Архитектура QEMU

- Интерпретатор моделирующий цикл исполнение CPU
- Динамический бинарный транслятор базовых блоков
- Трансляция: машинный код гостя -> простой IR -> непривилегированный машинный код хоста
- Внешние прерывания обновляют флаг проверяемый интерпретатором и порождённым кодом
- Программная модель MMU (логический TLB)
- Программные модели устройств
- Моделирование всей машины или только режима пользователя с трансляцией системных вызовов
- Опциональный ускоритель режима ядра (KVM)

Транслятор QEMU

- Используется TCG (tiny code generator) из простого компилятора C
- JIT компилятор базовых блоков машинного кода (translated block)
- IR – простая 3-х операндная нотация, типичный RISC LIR
- Результаты компиляции сохраняются в кэше индексированном адресом начала блока (PA или VA?)
- Компиляция только при отсутствии в кэше
- Регистры целевой архитектуры – глобалы компилятора
- Сложные операции – код на C вызываемый из порождённого кода

Управление памятью в QEMU

- Трансляция адресов при необходимости
- Программная модель MMU
- Кэш трансляции адресов (TLB) в состоянии виртуального процессора
- Обработчики доступа для MMIO и некоторых критических участков
- Точная обработка исключений виртуального CPU (благодаря примитивности компилятора)
- Сброс TLB при обновлении указателя таблицы страниц (ещё?)

Некоторые оптимизации в QEMU

- Фиксированное назначение некоторых регистров (каких?)
- Выход из цикла интерпретации longjmp()
- Ссылки на TB из описания страницы (зачем?)
- Прямая линковка TBs
- Использование MMU хоста (через сигналы ОС)
- Использование системы непосредственного исполнения непривилегированного кода при возможности

Valgrind

- Изначально система поиска ошибок работы с памятью (memcheck)
- Позднее расширена следующими инструментами
  - *Massif (профайлер хипа)*
  - *Helgrind (поиск race conditions)*
  - *Cachegrind (профилятор кэша)*
- Динамический бинарный транслятор порождающий инструментированный код
- Отслеживает состояние битов памяти
- Или кэша, или…

Вопросы

- Какие сложности может встретить попытка добавить в QEMU оптимизирующий компилятор?
- Какие процессоры легче моделировать с QEMU?
- Какие процессорные оптимизации можно добавить в QEMU?
- Какие расширения для Valgrind предложили бы вы?