ISCAS 中国科学院软件研究所学术年会'2024
暨重点实验室科技活动周

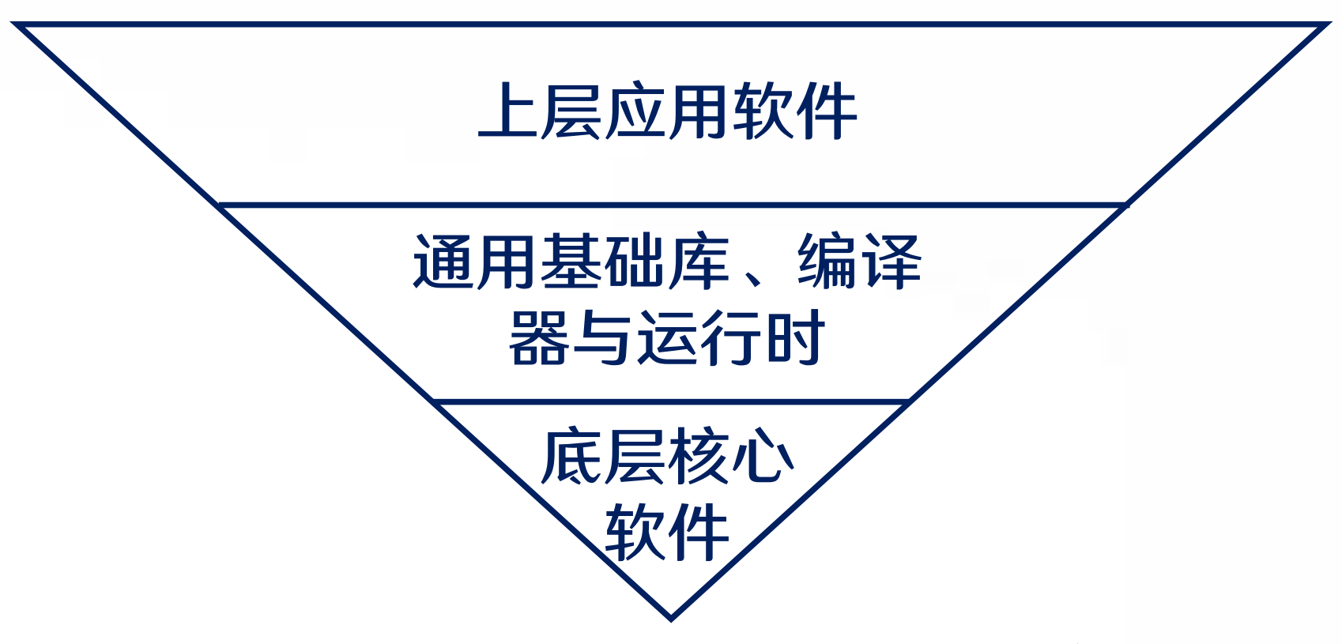
关键技术

基于开源软件演化历史的 RISC-V 架构适配相关问题挖掘与分析技术研究

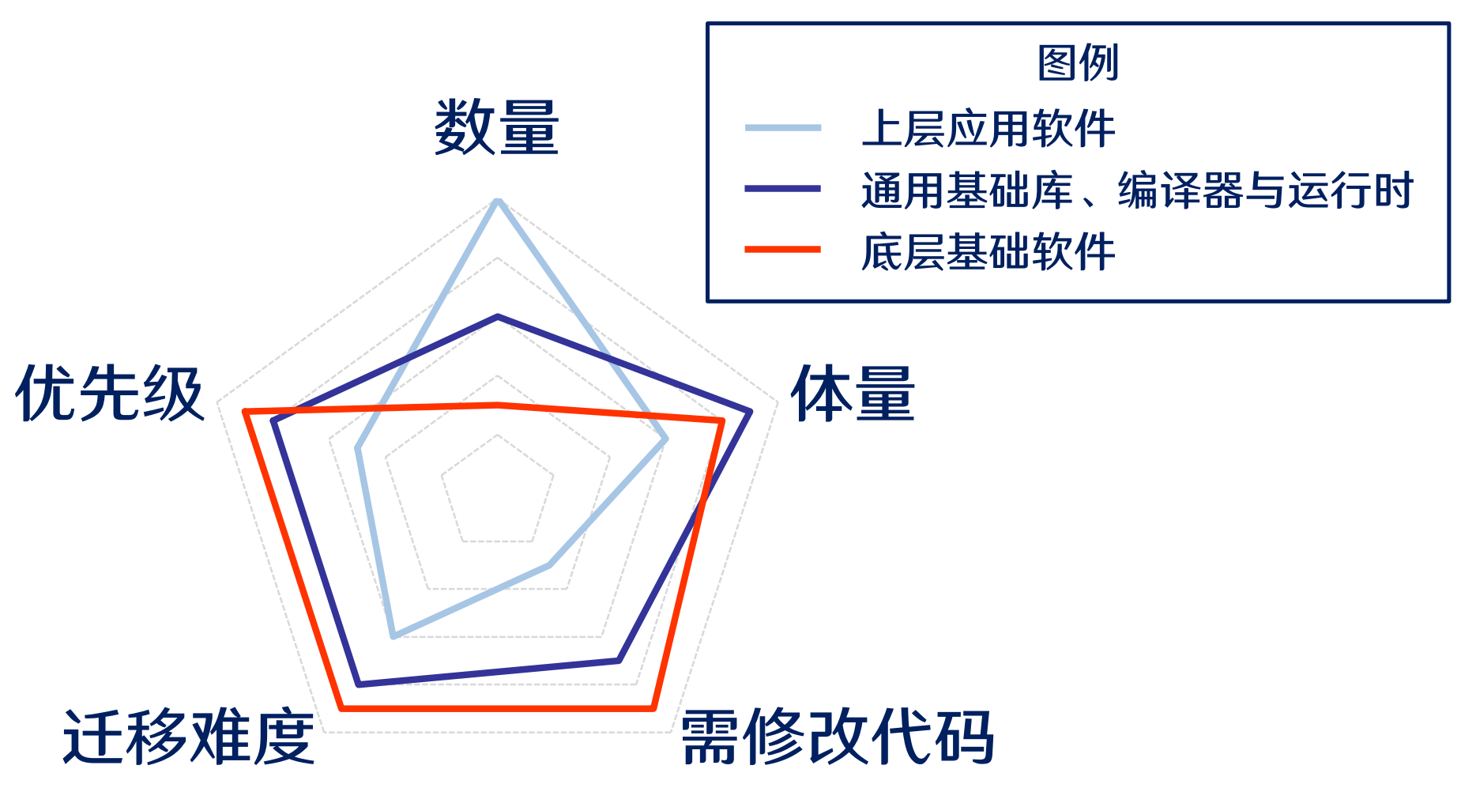
张世新, 孟鑫, 燕季薇, 严俊, 朱家鑫, 王伟

软件工程技术研究开发中心　zhangshixin@otcaix.iscas.ac.cn

研究背景



待适配 RISC-V 架构软件性质“倒三角”结构



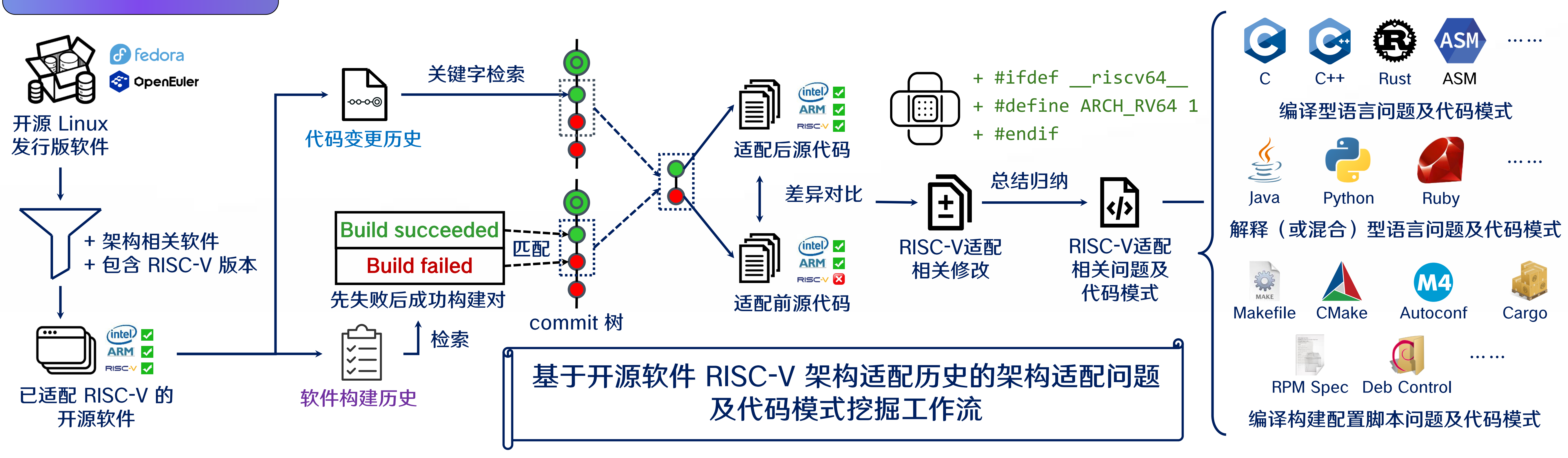
待适配 RISC-V 架构的各类软件整体特点雷达图

待适配 RISC-V 架构的各类软件整体特点

软件类型	数量	体量	需修改代码	适配难度	优先级	额外要求
上层应用软件	多	不定	少且零散	一般或较低	较低	功能正常
通用基础库、编译器与运行时	一般	大	较多且复杂	高	高	可靠性高
底层核心软件	少	大	多且复杂	高	高	可靠性高

- ➢ RISC-V 作为近年的新兴指令集，具有开源开放的优势和广阔的应用前景
- ➢ 将已支持现有指令集架构（如 X86、ARM 等）的开源软件迁移适配 RISC-V 架构，是快速丰富 RISC-V 软件生态的重要途径，目前开源社区已有大量实践
- ➢ 软件在新指令集架构上的适配是一个“层层依赖、环环相扣”的过程
- ➢ 不同性质的软件具有不同特点，其架构适配难度、优先级和额外要求存在差异
- ➢ 如何从已适配 RISC-V 架构的软件中挖掘并分析迁移适配相关问题、总结归纳解决方案，以指导未来其它软件的 RISC-V 架构迁移适配工作？

技术方案



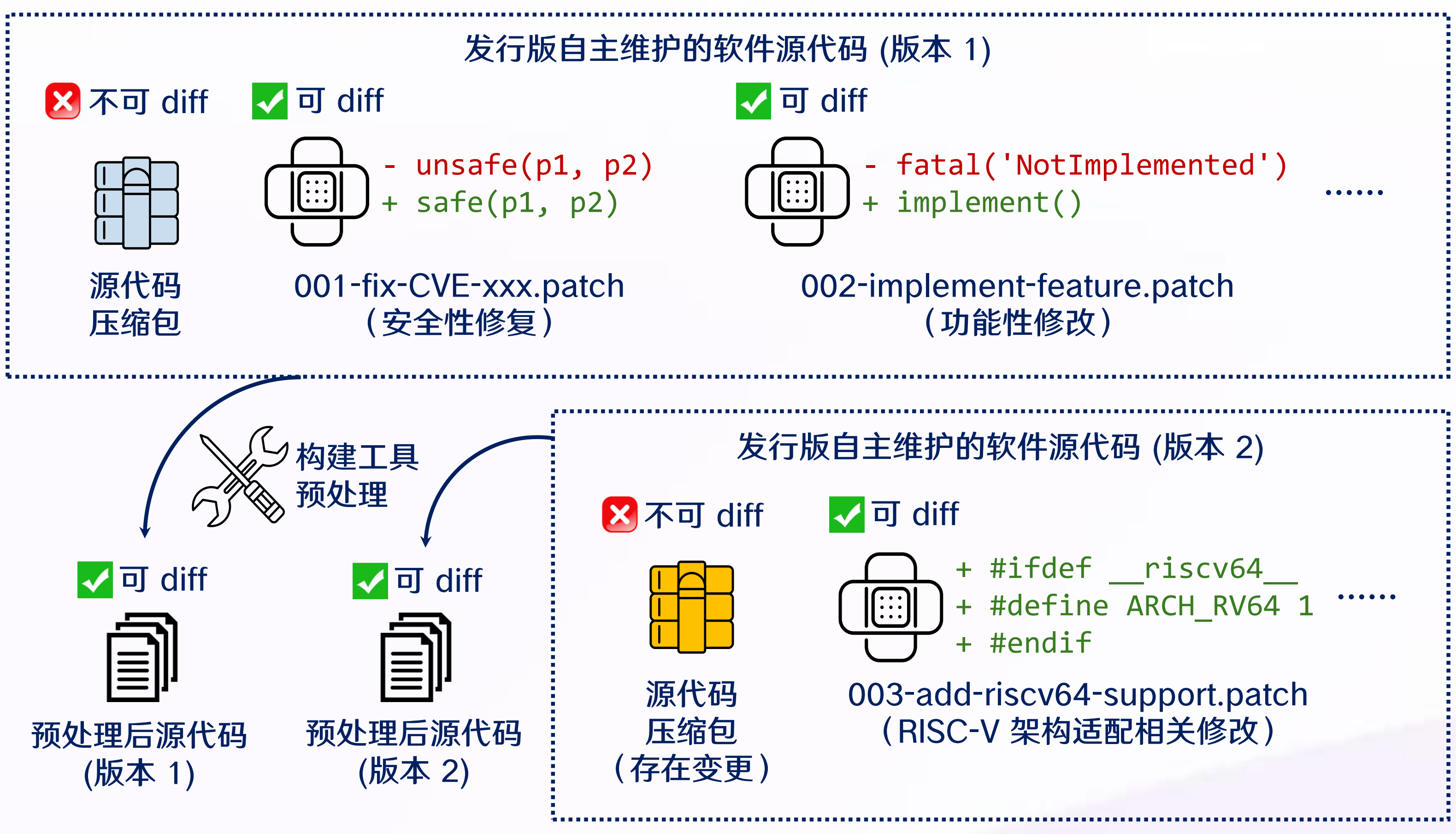
```
+ #ifdef __riscv64__
+ #define ARCH_RV64 1
+ #endif
```

基于开源软件 RISC-V 架构适配历史的架构适配问题及代码模式挖掘工作流

- ➢ 开源 Linux 操作系统发行版自主管理的开源软件仓库和构建平台中，包含软件在各种指令集架构下的最新版本制品包及源代码
- ➢ 软件的代码变更历史和构建历史中，包含软件针对 RISC-V 架构适配的修改和构建结果，其中蕴含专业开发者的迁移适配知识
- ➢ 通过关键字检索和“先失败后成功”构建对匹配的方式，从软件源代码的 commit 树中匹配出适配 RISC-V 架构前后的源代码
- ➢ 对匹配到的源代码进行差异对比，得到 RISC-V 适配相关修改，进一步按照编程语言的性质归纳总结架构适配问题和代码模式

源代码差异对比失效问题及解决方法

- ➢ 一般源码仓库：直接包含源代码文件
- ➢ 发行版源码仓库：源代码压缩包 + 补丁文件
- ➢ 从发行版仓库中提取软件源代码时，可能面临源代码差异对比失效的问题，即无法直接通过版本控制工具对比源代码压缩包中发生的变更
- ➢ 解决方案：模仿软件构建系统行为，将选定的版本内软件源代码压缩包展开，按照构建配置应用补丁文件，使压缩包内发生的源代码变更可通过版本控制工具进行对比
- ➢ 初步实验效果：95% 以上的软件均可正常处理



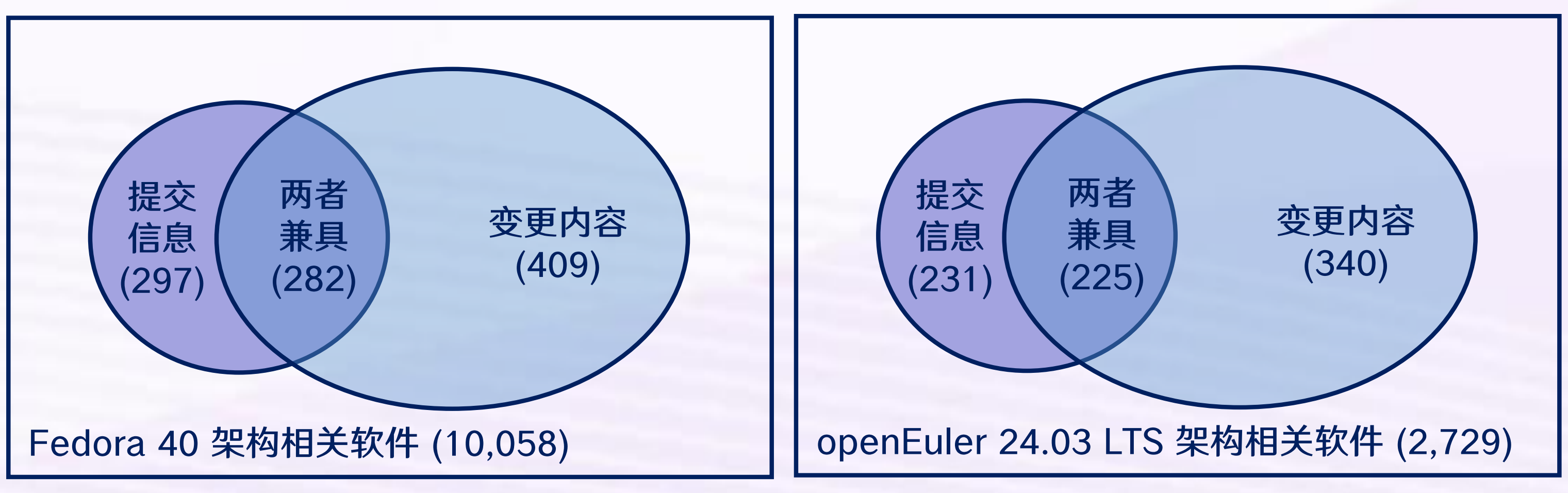
```
- unsafe(p1, p2)
+ safe(p1, p2)
```

```
- fatal('NotImplemented')
+ implement()
```

```
+ #ifdef __riscv64__
+ #define ARCH_RV64 1
+ #endif
```

源代码差异对比失效问题及解决方法示意图

应用效果



两种检索匹配策略对存在 RISC-V 指令集架构适配相关修改软件的贡献情况

开源软件 RISC-V 架构适配相关修改的识别效果

- ➢ 在 Fedora 40 RISC-V 版本提供的 23,614 款软件中，识别到 10,058 款架构相关软件，其中共有 424 款包含 RISC-V 架构相关修改的软件
- ➢ 在 openEuler 24.03 LTS RISC-V 版本提供的 6,012 款软件中，识别到 2,729 款架构相关软件，其中共有 346 款包含 RISC-V 架构相关修改的软件

开源软件 RISC-V 架构适配相关问题及代码模式总结

基于目前收集的开源软件 RISC-V 指令集架构适配相关修改内容，已总结形成“三大类七小类”软件源代码中常见的 RISC-V 架构适配问题和解决方案

- ➢ 编译型语言常见问题：内联汇编、Intrinsics 函数、内存对齐、其它要素
- ➢ 解释（或混合）型语言常见问题：Native 库、生成代码或编译配置
- ➢ 编译构建配置脚本常见问题：架构相关编译参数、架构特有软件依赖关系

```
#if defined __arm__ || defined __ARCH_ARM__
#define
X86 架构特有宏 c_cxx(CC-0003)
example.c(6, 15): Defined under macro: __i386__
#elif de
      de
查看问题 (Alt+F8)   没有可用的快速修复
#define RETURN_INSTR 0xC3C3C3C3 /* ret; ret; ret; ret */
#define FUNC __builtin_ia64_flush_rs()

#ifdef TEST_MACRO
asm ("mov eax, 1");
__asm ( "mov ebx, 0ffh" );
```

CC-0004 架构相关 builtin 函数

描述

::: warning 警告

通常而言，__builtin_ 开头的编译器内置函数具有其上层包装的方法，理论上 *非编译器的普通程序* 应当使用上层包装方法，而不应直接使用这些内置函数。特殊情况需要使用时，应添加 __has_builtin 预处理宏用于检查欲使用的内置函数是否存在。

RISC-V 架构适配目标代码定位辅助工具效果示意图