Objects:

```
    mode  | adr    conval  link     scope    leaf
   ------------------------------------------------
    Undef |                                         Not used
    Var   | vadr           next              regopt Glob or loc var or proc value parameter
    VarPar| vadr           next              regopt Var parameter (vis = 0 | inPar | outPar)
    Con   |        val                              Constant
    Fld   | off            next                     Record field
    Typ   |                                         Named type
    LProc | entry  sizes   firstpar scope    leaf   Local procedure, entry adr set in back-end
    XProc | entry  sizes   firstpar scope    leaf   External procedure, entry adr set in back-end
    SProc | fno    sizes                            Standard procedure
    CProc |        code    firstpar scope           Code procedure
    IProc | entry  sizes            scope    leaf   Interrupt procedure, entry adr set in back-end
    Mod   |                         scope           Module
    Head  | txtpos         owner    firstvar        Scope anchor
    TProc | entry  sizes   firstpar scope    leaf   Bound procedure, mthno = obj.num
```

Structures:

```
    form    comp  | n      BaseTyp   link     mno  txtpos   sysflag
  ----------------------------------------------------------------------------------
    Undef   Basic |
    Byte    Basic |
    Bool    Basic |
    Char8   Basic |
    Int8    Basic |
    Int16   Basic |
    Int32   Basic |
    Real32  Basic |
    Real64  Basic |
    Set     Basic |
    String8 Basic |
    NilTyp  Basic |
    NoTyp   Basic |
    Pointer Basic |        PBaseTyp           mno  txtpos   sysflag
    ProcTyp Basic |        ResTyp    params   mno  txtpos   sysflag
    Comp    Array | nofel  ElemTyp            mno  txtpos   sysflag
    Comp    DynArr| dim    ElemTyp            mno  txtpos   sysflag
    Comp    Record| nofmth RBaseTyp  fields   mno  txtpos   sysflag
    Char16  Basic |
    String16Basic |
    Int64   Basic |

Nodes:

design   = Nvar|Nvarpar|Nfield|Nderef|Nindex|Nguard|Neguard|Ntype|Nproc.
expr     = design|Nconst|Nupto|Nmop|Ndop|Ncall.
nextexpr = NIL|expr.
ifstat   = NIL|Nif.
casestat = Ncaselse.
sglcase  = NIL|Ncasedo.
stat     = NIL|Ninittd|Nenter|Nassign|Ncall|Nifelse|Ncase|Nwhile|Nrepeat|
           Nloop|Nexit|Nreturn|Nwith|Ntrap.


              class     subcl     obj      left      right     link
              ---------------------------------------------------------
```

design	Nvar		var			nextexpr	
	Nvarpar		varpar			nextexpr	
	Nfield		field	design		nextexpr	
	Nderef	ptr/str		design		nextexpr	
	Nindex			design	expr	nextexpr	
	Nguard			design		nextexpr	(typ = guard type)
	Neguard			design		nextexpr	(typ = guard type)
	Ntype		type			nextexpr	
	Nproc	normal	proc			nextexpr	
		super	proc			nextexpr	

expr	design						
	Nconst		const				(val = node.conval)
	Nupto			expr	expr	nextexpr	
	Nmop	not		expr		nextexpr	
		minus		expr		nextexpr	
		is	tsttype	expr		nextexpr	
		conv		expr		nextexpr	
		abs		expr		nextexpr	
		cap		expr		nextexpr	
		odd		expr		nextexpr	
		bit		expr		nextexpr	{x}
		adr		expr		nextexpr	SYSTEM.ADR
		typ		expr		nextexpr	SYSTEM.TYP
		cc		Nconst		nextexpr	SYSTEM.CC
		val		expr		nextexpr	SYSTEM.VAL
	Ndop	times		expr	expr	nextexpr	
		slash		expr	expr	nextexpr	
		div		expr	expr	nextexpr	

```
                        mod                expr      expr      nextexpr
                        and                expr      expr      nextexpr
                        plus               expr      expr      nextexpr
                        minus              expr      expr      nextexpr
                        or                 expr      expr      nextexpr
                        eql                expr      expr      nextexpr
                        neq                expr      expr      nextexpr
                        lss                expr      expr      nextexpr
                        leq                expr      expr      nextexpr
                        grt                expr      expr      nextexpr
                        geq                expr      expr      nextexpr
                        in                 expr      expr      nextexpr
                        ash                expr      expr      nextexpr
                        msk                expr      Nconst    nextexpr
                        len                design    Nconst    nextexpr
                        min                expr      expr      nextexpr MIN
                        max                expr      expr      nextexpr MAX
                        bit                expr      expr      nextexpr SYSTEM.BIT
                        lsh                expr      expr      nextexpr SYSTEM.LSH
                        rot                expr      expr      nextexpr SYSTEM.ROT
              Ncall               fpar     design    nextexpr  nextexpr
              Ncomp                        stat      expr      nextexpr

nextexpr      NIL
              expr

ifstat        NIL
              Nif                          expr      stat      ifstat

casestat      Ncaselse                     sglcase   stat          (minmax = node.conval)
```

sglcase	NIL						
	Ncasedo			Nconst	stat	sglcase	
stat	NIL						
	Ninittd					stat	(of node.typ)
	Nenter		proc	stat	stat	stat	(proc=NIL for mod)
	Nassign	assign		design	expr	stat	
		newfn		design	nextexp	stat	
		incfn		design	expr	stat	
		decfn		design	expr	stat	
		inclfn		design	expr	stat	
		exclfn		design	expr	stat	
		copyfn		design	expr	stat	
		getfn		design	expr	stat	SYSTEM.GET
		putfn		expr	expr	stat	SYSTEM.PUT
		getrfn		design	Nconst	stat	SYSTEM.GETREG
		putrfn		Nconst	expr	stat	SYSTEM.PUTREG
		sysnewfn		design	expr	stat	SYSTEM.NEW
		movefn		expr	expr	stat	SYSTEM.MOVE (right.link = 3rd par)
	Ncall		fpar	design	nextexpr	stat	
	Nifelse			ifstat	stat	stat	
	Ncase			expr	casestat	stat	
	Nwhile			expr	stat	stat	
	Nrepeat			stat	expr	stat	
	Nloop			stat		stat	
	Nexit					stat	
	Nreturn		proc	nextexpr		stat	(proc = NIL for mod)
	Nwith			ifstat	stat	stat	

Ntrap		expr	stat
Ncomp	stat	stat	stat