```
┌──────────────┐      ┌──────────────┐      ┌──────────────┐
│ bool         │      │ IdentifyZeros│      │TruncationKind│
├──────────────┤      ├──────────────┤      ├──────────────┤
│              │      │              │      │              │
├──────────────┤      ├──────────────┤      ├──────────────┤
│              │      │              │      │              │
└──────────────┘      └──────────────┘      └──────────────┘
```

-check_safe_integer_   -identify_zeros_   -kind_

```
┌────────────────────────────────────┐
│      v8::internal::compiler         │
│           ::Truncation              │
├────────────────────────────────────┤
│                                     │
├────────────────────────────────────┤
│ + IsUnused()                        │
│ + IsUsedAsBool()                    │
│ + IsUsedAsWord32()                  │
│ + IsUsedAsWord64()                  │
│ + TruncatesBooleanAndNullAnd        │
│ BigIntToNumber()                    │
│ + TruncatesOddballAndBigInt         │
│ ToNumber()                          │
│ + IdentifiesUndefinedAndZero()      │
│ + IdentifiesZeroAndMinusZero()      │
│ + operator==()                      │
│ + operator!=()                      │
│ + description()                     │
│ + IsLessGeneralThan()               │
│ + identify_zeros()                  │
│ + check_safe_integer()              │
│ + None()                            │
│ + Bool()                            │
│ + Word32()                          │
│ + Word32WithSafeIntCheck()          │
│ + Word64()                          │
│ + BooleanAndNullAndBigIntTo         │
│ Number()                            │
│ + OddballAndBigIntToNumber()        │
│ + Any()                             │
│ + Generalize()                      │
│ - Truncation()                      │
│ - kind()                            │
│ - Generalize()                      │
│ - GeneralizeIdentifyZeros()         │
│ - LessGeneral()                     │
│ - LessGeneralIdentifyZeros()        │
└────────────────────────────────────┘
```