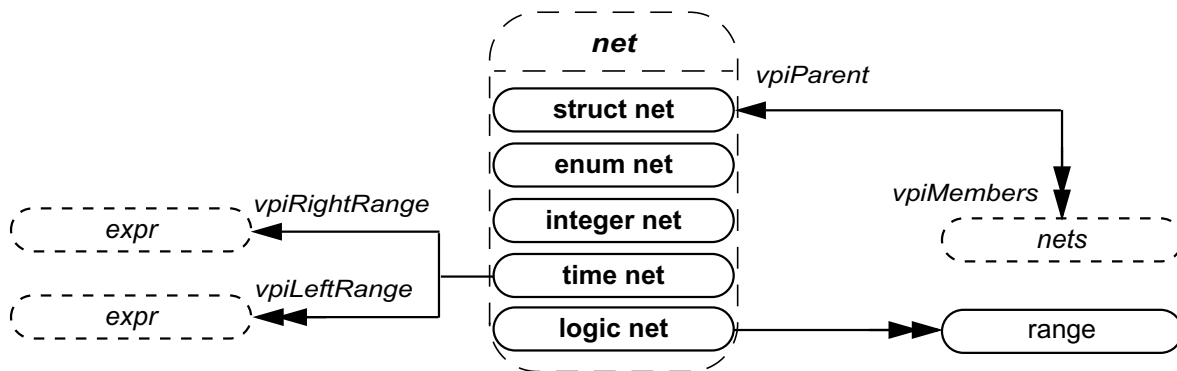


REPLACE:**32.13 Nets (supersedes P1364 26.6.6)**

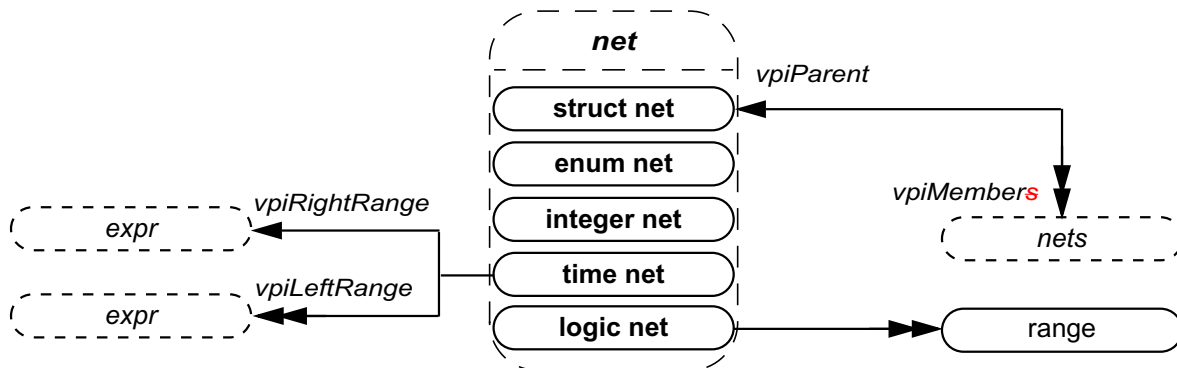
-> access by index <i>vpi_handle_by_index</i> <i>vpi_handle_by_multi_index</i>	-> net decl assign <i>bool: vpiNetDeclAssign</i>	
-> array member <i>bool: vpiArray (deprecated)</i> <i>bool: vpiArrayMember</i>	-> net type <i>int: vpiNetType</i> <i>int: vpiResolvedNetType</i>	-> strength <i>int: vpiStrength0</i> <i>int: vpiStrength1</i> <i>int: vpiChargeStrength</i>
-> delay <i>vpi_get_delays()</i>	-> scalar <i>bool: vpiScalar</i>	-> value <i>vpi_get_value()</i> <i>vpi_put_value()</i>
-> expanded <i>bool: vpiExpanded</i>	-> scaled declaration <i>bool: vpiExplicitScaled</i>	-> vector <i>bool: vpiVector</i>
-> implicitly declared <i>bool: vpiImplicitDecl</i>	-> sign <i>bool: vpiSigned</i>	-> vectored declaration <i>bool: vpiExplicitVectored</i>
-> name <i>str: vpiName</i> <i>str: vpiFullName</i>	-> size <i>int: vpiSize</i>	

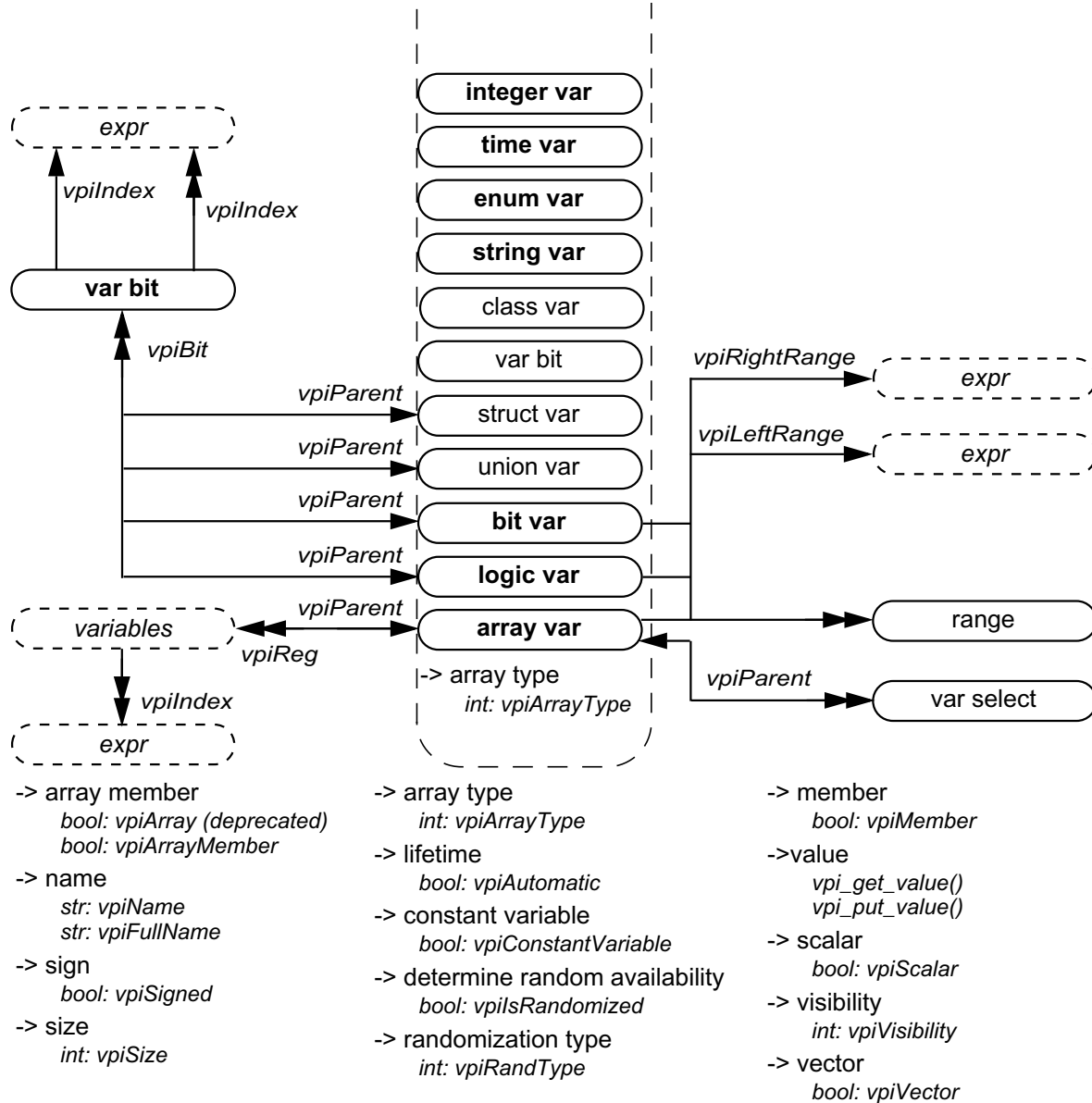


WITH:

32.13 Nets (supersedes P1364 26.6.6)

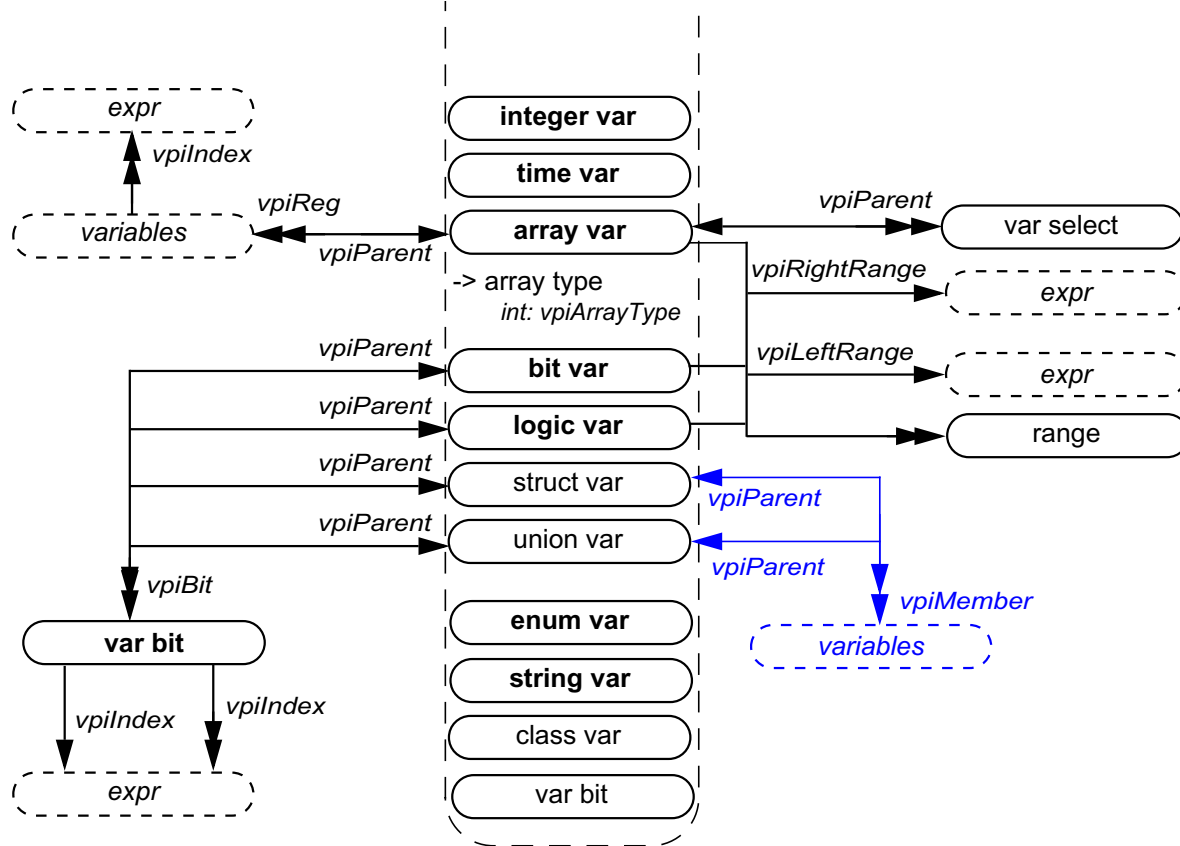
-> access by index <i>vpi_handle_by_index</i> <i>vpi_handle_by_multi_index</i>	-> net decl assign <i>bool: vpiNetDeclAssign</i>	-> strength <i>int: vpiStrength0</i> <i>int: vpiStrength1</i> <i>int: vpiChargeStrength</i>
-> array member <i>bool: vpiArray (deprecated)</i> <i>bool: vpiArrayMember</i>	-> net type <i>int: vpiNetType</i> <i>int: vpiResolvedNetType</i>	-> value <i>vpi_get_value()</i> <i>vpi_put_value()</i>
-> delay <i>vpi_get_delays()</i>	-> scalar <i>bool: vpiScalar</i>	-> vector <i>bool: vpiVector</i>
-> expanded <i>bool: vpiExpanded</i>	-> scaled declaration <i>bool: vpiExplicitScaled</i>	-> vectored declaration <i>bool: vpiExplicitVectored</i>
-> implicitly declared <i>bool: vpiImplicitDecl</i>	-> sign <i>bool: vpiSigned</i>	-> member <i>bool: vpiStructUnionMember</i>
-> name <i>str: vpiName</i> <i>str: vpiFullName</i>	-> size <i>int: vpiSize</i>	



REPLACE:**32.14 Variables (supersedes P1364 26.6.7 and 26.6.8)****NOTES**

- 16) **vpilsRandomized** is a property to determine whether a random variable is currently active for randomization.
- 17) When the **vpiMember** property is TRUE, it indicates that the variable is a member of a parent struct or union variable. See also the relations in 32.18
- 18) If a variable is an element of an array (the **vpiArrayMember** property is TRUE), the **vpiIndex** iterator shall return the indexing expressions that select that specific variable out of the array.

WITH:

32.14 Variables (supersedes P1364 26.6.7 and 26.6.8)

-> array member
 bool: *vpiArray* (deprecated)
 bool: *vpiArrayMember*

-> name
 str: *vpiName*
 str: *vpiFullName*

-> sign
 bool: *vpiSigned*

-> size
 int: *vpiSize*

-> array type
 int: *vpiArrayType*

-> lifetime
 bool: *vpiAutomatic*

-> constant variable
 bool: *vpiConstantVariable*

-> determine random availability
 bool: *vpiIsRandomized*

-> randomization type
 int: *vpiRandType*

-> member
 bool: *vpiStructUnionMember*

-> value
 vpi_get_value()
 vpi_put_value()

-> scalar
 bool: *vpiScalar*

-> visibility
 int: *vpiVisibility*

-> vector
 bool: *vpiVector*

NOTES

- 16) **vpiIsRandomized** is a property to determine whether a random variable is currently active for randomization.
- 17) When the *vpiStructUnionMember* property is TRUE, it indicates that the variable is a member of a parent struct or union variable. See also the relations in 32.18
- 18) If a variable is an element of an array (the **vpiArrayMember** property is TRUE), the **vpiIndex** iterator shall return the indexing expressions that select that specific variable out of the array.

REPLACE: Annex I

```

/***** methods used to traverse 1 to many relationships *****/
#define vpiTypedef          725
#define vpiImport           726
#define vpiDerivedClasses   727
#define vpiMethods          730
#define vpiSolveBefore      731
#define vpiSolveAfter       732
#define vpiWaitingProcesses  734
#define vpiMessages         735
#define vpiLoopVars         737
#define vpiConcurrentAssertions 740
#define vpiMatchItem        741
/***** methods both 1-1 and 1-many relations *****/
#define vpiInstance         745
/***** generic object properties *****/
/***** generic object properties *****/
#define vpiTop              600
#define vpiUnit             602
#define vpiAccessType       604
#define vpiForkJoinAcc      1
#define vpiExternAcc        2
#define vpiDPIExternAcc     3
#define vpiDPIImportAcc     4
#define vpiArrayType        606
#define vpiStaticArray      1
#define vpiDynamicArray     2
#define vpiAssocArray       3
#define vpiQueueArray       4
#define vpiArrayMember      607
#define vpiIsRandomized     608
#define vpiRandType         610
#define vpiNotRand          1
#define vpiRand             2
#define vpiRandC            3
#define vpiConstantVariable 612
#define vpiMember           615
#define vpiVisibility       620
#define vpiPublicVis        1
#define vpiProtectedVis     2
#define vpiLocalVis         3
#define vpiNullConst        622
#define vpiOneStepConst     623
#define vpiAlwaysType       624
#define vpiAlwaysComb       1
#define vpiAlwaysFF         2
#define vpiAlwaysLatch      3
#define vpiDistType         625
#define vpiEqualDist        1 /* constraint equal distribution */
#define vpiDivDist          2 /* constraint divided distribution */
#define vpiPacked           630
#define vpiTagged           632
#define vpiRef              633
#define vpiDefaultSkew      634
#define vpiVirtual          635

```

```

#define vpiUserDefined          636
#define vpiIsConstraintEnabled  638
#define vpiClassType            640
#define vpiMailboxClass         1
#define vpiSemaphoreClass       2
#define vpiUserDefineClass      3
#define vpiMethod               645
#define vpiIsClockInferred      649
#define vpiQualifier            650
#define vpiNoQualifier          0
#define vpiUniqueQualifier      1
#define vpiPriorityQualifier     2
#define vpiTaggedQualifier      4
#define vpiRandQualifier        8

```

WITH:

Annex I

```

/***** methods used to traverse 1 to many relationships *****/
#define vpiTypedef              725
#define vpiImport               726
#define vpiDerivedClasses       727
#define vpiMethods              730
#define vpiSolveBefore          731
#define vpiSolveAfter           732
#define vpiWaitingProcesses     734
#define vpiMessages             735
#define vpiLoopVars             737
#define vpiConcurrentAssertions 740
#define vpiMatchItem            741
#define vpiMember               746
/***** methods both 1-1 and 1-many relations *****/
#define vpiInstance             745
/***** generic object properties *****/
/***** generic object properties *****/
#define vpiTop                  600
#define vpiUnit                 602
#define vpiAccessType           604
#define vpiForkJoinAcc          1
#define vpiExternAcc            2
#define vpiDPIExternAcc         3
#define vpiDPIImportAcc         4
#define vpiArrayType            606
#define vpiStaticArray          1
#define vpiDynamicArray         2
#define vpiAssocArray           3
#define vpiQueueArray           4
#define vpiArrayMember          607
#define vpiIsRandomized         608
#define vpiRandType             610
#define vpiNotRand              1
#define vpiRand                 2
#define vpiRandC                3
#define vpiConstantVariable     612
#define vpiStructUnionMember    615
#define vpiVisibility           620
#define vpiPublicVis            1

```

```

#define vpiProtectedVis      2
#define vpiLocalVis          3
#define vpiNullConst         622
#define vpiOneStepConst      623
#define vpiAlwaysType        624
#define vpiAlwaysComb         1
#define vpiAlwaysFF           2
#define vpiAlwaysLatch       3
#define vpiDistType           625
#define vpiEqualDist          1 /* constraint equal distribution */
#define vpiDivDist            2 /* constraint divided distribution */
#define vpiPacked             630
#define vpiTagged             632
#define vpiRef                633
#define vpiDefaultSkew        634
#define vpiVirtual            635
#define vpiUserDefined        636
#define vpiIsConstraintEnabled 638
#define vpiClassType          640
#define vpiMailboxClass       1
#define vpiSemaphoreClass     2
#define vpiUserDefineClass    3
#define vpiMethod             645
#define vpiIsClockInferred    649
#define vpiQualifier          650
#define vpiNoQualifier        0
#define vpiUniqueQualifier    1
#define vpiPriorityQualifier   2
#define vpiTaggedQualifier    4
#define vpiRandQualifier      8

```

