WARNING: The AnnotationValueVisitor interface implemented by this class may have methods added to it in the future to accommodate new, currently unknown, language structures added to future versions of the Java? programming language. Therefore, methods whose names begin with "visit" may be added to this class in the future; to avoid incompatibilities, classes which extend this class should not declare any instance methods with names beginning with "visit".

When such a new visit method is added, the default implementation in this class will be to call the visitUnknown method. A new abstract annotation value visitor class will also be introduced to correspond to the new language level; this visitor will have different default behavior for the visit method in question. When the new visitor is introduced, all or portions of this visitor may be deprecated.

Note that adding a default implementation of a new visit method in a visitor class will occur instead of adding a default method directly in the visitor interface since a Java SE 8 language feature cannot be used to this version of the API since this version is required to be runnable on Java SE 7 implementations. Future versions of the API that are only required to run on Java SE 8 and later may take advantage of default methods in this situation.

WARNING: The AnnotationValueVisitor interface implemented by this class may have methods added to it in the future to accommodate new, currently unknown, language structures added to future versions of the Java? programming language. Therefore, methods whose names begin with "visit" may be added to this class in the future; to avoid incompatibilities, classes which extend this class should not declare any instance methods with names beginning with "visit".

When such a new visit method is added, the default implementation in this class will be to call the visitUnknown method. A new abstract annotation value visitor class will also be introduced to correspond to the new language level; this visitor will have different default behavior for the visit method in question. When the new visitor is introduced, all or portions of this visitor may be deprecated.

Note that adding a default implementation of a new visit method in a visitor class will occur instead of adding a default method directly in the visitor interface since a Java SE 8 language feature cannot be used to this version of the API since this version is required to be runnable on Java SE 7 implementations. Future versions of the API that are only required to run on Java SE 8 and later may take advantage of default methods in this situation.