений. В версии улучшено интеллектуальное взаимодействие периферийных и облачных систем, что способствует оптимизации процесса развертывания приложений на базе искусственного интеллекта, реализации функционалов обнаружения взаимодействия облако-периферия и пересылки трафика, а также улучшению работы функционалов в южном направлении (southbound).

Описание функций



openEuler 21.09 Edge отличается следующими особенностями:

Унифицированный фреймворк KubeEdge+, в котором эффективно сочетаются возможности периферии и облака. Это означает, что он предоставляет не только базовые возможности, такие как совместное развертывание приложений для периферии и облака и управление ими, но также и реализует высококачественное взаимодействие между периферийными и облачными устройствами и управляет нижестоящими периферийными устройствами.

Хотя версия 21.09 подходит для большинства сценариев развертывания, последующие версии будут включать функции:

- **Взаимодействие облако-периферия.** На границе развертывается EdgeMesh Agent, в облаке — EdgeMesh Server. Эти компоненты выполняют интеллектуальное обнаружение и маршрутизацию запросов на обслуживание.
- **Оптимизация нижестоящей периферии.** Для доступа в южном направлении используется Device Mapper, предоставляющий периферийные механизмы профилирования и синтаксического анализа. Компонент контролирует периферийные устройства и сервисные потоки в южном направлении и управляет ими. Совместим с экосистемой EdgeX Foundry с открытым исходным кодом.

- **Службы по работе с данными на периферии.** Эти службы сохраняют данные по запросу в сообщениях, на носителях или в других потоках, а также выполняют анализ и экспорт данных.
- **Интеллектуальная архитектура Sedna.** Фреймворк Sedna с открытым исходным кодом предоставляет базовые функционалы логического вывода на периферии и в облаке, возможности федеративного и инкрементного обучения. Фреймворк упрощает управление моделями и наборами данных, благодаря чему разработчики могут быстро разрабатывать и развертывать приложения ИИ на периферии, а также повышает эффективность обучения и развертывания.

Сценарии применения

Периферийные вычисления можно использовать для более тесной связи периферии и облака в широком спектре сценариев: интеллектуальное производство, городской транспорт, инспекция на платных дорогах, интеллектуальные АЗС, распознавание медицинских снимков и интеллектуальные кампусы.

Встраиваемые системы

Версия openEuler 21.09 Embedded, созданная для встраиваемых систем, выделяется своей облегченной архитектурой. В данной ОС запускаются безопасные и облегченные контейнеры, которые поддерживают архитектуру микросхем ARM32 и ARM64. Решение предназначено для создания встроенных ОС и выстраивания бесперебойной коммуникации между партнерами по экосистеме, пользователями и разработчиками сообщества openEuler. Кроме того, версия улучшает совместимость архитектур микросхем (PowerPC и RISC-V) и связанных с ними возможностей (детерминированная задержка, промышленное межплатформенное программное обеспечение и системы моделирования).

Описание функций



openEuler 21.09 Embedded отличается следующими особенностями:

- **Облегченная архитектура.** Фреймворк компиляции Yocto позволяет отдельно настраивать системные элементы, формируя сжатые, облегченные модели. Например, образ ОС можно сжать до размера менее 5 МБ, сократить потребление ресурсов памяти на служебные задачи до менее чем 15 МБ, а время запуска до менее чем 5 секунд.
- **Усиление безопасности.** Чтобы снизить риск появления уязвимостей, усиливается защита паролей учетных записей и прав доступа к файлам. Функция безопасности ОС включена по умолчанию.
- **Облегченные контейнеры.** Облегченная среда выполнения контейнера для встроенных сценариев поддерживает стандартные образы контейнеров.
- **Поддержка разных архитектур.** Поддерживаются архитектуры микросхем ARM32 и ARM64.

Хотя версия 21.09 удобна для большинства сценариев развертывания, в последующие версии будут добавлены следующие функции:

- **Детерминированная задержка.** Эта функция обработки работает на многоядерной архитектуре и поддерживает возможность выбора уровня задержки в зависимости от области промышленного применения.
- **Отраслевые сертификаты безопасности.** Планируется, что последующие версии будут проходить сертификацию безопасности IEC61508 и EC62443.

Сценарии применения

Встроенные программные решения способны повысить производительность в различных областях — аэрокосмической промышленности, управлении промышленными объектами, телекоммуникациях, автомобилестроении и здравоохранении. Благодаря усовершенствованным технологиям 5G и ИИ, встроенные системы можно применять для Интернета вещей и в периферийных вычислительных устройствах.

Распределенная память (выпуск планируется в ближайшее время)

Необходимость своевременной обработки больших объемов данных приводит к увеличению масштаба приложений. Однако крупномасштабные распределенные кластерные приложения сложны, в работе часто возникают проблемы с производительностью, причиной которых являются узкие места в существующей вычислительной архитектуре. Модуль распределенной памяти — это платформа ускорения работы распределенных кластеров, предназначенная для приложений. Используя технологии обработки данных в памяти и «рядом с данными», модуль повышает производительность приложений в таких сценариях распределенных приложений, как аналитика больших данных, высокопроизводительные вычисления, системы искусственного интеллекта, распределенное хранилище, базы данных, облака и виртуализация. Этот модуль также ускоряет работу приложений, которым требуются вычислительные ресурсы разного уровня, и обеспечивает тесное эффективное взаимодействие устройств, периферии и облака.



07/

Расширенные функции

A-OPS: интеллектуальные средства эксплуатации и техобслуживания

Зрелость и высокий уровень развития технологий обработки больших данных и машинного обучения приводят к генерированию огромных объемов данных, которые ежегодно увеличиваются в 2–3 раза. Эти изменения стимулируют инвестиции в новые, эффективные и интеллектуальные системы O&M, способные сократить расходы предприятий. В Red Hat Insights используются интеллектуальные управляемые данными движки, которые выполняют автоматический прогноз, диагностику и локализацию неисправностей, помогая быстро решать проблемы, связанные с безопасностью и производительностью ОС, от нескольких часов до нескольких минут.



A-OPS в ОС openEuler представляет собой базовый интеллектуальный фреймворк эксплуатации и техобслуживания, который обеспечивает работу таких профессиональных функционалов, как трассировка конфигурации, информированность о топологии приложений и локализация неисправностей. Фреймворк гарантирует быстрое устранение проблем и снижает затраты на эксплуатацию и техобслуживание.

Описание функций

- **Информированность о топологии приложений.** Это основанный на eBPF фреймворк зонда, который, не нагружая систему, выполняет автоматическое распознавание топологии сети и определение состояния на уровне приложений. Фреймворк визуализирует состояние сети через веб-интерфейс пользователя, помогая обслуживающему персоналу быстро обнаруживать проблемы в сети.
- **Трассировка конфигурации.** Изменения конфигурации происходят часто и могут стать причиной проблем во время обслуживания. A-OPS позволяет настраивать кластерную ОС, управлять ею, указывать домен. Здесь реализованы функции согласования базового состояния конфигурации и сравнения, которые помогают быстро устранить проблемы конфигурации.
- **Обнаружение сбоев.** A-OPS предоставляет экспертный движок для обнаружения системных сбоев в режиме реального времени и устранения их, который сокращает время простоя системы и затраты на эксплуатацию и техобслуживание.

Сценарии применения

A-OPS станет идеальным помощником для любой команды технического обслуживания ОС. Использование существующего в A-OPS дерева ошибок или нового добавленного дерева ошибок способствует эффективному обслуживанию и сокращает время простоев.

secPaver: инструмент настройки политики безопасности

secPaver — это инструмент разработки политик SELinux, который поможет разработчикам создавать политики безопасности приложений. Абстрагируя обычные конфигурационные файлы политик, secPaver инкапсулирует унифицированные интерфейсы для работы с политиками, что в результате позволяет скрыть детали механизма от разработчиков политик, упростив процесс разработки и повысив его эффективность. Инструмент можно использовать для разработки и упрощения политик SELinux, а также можно применять с другими механизмами безопасности, например с AppArmor.

Описание функций

Функция secPaver: комплексный инструмент разработки политик



В secPaver предусмотрен ряд функций для системных политик:

- **Проектирование.** Для создания проектов и изменения конфигурационных файлов политик используется следующая команда:
pav project create *project_name*/opt/policy
- **Генерация.** Для генерации файла политики SELinux на основе конфигурационного файла политики используется следующая команда:
pav project build --engine selinux -d/opt/policy/*project_name*
- **Тестирование.** Для установки, запроса и удаления политик используются следующие команды:
pav policy list
pav policy install/uninstall *policy_name*
- **Выпуск.** Для выпуска пакетов политик (содержащих файлы политик и файлы со скриптами установки и удаления) используется следующая команда:
pav policy export *policy_name*

Сценарии применения

secPaver предназначен для разработчиков и других специалистов, занимающихся конфигурированием, разработкой файлов политики SELinux для приложений.

Kunpeng GCC

Kunpeng GCC разработан на основе набора компиляторов GCC 10.3 с открытым исходным кодом. Это высокопроизводительный компилятор для процессора Kunpeng 920, обеспечивающий программно-аппаратное взаимодействие, оптимизацию памяти, SVE и математическую библиотеку.

- Данный компонент полностью использует аппаратные возможности процессоров Kunpeng, повышая операционную эффективность. В эталонных тестах, например SPEC CPU 2017, Kunpeng GCC показал производительность гораздо выше, чем компилятор GCC 10.3, разработанный в исходном сообществе.
- Более того, он поддерживает флаги `mmodel=medium` и `fp-model`, четверную точность представления чисел с плавающей запятой и векторизованную математическую библиотеку.

Описание функций

- **Адресация по флагу `mmodel=medium`:** обеспечивает корректный доступ к данным (>4 ГБ), устраняя ошибку, вызванную переполнением буфера в архитектуре ARM.
- **Четверная точность:** эффективное повышение точности 128-битных арифметических операций с плавающей запятой по сравнению с архитектурой ARM.

- **Векторизованная математическая библиотека:** автоматический поиск доступной векторизованной математической библиотеки на этапе векторизации.
- **Оптимизация SVE:** значительное повышение производительности выполнения программы на ARM-машинах, поддерживающих инструкции SVE.
- **Оптимизация векторизации SLP:** анализ и векторизация оптимизирующих групп цепей с целью повышения производительности программы.
- **Оптимизация структуры памяти:** элементы структуры переупорядочиваются таким образом, чтобы элементы, доступ к которым осуществляется часто, размещались в непрерывном пространстве памяти. Это увеличивает коэффициент попадания в кэш и производительность работы программы.
- **Контроль точности по флагу fp-model:** контроль и повышение точности операций с плавающей запятой.

Сценарии применения

В HPC-тестировании приложений прогнозирования погоды, а также приложений по микросистемной технике и материаловедению Kunpeng GCC показывает производительность на 10% выше, чем компилятор GCC 10.3, разработанный исходным сообществом. В эталонном тестировании SPEC CPU 2017 компилятор Kunpeng GCC демонстрирует производительность на 15% выше, чем GCC 10.3.

BiSheng JDK

BiSheng JDK — это расширенный комплект разработки приложений на языке Java, созданный на основе Open Java Development Kit (OpenJDK). Комплект отличается высокими показателями производительности и доступности, оптимизирует производственные среды в любых областях и отраслях промышленности, а также улучшает производительность, особенно в сценариях с ARM-архитектурой. BiSheng JDK поддерживает OpenJDK версий 8 и 11 и совместим со стандартной версией платформы Java (Java SE). BiSheng JDK обладает следующими достоинствами:

- **Стабильность и эффективность.** В эталонных тестах, в частности SPECjbb 2015, BiSheng JDK показывает намного выше производительность, чем OpenJDK.
- **Конвергенция программного и аппаратного обеспечения.** Комплект полностью использует аппаратные возможности серверов Kunpeng, которые повышают эффективность.
- **Высокий уровень безопасности.** BiSheng JDK синхронизирует обновления с версиями, выпускаемыми сообществом OpenJDK, проводит строгий анализ и контроль, а также при необходимости применяет патчи в базе данных уязвимостей CVE.
- **Открытый исходный код.** Исходные сообщества получают новые версии BiSheng JDK, исходные коды и другие результаты.

Описание функций

- **Java Flight Recorder (JFR).** JFR — это онлайн-инструмент, используемый для сбора данных диагностики и профиля, который помогает свести к минимуму неэффективное использование производительности. Для приложений, работающих в течение длительного времени, влияние на производительность составляет менее 1%. Чтобы включить JFR в производственной среде, выполните команду `-XX:+FlightRecorder`. Для анализа генерируется файл дампа. Работая с Java Mission Control (JMC), JFR улучшает визуализацию анализа производительности.
- **AppCDS (только в версии BiSheng JDK 8).** Компонент AppCDS является расширенной версией Class-Data Sharing (CDS). Служит для выгрузки классов приложений в файлы JSA с целью расширения области использования классов и повышения скорости запуска и загрузки приложений.
- **Сборщик мусора Garbage-First (G1 GC) (только в версии BiSheng JDK 8).** Виртуальная машина Java Virtual Machine (JVM) обнаруживает снижение нагрузки приложений и незанятое пространство кучи Java, автоматически уменьшает размер выделяемой кучи Java и возвращает свободные ресурсы памяти в ОС. Данный сборщик мусора в виртуальной машине помогает снизить непроизводительный расход ресурсов в сценарии контейнера, когда оплата за ресурсы идет на основе использованного объема.

- **Сборщик мусора Z (ZGC) (только в версии BiSheng JDK 11).** ZGC — это масштабируемый сборщик мусора с низкой задержкой и максимальным временем паузы 10 миллисекунд, которое не меняется даже при изменении размера кучи. Совместим с ARM64.
- **NUMA-Aware для сборщика мусора G1 GC (версии BiSheng JDK 8 и BiSheng JDK 11).** Виртуальная машина Java может полностью использовать аппаратные возможности. Во время работы приложения для размещения объектов в памяти преимущественно используется память локального узла. Во время сбора мусора репликация памяти выполняется преимущественно на одном и том же узле. Это гарантирует максимальное сходство (affinity) данных приложений после окончания сбора мусора.
- **Новый быстрый компонент сериализации (в версиях BiSheng JDK 8 и BiSheng JDK 11).** Компонент сокращает избыточное количество новых передач и повышает эффективность процессов сериализации и десериализации.
- **KAЕ Provider (только в версии BiSheng JDK 8).** Используя данный механизм, BiSheng JDK 8 поддерживает шифрование и дешифрование KAE в серверах Kunpeng, выгодно оптимизируя данные операции на серверах Kunpeng AArch64. KAE Provider на 90% увеличивает производительность шифрования и дешифрования по сравнению с механизмом самого JDK.
- **Параллельное сканирование jmap (в версиях BiSheng JDK 8 и BiSheng JDK 11).** По умолчанию jmap OpenJDK выполняет операцию дампа Java, используя один поток. Однако jmap в версиях BiSheng JDK 8 и BiSheng JDK 11 реализует параллельное и инкрементное сканирование, эффективно повышая скорость сканирования в сценариях с большими размерами кучи. В настоящее время эта возможность сканирования доступна в сборщике мусора G1 GC, ParallelGC и CMS.

Сценарии применения

Сценарий применения 1. Приложение для обработки больших данных.

BiSheng JDK оптимизирует процессы выделения и освобождения ресурсов памяти для сборщика мусора в приложениях, обрабатывающих большие данные, и устраняет избыточные барьеры памяти в коде JIT. В соответствующем эталонном тестировании BiSheng JDK продемонстрировал уровень производительности выше, чем у OpenJDK, на 5%–20%.

Сценарий применения 2. Общие Java-приложения.

BiSheng JDK оптимизирует слабую модель памяти сервера Kunpeng, что позволяет избежать непродуктивных барьеров памяти. Программно-аппаратный компонент NUMA-Aware повышает эффективность доступа к памяти приложений и позволяет полностью использовать производительность приложений. В BiSheng JDK усовершенствованы такие инструменты Java, как JFR и jmap, используя которые разработчики смогут быстро профилировать производительность и локализовывать проблемы.

08/

**Управление
сообществом**



I Видение

Создание инновационной платформы путем сотрудничества в рамках сообщества openEuler. Организация единого и открытого сообщества разработчиков операционной системы openEuler с целью продвижения многопроцессорной архитектуры. Формирование процветающей программно-аппаратной экосистемы.

I Связь с сообществом

Сообщество openEuler ведет множество проектов, для реализации которых организуются группы. Каналы связи с данными группами, в том числе адреса ящиков рассылок, можно найти в соответствующих файлах README.

Список рассылок

Обсуждение открытой темы можно начать, отправив письмо по адресу ящиков рассылок.

Адреса ящиков рассылок сообщества приведены на странице <https://openeuler.org/en/community/mailling-list/>.

Далее описываются два способа подписки на рассылку: через веб-страницу или по электронной почте.

Веб-страница

1. Щелкните имя в списке рассылок на странице <https://openeuler.org/en/community/mailling-list/>, чтобы перейти на страницу подписки.
2. Введите адрес электронной почты и нажмите **Subscribe**.
3. Войдите в почтовый ящик и ответьте на письмо-подтверждение, отправленное с ресурса openeuler.org.

После этого вы получите с адреса @openeuler.org второе электронное письмо, содержащее слова приветствия «Welcome», означающие, что вы успешно подписались.

Примечание: Если вы не получили электронное письмо, содержащее слова приветствия «Welcome», значит, была неверно указана тема электронного письма-ответа. В этом случае снова ответьте на оригинальное письмо, указав его исходную тему.

Электронная почта

1. Отправьте письмо с заголовком «subscribe» на адрес подписки, отображаемый рядом с именем проекта в списке адресов рассылок. Адрес подписки должен быть сформирован на основе адреса рассылки с добавлением суффикса «-join» (см. пример ниже).
2. Ответьте на письмо-подтверждение, отправленное с ресурса openeuler.org.
Формирование адреса электронного письма-подписки на примере имени Dev (dev@openeuler.org):
Кому: dev-join@openeuler.org
Тема: Subscribe
Тело письма: --

В настоящее время по электронной почте обсуждаются две темы: «announce» и «discussion». Письмо с темой «discussion» отправляется по адресам из списка рассылок таким же способом, как и по частным адресам.

Рекомендуется, хотя и не обязательно, добавить к теме электронного письма пару квадратных скобок и тему в качестве префикса. Электронные письма с темой «announce» используются только для рассылки объявлений и предупреждений и не предназначены для отправки вопросов.

Примечание: Если сообщения в папке «Входящие» вашего ящика не появляются, проверьте папку «Спам».

Чтобы отписаться от рассылки, выполните следующие действия:

1. Отправьте письмо с темой «unsubscribe» по адресу, сообщения с которого вы не хотите больше получать. Как правило, здесь добавляется суффикс «-leave» (например: dev-leave@openeuler.org).
2. Проверьте информацию, содержащуюся в электронном письме-подтверждении, и ответьте на него.

Отмена подписки произойдет после отправки ответа на письмо-подтверждение.

Получение помощи

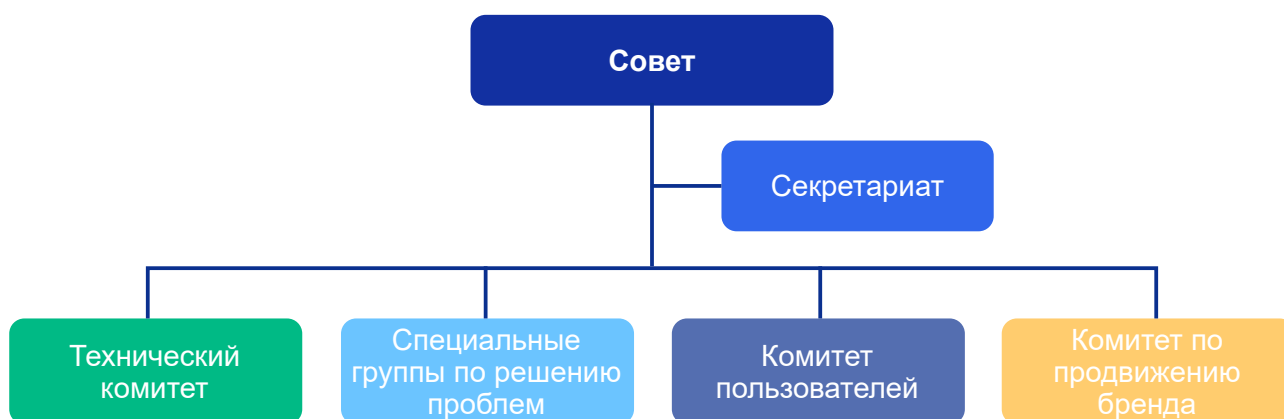
Если у вас возникли проблемы при отправке электронного письма, свяжитесь со специалистами из группы поддержки:

Эл. почта: infra@openeuler.org

Если вы нашли какие-либо ошибки, связанные с рассылкой, отправьте вопрос специалистам группы поддержки. Подробная информация об отправке вопросов приведена далее.

I Структура руководства сообществом

Руководящий состав руководит процессам разработок, которые ведет сообщество openEuler. В структуру руководства входят Совет, Секретариат, Технический комитет, Комитет пользователей, Комитет по продвижению бренда и Специальные группы по решению проблем (Special Interest Group; SIG).



Совет формулирует долгосрочный план развития и вырабатывает отдельные и коллективные политики для сообщества. Совет проверяет работу Комитета пользователей и Комитета по продвижению бренда, управляет планами работы комитетов. Совет также отвечает за продвижение самого сообщества openEuler и соответствующих дистрибутивов операционной системы с целью их массового использования в различных отраслях по всему миру и строительства экосистемы.

Секретариат

Повседневная работа Секретариата под руководством Совета openEuler заключается в следующем:

1. Организация исполнения решений Совета сообществом, организация и проведение заседаний Совета, координация работы представителей сообщества.
2. Подготовка квартальных и ежегодных отчетов о работе сообщества, их публикация после утверждения советом директоров.

Технический комитет

Технический комитет openEuler осуществляет техническое руководство сообществом openEuler.

Основные обязанности комитета заключаются в следующем:

1. Принятие окончательного решения по техническим вопросам.
2. Окончательное утверждение видения и направления, в котором ведутся разработки сообществом.
3. Формулирование, координация и выполнение других обязанностей групп SIG сообщества. Решение конфликтов, возникающих в ходе сотрудничества между группами SIG и инструкторами, проверка и контроль повседневной работы SIG.
4. Выполнение повседневных задач сообщества по разработке и выпуск высококачественных версий ОС openEuler.

5. Внедрение инноваций в системную архитектуру и ядро, внедрение технологий виртуализации, cloud native и безопасности для поддержки продуктов сообщества конкурентоспособными.
6. Руководство сообществом в организации и реализации проектов разработки программного обеспечения с открытым исходным кодом, постоянная работа над укреплением технического авторитета сообщества.

Организация конференций

Официальная конференция (открытая): Технический комитет проводит открытое онлайн-обсуждение в 10:00 (GMT + 8) по средам раз в две недели.

Подробная информация о том, как присоединиться к конференции, публикуется в письме-рассылке за день до проведения.

Эта конференция открыта для разработчиков, которые заинтересованы в участии в сообществе openEuler.

Специальные группы по решению проблем

Специальная группа по решению проблем (Special Interest Group; SIG) представляет собой команду, организованную для определенной области деятельности, которая выполняет регулярные задачи и ведет свою деятельность для достижения поставленных целей. В группах SIG действуют прозрачные правила, и участники обязаны соблюдать кодекс поведения openEuler. Присоединиться к любой группе SIG и внести свой вклад в проекты группы может любой участник сообщества. Все группы SIG можно найти по адресу <https://openeuler.org/en/sig/sig-list/>.

Комитет пользователей

Комитет пользователей отвечает за связь с конечными пользователями сообщества openEuler и выполняет следующие обязанности:

- Сбор технических требований и требований к продуктам для подготовки релизов openEuler, выпуск версий ОС openEuler и организация других проектов разработки программного обеспечения с открытым исходным кодом сообщества openEuler. Передача отзывов и замечаний техническому комитету и совету, адаптация технологической дорожной карты под потребности конечных пользователей, улучшение рыночной атмосферы и долгосрочное планирование.
- Организация передовых методов работы в плане выпуска версий ОС и других проектов разработки ПО с открытым исходным кодом сообщества openEuler. Публикация информации о примерах из практики с разрешения конечных пользователей в сотрудничестве с Комитетом по продвижению бренда.

Комитет по продвижению бренда

Комитет отвечает за продвижение openEuler и связанных с сообществом брендов. Основные обязанности заключаются в следующем:

- Продвижение технологии ОС openEuler и повышение узнаваемости бренда openEuler.
- Проведение мероприятий, способствующих широкому распространению технологии ОС openEuler и созданию глобальной экосистемы.

I Структура руководства сообществом

Чтобы внести свой вклад в сообщество openEuler, в первую очередь следует выбрать интересующий проект из списка групп SIG или списка проектов сообщества openEuler. После подписки вы можете посещать совещания групп SIG или совещания по проектам и подписаться на рассылку. В каждой группе SIG и в каждом проекте обозначен круг требующих решения вопросов, над которыми вы можете работать.

Подписание CLA

Прежде чем приступить к участию в деятельности сообщества, необходимо подписать лицензионное соглашение участника (Contributor License Agreement; CLA).

Кодекс поведения сообщества

Сообщество openEuler соблюдает кодекс поведения, предусмотренный в *Конвенции участников V1.4 (Contributor Convention V1.4)*. Подробную информацию можно найти в документе *версии V1.4*.

Сообщения о неподобающем или недостойном поведении можно присылать Техническому комитету openEuler по адресу: tc@openeuler.org.

Обязательства участников

Сообщество openEuler старается поддерживать открытую и профессиональную среду, в которой недопустимы преследования, травля, агрессия или любые другие виды неподобающего поведения. К такому поведению относятся, помимо прочего, дискриминация по цвету кожи, расовой принадлежности, этническому признаку или национальному происхождению; дискриминация по возрасту; дискриминация по признаку пола, сексуальной ориентации и гендерной идентичности; бодишейминг; дискриминация людей с физическими и умственными недостатками; дискриминация по уровню опыта, образования или социального положения.

Наши принципы

Меры, способствующие созданию позитивной атмосферы, включают, в том числе:

- Дружелюбное общение, использование уважительной ко всем лексики.
- Уважительное отношение к различным точкам зрения и опыту.
- Открытость для критики и предложений.
- Действия с учетом интересов сообщества.

В сообществе openEuler запрещены преследования, неподобающее и любое другое неприемлемое поведение как в сети, так и при живом общении, включая, в том числе:

- Распространение комментариев, изображений и видео, которые носят порнографический и сексуальный характер, содержат насильственные или другие неподобающие действия. Уважительное отношение к различным точкам зрения и опыту.
- Распространение комментариев, которые разжигают ненависть, содержат недостойные выражения, слухи, оскорбления или уничижительные заявления, а также нападки в связи с личными и политическими убеждениями. Действия с учетом интересов сообщества.
- Преследования в публичном или частном порядке (сексуальное домогательство, травля или иные виды преследований) в отношении коллег или подчиненных.
- Разглашение личной информации, например адреса проживания или адреса электронной почты, без явного предварительного согласия.
- Другие поступки, которые могут быть отнесены к нарушению профессиональной этики.

Наши обязательства

Сотрудники по сопровождению проектов сообщества имеют право разъяснять смысл термина «надлежащее поведение» и делать справедливые замечания, если усмотрят нарушения в поведении кого-либо. Сотрудники по сопровождению проектов имеют право удалять, редактировать и отклонять любые комментарии, коды или вопросы, которые нарушают кодекс поведения. Сотрудники по сопровождению могут временно или на постоянной основе отстранить от проектов любого участника, поведение которого они сочли недопустимым, носящим угрожающий или оскорбительный характер, оказывающим тлетворное влияние.

Область применения

Кодекс поведения должны соблюдать все участники сообщества openEuler. Кодекс поведения распространяется на всех, кто выступает от имени сообщества, а также на публичные платформы, к которым присоединяется любой выступающий от имени сообщества.

Выступать от имени сообщества подразумевает в том числе следующее: использование официальной электронной почты сообщества, размещение сообщений в официальных СМИ и участие в онлайн- или офлайн-мероприятиях в качестве назначенного представителя.

Кодекс поведения от имени сообщества может быть в дальнейшем дополнен и интерпретирован в Секретариате и передан Совету для утверждения и опубликования.

Надзор и расследования

Чтобы сообщить об оскорблениях, преследованиях и недопустимом поведении, замеченных в какой-либо группе по проекту, отправьте электронное письмо Секретариату по адресу: secretary@openeuler.io.

Секретариат рассмотрит все жалобы, разберет все случаи недопустимого поведения и ответит в соответствии с принятым решением. Проектная группа обязана хранить в секрете информацию о лицах, сообщивших о нарушениях.

Исполнение

Сообщения о замеченных в проектной группе случаях оскорбления, преследования и недопустимого поведения отправляйте электронным письмом по адресу tc@openeuler.org.

Группа поддержки рассмотрит все жалобы, разберет все случаи недопустимого поведения и ответит в соответствии с принятым решением. Проектная группа обязана хранить в секрете информацию о лицах, сообщивших о нарушениях. Конкретные инструкции по исполнению могут быть в дальнейшем опубликованы отдельно.

Любое лицо, которое не соблюдает или не выполняет кодекс поведения, может быть временно или на постоянной основе лишено права участвовать в проекте до окончательного решения данного вопроса с ответственным по проекту или другими членами.

Вклад в сообщество

Мы прилагаем все усилия, чтобы документы и программное обеспечение, выпускаемые сообществом, были максимально высокого качества. Но как известно, любой документ можно улучшить (как и тот, которые вы читаете в данный момент), код требует проверки, функции или переменные могут быть доработаны или должны быть скорректированы в соответствии с замечаниями, а примеры тестирования могут быть дополнены и оптимизированы. Мы поможем вам понять организацию работы групп SIG и будем сопровождать вас во время первых шагов в сообществе openEuler.

Кратко о группах SIG

SIG (сокращение от Special Interest Group) — это группа, организованная в сообществе openEuler для решения той или иной проблемы и призванная улучшить менеджмент и оптимизировать процесс работы:

- Группы SIG открыты для всех, кто желает присоединиться и внести свой вклад.
- Каждая группа SIG создается для одного или нескольких конкретных технологических проектов. Члены группы SIG стремятся получить определенные результаты работы и сделать их принадлежностью сообщества openEuler.
- Ключевые члены SIG управляют группой. Ключевые члены имеют серьезный опыт и большой вклад, сделанный в сообщество.

- Каждая группа SIG ведет один или несколько проектов на Gitee, каждый проект располагает одним или несколькими репозиториями. В этих репозиториях хранятся результаты работы SIG.
- Будучи членом конкретной группы SIG, вы можете отправлять свои результаты в ее репозиторий, участвовать в обсуждениях, решать проблемы и заниматься проверкой.
Члены группы могут связываться с остальными членами SIG по электронной почте через адрес рассылки и во время видеоконференций.
Чтобы быстро получать ответы на отправляемые вами вопросы, необходимо найти подходящую группу SIG, которая занимается вопросами в интересующей вас области. Существует два способа поиска подходящей группы SIG.
- **Способ 1.** Чтобы найти подходящую группу SIG или подходящий проект, просмотрите список всех SIG, созданных в сообществе openEuler. Предоставляется следующая информация о SIG:
 - Проекты, контролируемые группой SIG и соответствующие адреса репозитория.
 - Способы связи с группой SIG, включая адрес рассылки электронной почты и видеоконференцию.
 - Контактная информация лица, сопровождающего проект.
- **Способ 2.** Если вам известно название проекта, вы можете быстро найти адрес домашней страницы требуемого проекта, выполнив поиск по неточному соответствию в списке репозиториях на веб-сайте openEuler. Как правило, информация о группе SIG, способы связи, имена членов группы и контактные данные содержатся в файле **README.md**, который можно найти на домашней странице проекта.

Если вы не можете найти интересующую группу SIG ни одним из указанных выше способов, отправьте письмо с вопросом по адресу community@openeuler.org. Рекомендуется указать тему письма [Вопрос по процессу разработки] и включить в тело письма детальные сведения по группе SIG или проекту, которые вы ищете.

Как внести свой вклад в проекты

Проверка проблемы

- **Поиск нужной проблемы.** На панели инструментов домашней страницы (репозиторий проекта на Gitee) интересующего проекта нажмите Issues, чтобы найти список проблем, решаемый данной группой SIG. Например, адрес списка проблем команды сообщества: <https://gitee.com/openeuler/community/issues>.
- **Проверка проблемы.** Если вы хотите заняться решением одной из проблем, назначьте сами себя. В поле ввода комментариев введите `/assign` или `/assign @[yourself]`. Бот сообщества назначит ваш аккаунт для решения проблемы, и ваше имя будет отображаться в списке ответственных.
- **Участие в обсуждении проблемы.** Любая проблема, возможно, уже обсуждалась участниками. Если вы заинтересованы в проблеме, вы можете комментировать ее в поле ввода комментариев.

Задание вопросов или внесение предложений

- **Как задать вопрос?** Обнаружив недочет или ошибку, вы можете задать вопрос администратору, отправив проблему в репозиторий списка проблем конкретного проекта. Задавая вопрос, соблюдайте правила его отправки.
- **Обратная связь.** Вы можете поделиться своими мнениями или предложениями с членами группы SIG, отправив вопрос, который будет считаться открытым для всех. Чтобы привлечь больше внимания, в электронном письме можно указать ссылку на проблему и отправить такое письмо всем участникам списка рассылки.

Создание среды разработки

1. Установите openEuler.
2. Подготовьте среду разработки.
3. Загрузите и настройте пакеты программного обеспечения.

Подробная информация приведена в *Приложении 1*.

Участие в написании кода

- Понимание принципов SIG и меры предосторожности при работе над проектом
В разных проектах SIG могут применяться разные язык, среда и правила написания кода. Чтобы понять принципы написания кода и участвовать в этом, необходимо ознакомиться с руководством для участников, которое адаптируется к каждому проекту. Руководство обычно содержится в файле **CONTRIBUTING.md** на домашней странице SIG или в файле **README.md** проекта. Помимо этих документов, SIG может также предоставлять другие руководящие принципы, которые вносятся в специально выделенный каталог группы SIG или каталог конкретного проекта. Если у вас нет нужной информации или появились вопросы по имеющейся информации, отправьте вопрос в группу SIG или по адресу рассылки SIG, к которому привязан проект. Если вы не получили ответа, отправьте вопрос по адресу: community@openeuler.org.
- Загрузка кода и запрос на создание нового ответвления проекта
Чтобы внести вклад в проект, необходимо научиться скачивать код с ресурса Gitee и отправлять запрос на принятие изменений кода (Pull Request). Метод использования хостинговой платформы аналогичен методу, применяемому GitHub. Если вы ранее работали с GitHub, можете ознакомиться с данной главой в общих чертах или даже пропустить ее.
- Изменение, сборка и локальная проверка кода
Завершив изменения в локальном ответвлении, выполните сборку и локальную проверку кода.
- Отправка PR-запроса
Отправка PR означает, что вы делаете вклад в сообщество, предлагая свои изменения кода.
- Добавление пакета программного обеспечения
После добавления пакета программного обеспечения в Gitee в сообществе openEuler автоматически создается репозиторий с тем же именем в **openEuler:Factory** на OBS. Таким образом, когда код передается в созданный репозиторий Gitee, автоматически проверяется скомпилированный код.

Проверка кода

openEuler — это открытое сообщество. Мы надеемся, что все участники сообщества смогут стать активными редакторами кода.

Чтобы сделать процесс принятия изменений удобным, все участники должны:

- Соблюдать правила участия в написании кода SIG, если таковые имеются.
- Подготовить полную информацию о предлагаемых изменениях.
- В случае отправки большого фрагмента кода рекомендуется разделить контент на небольшие логические части. Отправка контента частями облегчит редакторам задачу понимания предлагаемых вами идей.
- Отметьте PR-запрос соответствующей информацией группы SIG и контроля. Таким образом, бот сообщества поможет ускорить процесс подачи запроса PR.
В целях повышения эффективности сотрудничества редакторам настоятельно рекомендуется соблюдать кодекс поведения и уважительно относиться друг к другу. В статье *The Gentle Art Of Patch Review* содержатся ключевые моменты, касающиеся редактирования, которые должны понимать редакторы кода. Также редакторы должны поощрять новых участников к активному участию в проекте, не препятствовать новым начинаниям и не писать критикующие комментарии. Во время проверки обратите внимание на следующие моменты:
- Понятна ли идея в предлагаемых изменениях?
- Правильно ли оформлены изменения?
- Потребуется ли дальнейшие изменения?

Примечание. Если ваш запрос PR не привлекает достаточного внимания, вы можете обратиться за помощью, отправив письмо по адресу рассылки группы SIG или по адресу dev@openeuler.org.

Выбор пакета компонентов сообщества

Процесс создания пакета RPM, также называемый сборкой пакета, представляет собой задачу компиляции и привязки программного обеспечения и метаданных, например полного названия программы, описания и списка зависимостей, необходимых для нормальной работы. Таким образом, пользователи смогут устанавливать, удалять или обновлять соответствующее программное обеспечение с помощью инструмента управления пакетами.

- Руководство по сборке пакета. Сообщество openEuler стандартизирует различные проекты разработки ПО с открытым исходным кодом в логически последовательную систему. Для этого разрабатывается руководство по сборке пакета RPM.
- Чтобы стандартизовать создание дистрибутивов, openEuler следует проекту Linux Standard Base (LSB). Также openEuler следует стандарту иерархии файловой системы Linux Filesystem Hierarchy Standard (FHS). Этот стандарт призван унифицировать размещение файлов и каталогов в файловой системе Linux.
- Помимо этих общих правил, по которым создаются дистрибутивы Linux, этот документ унифицирует фактическую информацию о пакетах версий, выпускаемых сообществом openEuler.

Обнаружение проблем безопасности в сообществе

- Процесс реагирования на проблемы безопасности
- Информация о политике безопасности

Подробные сведения о процессе реагирования на проблемы безопасности и информации о политике безопасности приведены в *Приложении 2*.

09/ Авторские права

Все материалы и содержимое данного документа защищены законом об авторском праве, и все авторские права принадлежат сообществу openEuler, если иное не указано со ссылкой на третьи лица. Без предварительного письменного разрешения сообщества openEuler или других соответствующих сторон ни одно лицо или организация не имеют право воспроизводить, распространять, перепечатывать или публиковать любой материал настоящего документа в любой форме; ссылаться на содержание документа или передавать его путем указания гиперссылок; выгружать контент на другие серверы в виде скриншотов или сканов; хранить контент в поисковых системах; или использовать контент для любых других коммерческих целей. Контент веб-сайта можно скачивать или распечатывать для некоммерческого и личного пользования при условии, что он не будет изменен и что все заявления о правах будут сохранены.

10/ Товарные знаки

Все товарные знаки и логотипы, используемые и приведенные в этом документе, принадлежат сообществу openEuler, за исключением товарных знаков, логотипов и торговых наименований, принадлежащих другим организациям. Без письменного согласия сообщества openEuler или других сторон любой материал в настоящем документе не дает разрешение или право на использование каких-либо из вышеупомянутых товарных знаков и логотипов косвенным образом или по причине отсутствия возражений со стороны владельцев, или по другим причинам. Без предварительного письменного согласия не разрешается использовать наименование, товарный знак или логотип сообщества openEuler в любой форме.

11/ Приложения

Приложение 1. Создание среды разработки

Подготовка среды	Адрес
Загрузка и установка ОС openEuler	https://openeuler.org/en/download/
Подготовка среды разработки	https://gitee.com/openeuler/community/blob/master/en/contributors/prepare-environment.md
Компоновка пакета программного обеспечения	https://gitee.com/openeuler/community/blob/master/en/contributors/package-install.md

Приложение 2. Процесс реагирования на проблемы безопасности и информация о политике безопасности

Обнаружение проблем безопасности в сообществе	Адрес
Процесс реагирования на проблемы безопасности	https://gitee.com/openeuler/community/blob/master/zh/security-committee/security-process.md
Информация о политике безопасности	https://gitee.com/openeuler/community/blob/master/zh/security-committee/security-disclosure.md

