



# GCC for openEuler 优化介绍及新特性前瞻

华为技术有限公司

Compiler SIG Maintainer 李彦成

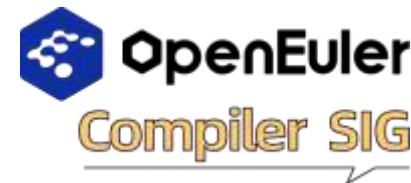
# 目录

1. GCC for openEuler整体介绍
2. 主要特性示例
3. 新特性前瞻

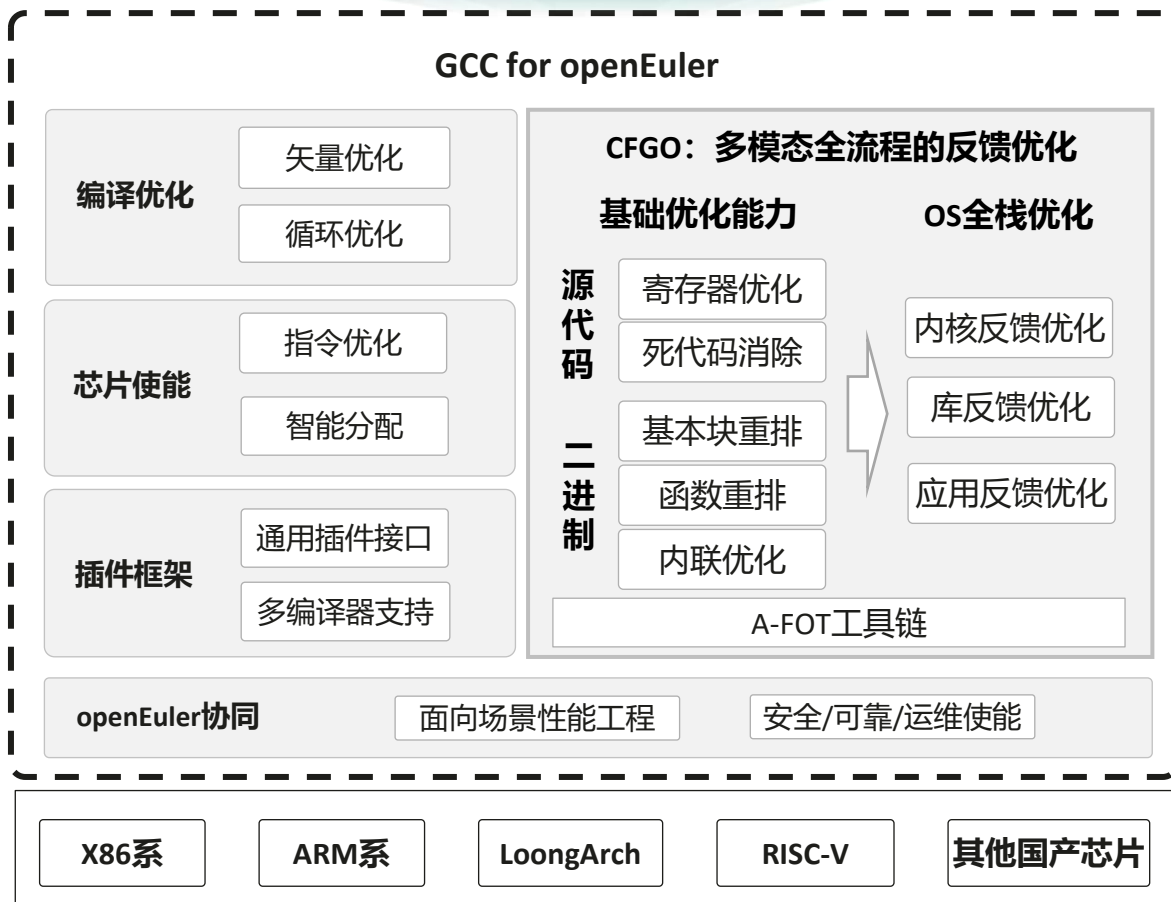
# 目录

1. GCC for openEuler整体介绍
2. 主要特性示例
3. 新特性前瞻

# GCC for openEuler: 使能多样算力, 聚焦便捷易用、生态兼容和场景化性能增强

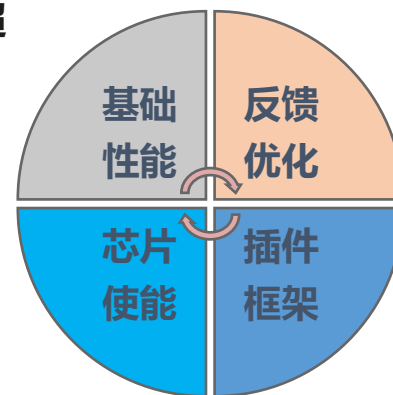


GCC for openEuler



■ SPEC2017 INT性能超  
开源GCC 20%

■ 使能多样算力,  
HPC内核函数性能  
提升30%



■ 一键启用反馈优化,  
数据库性能提升18%

■ 一套插件兼容不同编译  
框架, 使能多样算  
力差异化编译诉求

GCC for openEuler使能多样算力, 聚焦于便捷易用、生态兼容和场景化性能增强, 并在以下四个方向实现主要突破。

- **基础性能:** 基于GCC开源版本构建竞争力, 通过在openEuler社区推进鲲鹏特性的支持和短板的补齐, 提升通用场景性能。
- **反馈优化:** 整合业界领先的反馈优化技术, 实现程序全流程和多模态反馈优化, 提升数据库等云原生场景重点应用性能。
- **芯片使能:** 使能多样算力指令集, 围绕内存等硬件系统, 发挥算力优势, 提升HPC等场景化性能。
- **插件框架:** 使能多样算力差异化编译诉求, 一套插件兼容不同编译框架, 打通GCC和LLVM生态。

# 目录

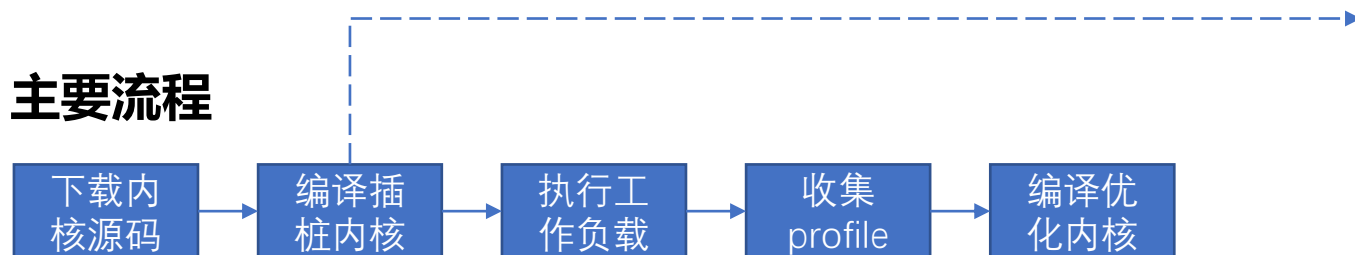
1. GCC for openEuler整体介绍
2. 主要特性示例
3. 新特性前瞻

# 内核反馈优化

## 特性进展

- 编译器软件插桩反馈优化 (*aka PGO, FDO*) 在 openEuler linux kernel 上使能

## 主要流程



Configure openEuler kernel with

- `CONFIG_PGO_KERNEL=y`
- `CONFIG_GCOV_KERNEL=y`
- `CONFIG_ARCH_HAS_GCOV_PROFILE_ALL=y`
- `CONFIG_GCOV_PROFILE_ALL=y`
- `CONFIG_DEBUG_FS=y`

## 适用场景

- 工作负载相对固定
- 期望最大化性能
- 接受重新编译内核

## 易用性提升

- A-FOT新模式: Kernel PGO
- 和以往一样: 用户通过简单配置, 使用A-FOT可一键自动完成整个优化流程, 得到新的内核

## 优化效果

| 应用    | 内核反馈优化 | 内核+应用反馈优化 |
|-------|--------|-----------|
| MySQL | +2~4%  | +10~18%   |
| Ceph  | +5~7%  | +7~10%    |
| Nginx | +5~15% | +10~20%   |

# 插件框架

## ➤ 个性化编译的便捷开发

- 无需深入修改编译器内部逻辑，帮助开发者可以**实现独立编译优化和编译工具的便捷开发**

## • 特性进展

- ① 能够覆盖80%的GIMPLE
- ② 支持LTO阶段使能插件
- ③ LLVM客户端原型

(<https://gitee.com/openeuler/pin-llvm-client>)

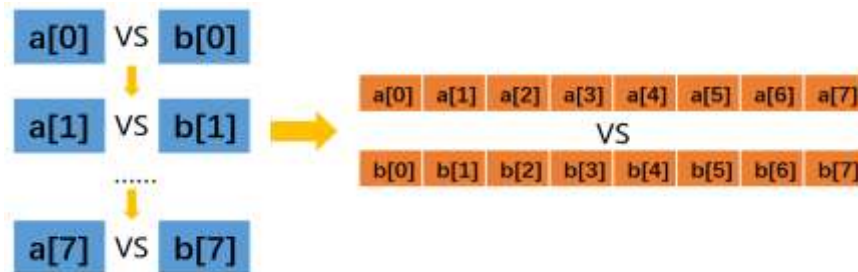
## • 欧洲开源峰会（OSSEU23）主题报告



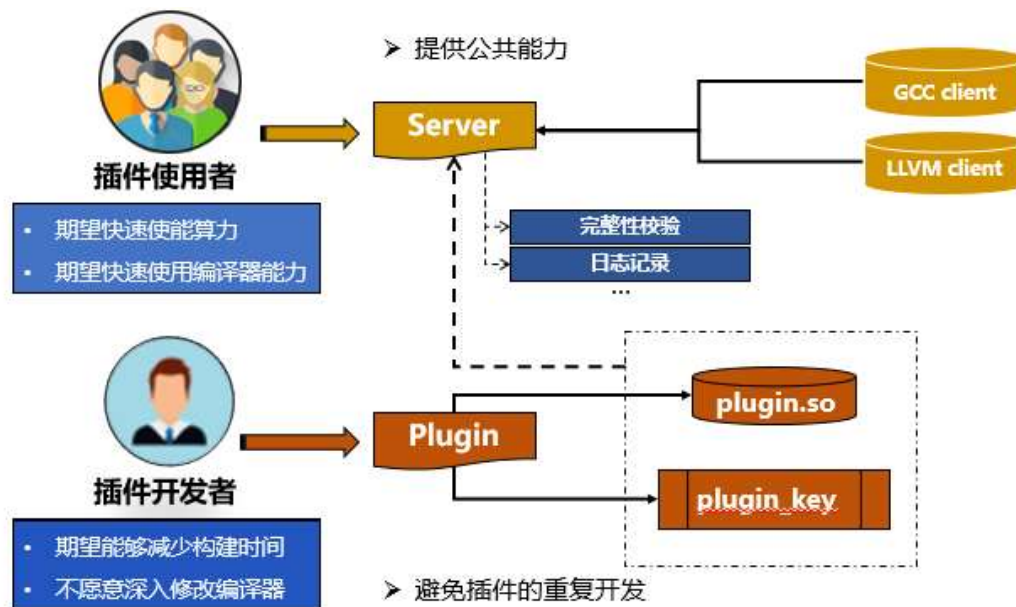
The compiler plugin framework to facilitate customized compilation and development

### 查询元素扩展优化插件

- 可以支撑GCC for openEuler的优化特性，往插件框架迁移



- **提升效果：** SPEC子项提升**7%**；开发效率由1人月降至**7人天**



# 恒安嘉新：GCC for openEuler加速核心业务性能



业务逻辑

恶意程序  
僵木蠕

SQL注入  
XSS  
webshell  
网站安全事件

钓鱼网站  
赌博网站  
游戏私服

虚拟身份  
轨迹、敏感信息

高速正则  
事件引擎

内容结构  
监测引擎

异常行为  
识别引擎

文件还原  
研判引擎

流量回溯  
引擎

应用协议识别解析

基础流量处理

GCC for openEuler

Boostkit基础加速库

openEuler

鲲鹏920

基础解析  
扫描引擎

基础平台

恒安嘉新是一家数字应用智能化服务商，深耕网络安全领域，构建面向数字中国基础设施的数字安全保护屏障，包含金川（网络空间治理）、金御（大数据安全应用）、云河（SaaS服务）三大核心业务，与三大运营商合作，服务范围覆盖31省

## 业务挑战

- **大流量**：单台设备吞吐量大于100Gbps，并发连接数不少于1000万，新建连接不低于每秒5万；
- **高准确**：处置成功应不低于95%，协议识别准确率不低于90%

## 解决方案

- **鲲鹏亲和优化**：GCC for openEuler通过亲和鲲鹏920的流水线优化，ccmp等指令级优化，实现汇编指令鲲鹏亲和，核心业务性能提升5%
- **全局优化**：GCC for openEuler通过全局优化，实现跨文件函数内联，减小调用开销，跨文件函数常量传播，消除冗余代码，核心业务性能提升5%

## 客户收益

- **问题定位**：联合GCC for openEuler团队快速定位解决CPU使用率不均衡、系统升级后业务系统严重丢包问题，保障客户DPI&协议解析平台业务稳定运行
- **迁移结果**：GCC for openEuler在恒安嘉新DPI&协议解析场景落地，支撑XXX节点稳定运行，业务性能提升5%~10%



# 目录

1. GCC for openEuler整体介绍
2. 主要特性示例
3. 新特性前瞻

# 基础性能：编译优化能力持续增强

## Double-sized mul

将64位乘法算法转换为更高效的指令  
openssl提升20%

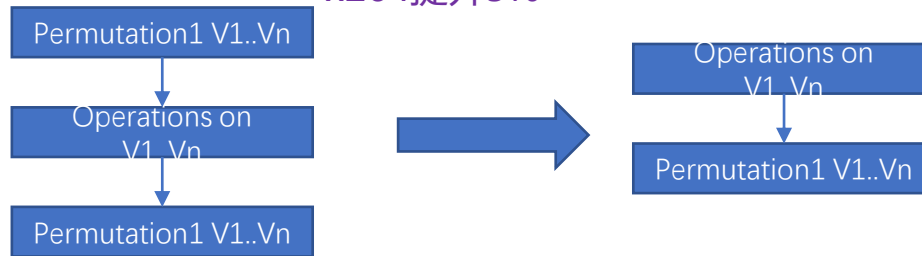
```
uint128_t mul128_perm(uint64_t a, uint64_t b)
{
    uint64_t a_lo = a & 0xFFFFFFFF;
    uint64_t b_lo = b & 0xFFFFFFFF;
    uint64_t a_hi = a >> 32;
    uint64_t b_hi = b >> 32;
    uint64_t lolo = a_lo * b_lo;
    uint64_t lohi = a_lo * b_hi;
    uint64_t hihl = a_hi * b_lo;
    uint64_t hihh = a_hi * b_hi;
    uint64_t middle = hihl + lohi;
    uint64_t middle_hi = middle >> 32;
    uint64_t middle_lo = middle << 32;
    uint64_t res_lo = lolo + middle_lo;
    uint64_t res_hi = hihh + middle_hi;
    res_lo = res_lo < middle_lo ? res_lo + 1 : res_lo;
    res_hi = middle < hihl ? res_hi + 0x100000000 : res_hi;
    uint128_t res = ((uint128_t) res_hi) << 64;
    res += res_lo;
    return res;
}
```

mov x2, x1  
umulh x1, x0, x1  
mul x0, x0, x2

利用硬件资源

## Forward propagation of permutations

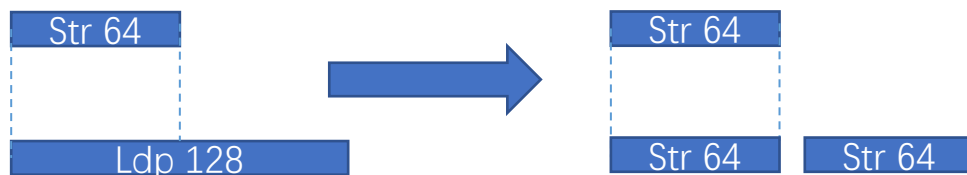
优化矢量化指令数  
x264提升5%



优化指令数量

## Split complex instructions

针对ldp/stp表现较差的场景，拆分为2个ldr/str  
rapidjson提升7%



补充硬件能力

除此之外，还计划推出如下增强优化，期待大家的尝试与反馈：

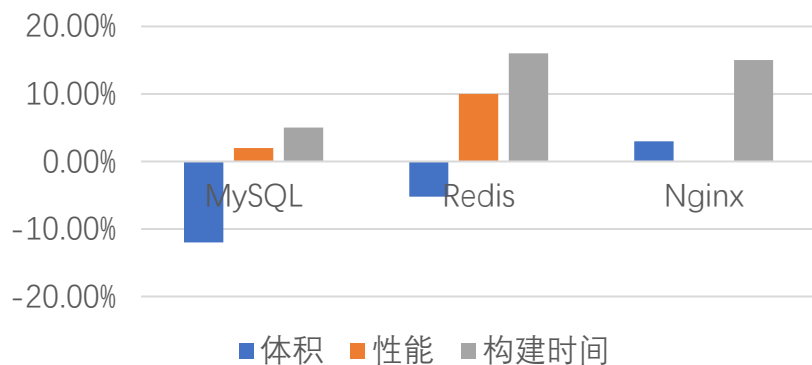
- LLC Allocation optimization
- CRC32 optimization
- Indirect Call Promotion
- IPA prefetch
- ...

# 高级优化：LTO和AI4Compiler

## LTO (link time optimization) openEuler by default

### 策略

- 与构建工程团队合作，默认所有包使能LTO
- 遇到无法解决的issue再单包disable



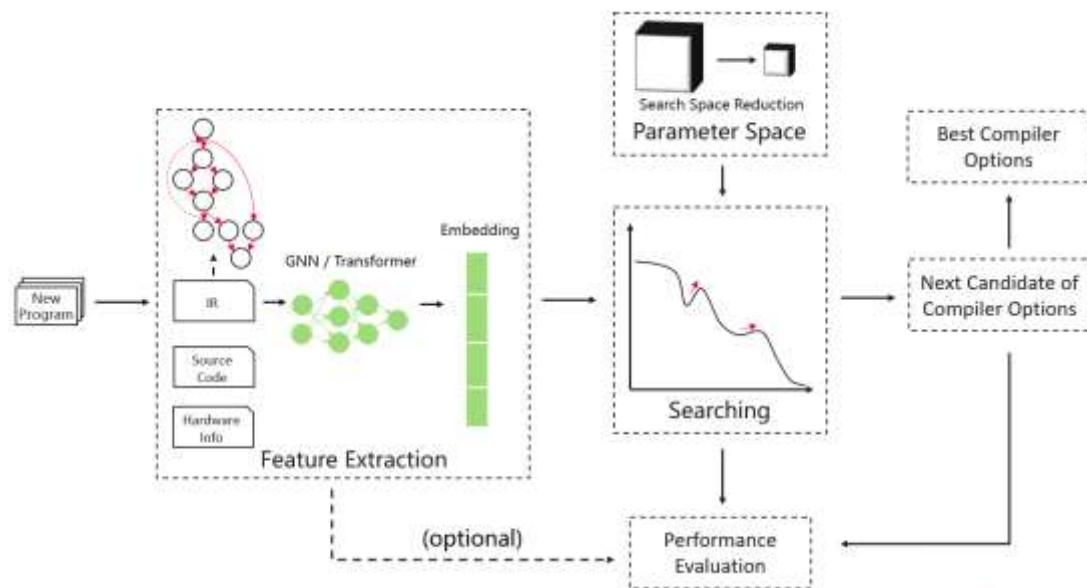
### 优势

- 更小的软件包体积
- 更优的性能（开箱）

## AI4compiler——编译选项自动调优

### 策略

- 与算法团队合作，实现对白盒内容的编码与代码结构特征识别
- 优化搜索算法与参数搜索空间



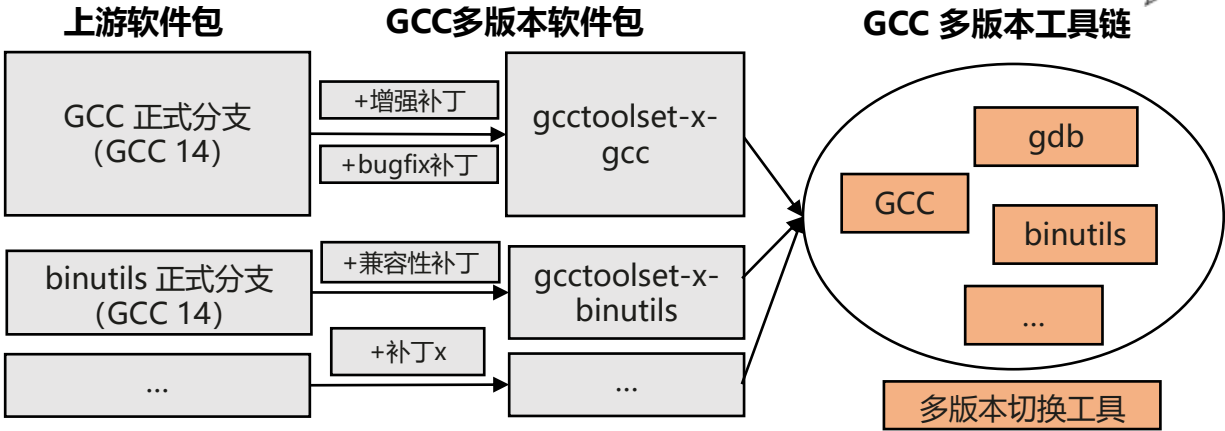
### 优势

- 白盒分析代码特征，缩短应用的调优时长
- 加速迁移对不同环境和硬件底座的调优

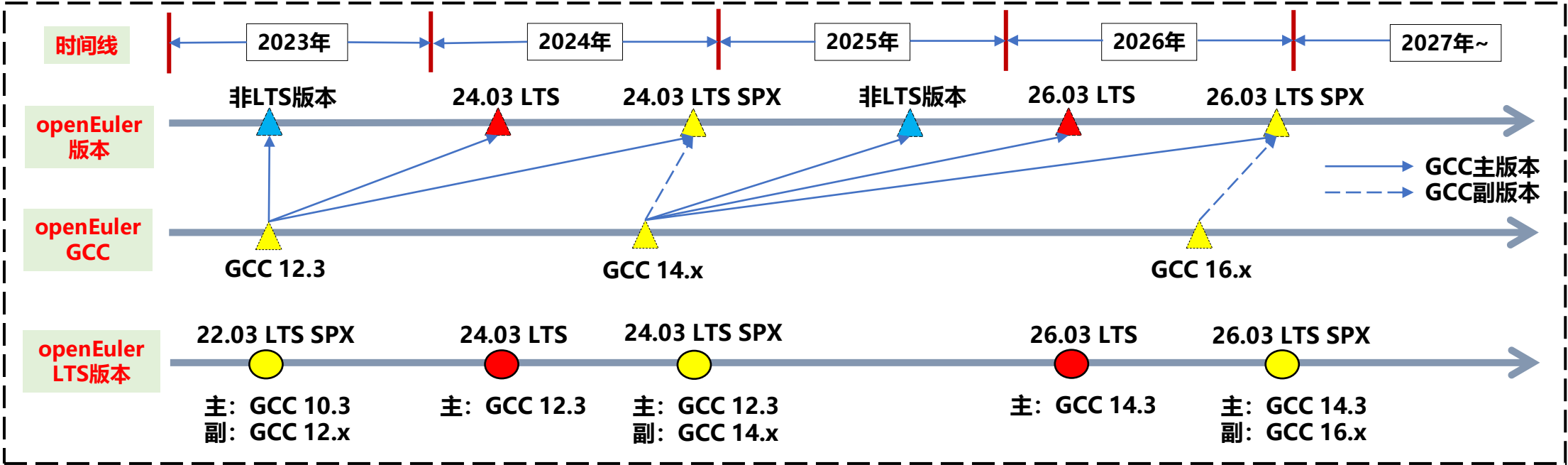
# 用户体验：GCC多版本支持



- **GCC多版本工具链增强兼容性、易用性**
- 工具链代码由三个层次演进构成：
  - **上游软件包**：GCC、binutils等上游社区正式分支
- ① **GCC多版本软件包**：增加bugfix、兼容性等补丁
- ② **GCC多版本工具链**：整合各软件包，提供切换工具

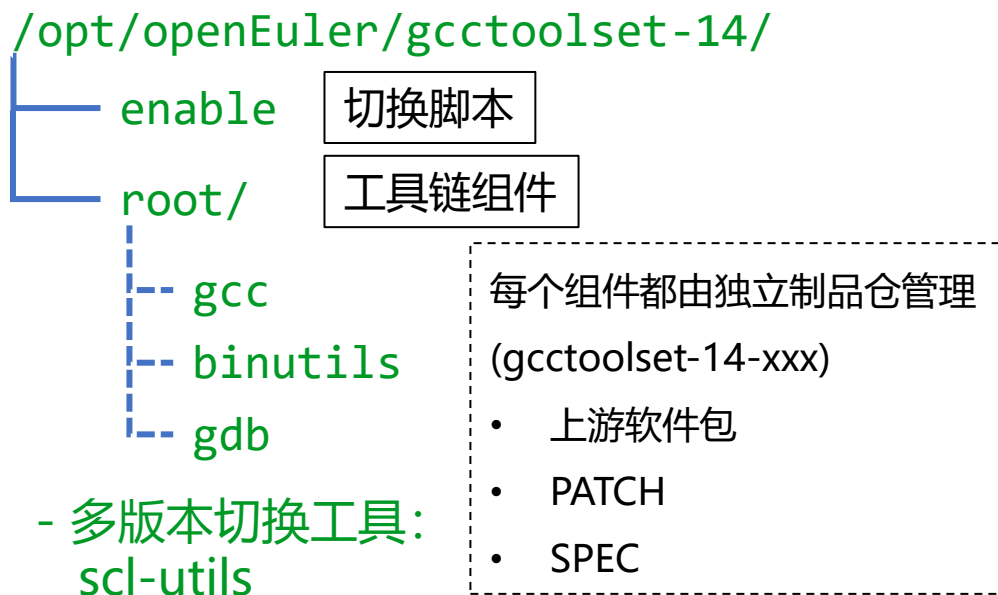


## GCC多版本版本规划



# openEuler GCC多版本支持方案

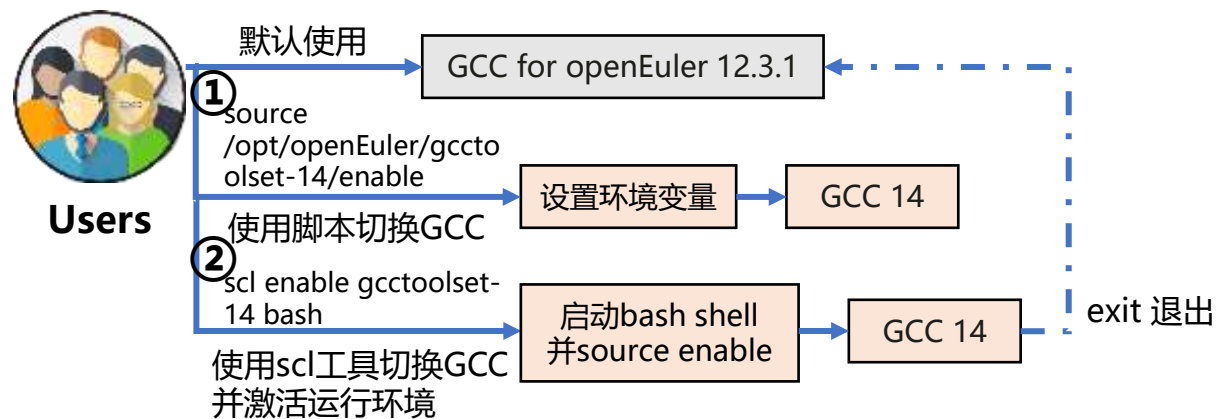
## • 引入GCC14副版本工具链：gcctoolset-14



## • 测试计划

- 默认版本的测试方案：基础功能测试、可靠性测试...
- 多版本兼容性测试（设计中，收集建议）

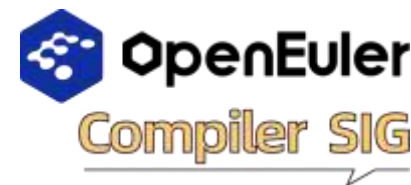
## • 用户使用：提供两种切换方法



## • 兼容性说明

- 场景：默认版本GCC的环境运行高版本GCC编译出的应用
- 问题：高版本GCC链接自带的高版本GLIBCXX，使用的高版本符号，在默认版本运行出现找不到符号错误
- 方案：拆分gcctoolset-14-gcc的libstdc++为两部分
  - 与系统默认GCC环境一致的shared.so文件
  - 高版本libstdc++增加的部分：**non-shared.a**文件
- 约束：仅保证当前多版本工具链和系统默认GCC版本兼容

# devtoolset-14组件选型



| 组件       | 描述     | 建议选型  | c7-dts10              | c7-dts12               | openEuler               | 最新版本                  |                  |
|----------|--------|-------|-----------------------|------------------------|-------------------------|-----------------------|------------------|
| GCC      | 编译器套件  | 14.x  | 10.2.1<br>(2020-7-23) | 12.2.1<br>(2022-08-19) | 12.3.1                  | 13.2<br>(2023-07-27)  |                  |
| binutils | 二进制工具集 | 2.41  | 2.35<br>(2021-01-09)  | 2.36.1<br>(2021-02-06) | 2.40-5<br>(2023-01-14)  | 2.41<br>(2023-07-30)  | 最新               |
| gdb      | 命令行调试器 | 13.2  | 9.2<br>(2020-05-23)   | 11.2<br>(2022-01-16)   | 12.1-10<br>(2022-05-01) | 14.1<br>(2023-11-03)  | 13.2 (23-05-27)  |
| annobin  | 安全检查工具 | 12.22 | 9.23<br>(2020-07-01)  | 11.08<br>(2023-01-31)  | 8.92-1<br>(2019-12-06)  | 12.32<br>(2023-11-24) | 12.22 (23-07-26) |
| elfutils | Elf工具集 | 0.190 | 0.182<br>(2020-10-31) | 0.187<br>(2022-04-25)  | 0.189-3<br>(2023-05-03) | 0.190<br>(2023-11-03) | 最新               |

- 由于GCC 14暂未release, 其他相关组件参考GCC 13.2进行选型



# Thank You.

---

Compiler SIG 专注于编译器领域技术交流探讨和分享，包括 GCC/LLVM/OpenJDK 以及其他的程序优化技术，聚集编译技术领域的学者、专家、学术等同行，共同推进编译相关技术的发展。



毕昇编译公众号



Compiler 交流群小助手