```
┌─────────────────────────────────┐
│   MathExpressions::Abstract     │
│         Expression              │
├─────────────────────────────────┤
├─────────────────────────────────┤
│  +  AbstractExpression()        │
│  +  ~AbstractExpression()       │
│  +  paintDevice()               │
│  +  next()                      │
│  +  parent()                    │
│  +  font()                      │
│  +  pen()                       │
│  +  brush()                     │
│  +  color()                     │
│  +  hasNext()                   │
│     и 23 больше...              │
│  #  fontChanged()               │
│  #  colorChanged()              │
│  #  paintDeviceChanged()        │
│  #  paint()                     │
│  #  calcWidth()                 │
│  #  calcHeight()                │
│  #  calcAscent()                │
│  #  calcDescent()               │
│  #  calcSuperscriptX()          │
│  #  calcSuperscriptY()          │
│     и 9 больше...               │
│  #  assignFont()                │
│  #  assignParent()              │
│  #  assignPaintDevice()         │
└─────────────────────────────────┘
                 △
                 │
┌─────────────────────────────────┐
│   MathExpressions::Abstract     │
│        ParentExpression         │
├─────────────────────────────────┤
├─────────────────────────────────┤
│  +  AbstractParentExpression()  │
│  +  ~AbstractParentExpression() │
│  +  setSon()                    │
│  +  son()                       │
│  +  hasSon()                    │
│  +  cutOffSun()                 │
│  #  fontChanged()               │
│  #  colorChanged()              │
│  #  paintDeviceChanged()        │
│  #  updateSonFont()             │
│  #  updateSonColor()            │
└─────────────────────────────────┘
                 △
                 │
┌───────────────────────────────────┐
│    MathExpressions::Abstract      │
│      TwinParentExpression         │
├───────────────────────────────────┤
├───────────────────────────────────┤
│ + AbstractTwinParentExpression()  │
│ + ~AbstractTwinParentExpression() │
│ + firstTwin()                     │
│ + secondTwin()                    │
│ + hasFirstTwin()                  │
│ + hasSecondTwin()                 │
│ + setFirstTwin()                  │
│ + setSeconsTwin()                 │
│ # fontChanged()                   │
│ # colorChanged()                  │
│ # paintDeviceChanged()            │
│ # updateFirstTwinFont()           │
│ # updateFirstTwinColor()          │
│ # updateSecondTwinFont()          │
│ # updateSecondTwinColor()         │
└───────────────────────────────────┘
                 △
                 │
┌───────────────────────────────────┐
│  MathExpressions::GroupExpression │
├───────────────────────────────────┤
├───────────────────────────────────┤
│  +   GroupExpression()            │
│  #   fontChanged()                │
│  #   paint()                      │
│  #   calcWidth()                  │
│  #   calcHeight()                 │
│  #   calcAscent()                 │
│  #   calcDescent()                │
│  #   updateFirstTwinFont()        │
│  #   updateSecondTwinFont()       │
│  #   calcSymbolWidth()            │
│  #   calcSymbolHeight()           │
│  #   drawSymbol()                 │
│  #   symbol()                     │
│  #   smallFont()                  │
│  #   symbolFont()                 │
└───────────────────────────────────┘
                 △
                 │
┌─────────────────────────────────┐
│   MathExpressions::Integral     │
│         Expression              │
├─────────────────────────────────┤
├─────────────────────────────────┤
│  +   IntegralExpression()       │
│  #   calcSymbolWidth()          │
│  #   drawSymbol()               │
└─────────────────────────────────┘
```