

v8::internal::compiler
::anonymous_namespace
{instruction-selector-arm64
::cc}::VisitAtomicStore

v8::internal::compiler
::anonymous_namespace
{instruction-selector-loong64
::cc}::VisitAtomicStore

v8::internal::compiler
::anonymous_namespace
{instruction-selector-mips64
::cc}::VisitAtomicStore

v8::internal::compiler
::anonymous_namespace
{instruction-selector-riscv64
::cc}::VisitAtomicStore

v8::internal::compiler
::AtomicWidthSize

```
graph LR; A["v8::internal::compiler::anonymous_namespace{instruction-selector-arm64::cc}::VisitAtomicStore"] --> D["v8::internal::compiler::AtomicWidthSize"]; B["v8::internal::compiler::anonymous_namespace{instruction-selector-loong64::cc}::VisitAtomicStore"] --> D; C["v8::internal::compiler::anonymous_namespace{instruction-selector-mips64::cc}::VisitAtomicStore"] --> D; E["v8::internal::compiler::anonymous_namespace{instruction-selector-riscv64::cc}::VisitAtomicStore"] --> D;
```

The diagram illustrates a code transformation or specialization process. On the left, there are four white rectangular boxes, each representing a function from a different instruction selector architecture: arm64, loong64, mips64, and riscv64. Each box contains the text 'v8::internal::compiler::anonymous_namespace{instruction-selector-arch::cc}::VisitAtomicStore', where 'arch' is the specific architecture. Blue arrows point from each of these four boxes to a single gray rectangular box on the right. This gray box is labeled 'v8::internal::compiler::AtomicWidthSize', representing a common, specialized function that all the architecture-specific functions are being mapped to.