

v8::internal::RecordMigrated
SlotVisitor::VisitInstructionStream
Pointer

v8::internal::Unreachable
ObjectsFilter::MarkingVisitor
::VisitInstructionStreamPointer

v8::internal::MarkCompact
Collector::CustomRootBodyMarking
Visitor::VisitInstructionStreamPointer

v8::internal::IndexedReferences
Extractor::VisitInstructionStreamPointer

v8::internal::ObjectVisitor
WithCageBases::code_cage_base

```
graph LR; A[v8::internal::RecordMigrated SlotVisitor::VisitInstructionStream Pointer] --> D[v8::internal::ObjectVisitor WithCageBases::code_cage_base]; B[v8::internal::Unreachable ObjectsFilter::MarkingVisitor ::VisitInstructionStreamPointer] --> D; C[v8::internal::MarkCompact Collector::CustomRootBodyMarking Visitor::VisitInstructionStreamPointer] --> D; E[v8::internal::IndexedReferences Extractor::VisitInstructionStreamPointer] --> D;
```

The diagram illustrates a dependency or inheritance relationship. On the left, there are four white rectangular boxes, each containing a text string representing a specific visitor or filter class and its associated `VisitInstructionStreamPointer` method. Arrows from each of these four boxes point towards a single gray rectangular box on the right. This gray box represents a base class or interface, `v8::internal::ObjectVisitor WithCageBases::code_cage_base`, which likely defines the common `VisitInstructionStreamPointer` method that the other classes implement or inherit from.