

# Aspic Grammar (Fast variant)

Laure Gonnord, university of Lille

2007-2012

**Aspic vs Fast** The input format for Aspic (<http://laure.gonnord.org/pro/aspic/aspic.html>) is a minor variation of the input format of the Fast tool (<http://www.lsv.ens-cachan.fr/Software/fast/>).

```
prog: model strat
```

```
/*MODEL*/
```

```
model : TK_MODEL id TK_LBRACE vars states transitions TK_RBRACE
```

```
vars : TK_VAR idlist TK_SEMI
```

```
id:  
TK_ID
```

```
idlist:  
id | idlist TK_COMMA id
```

```
states : TK_STATES idlist TK_SEMI
```

```
transitions : translist
```

```
translist :  
trans | translist trans
```

```
trans : TK_TRANS id TK_ASSIGN TK_LBRACE intrans TK_RBRACE TK_SEMI
```

```
intrans : TK_FROM TK_ASSIGN id TK_SEMI TK_TO TK_ASSIGN id TK_SEMI  
          TK_GUARD TK_ASSIGN guard TK_SEMI  
          TK_ACTION TK_ASSIGN action TK_SEMI
```

```
action :  
| /*do nothing */  
| affectation  
| action TK_COMMA affectation
```

```
affectation :  
|id TK_PRIME TK_EQUAL TK_INDET /*x:=?*/  
| id TK_PRIME TK_EQUAL numexpr
```

```

/*STRATEGY*/

strat : TK_STRATEGY id TK_LBRACE instructionlist TK_RBRACE

instructionlist :
instruction | instructionlist instruction

instruction :
TK_REGION TK_REGINIT TK_ASSIGN region TK_SEMI
| TK_REGION TK_REGBAD TK_ASSIGN region TK_SEMI
| TK_REGION id TK_ASSIGN region TK_SEMI
| TK_STRATTRANS id TK_ASSIGN idlistbraq TK_SEMI
| TK_BOOLEAN id TK_ASSIGN boolfastexpr TK_SEMI /*ignore*/
| TK_SETMAXSTATE TK_LPAREN integ TK_RPAREN TK_SEMI /*ignore*/
| TK_SETMAXACC TK_LPAREN integ TK_RPAREN TK_SEMI /*ignore*/
| TK_PRINT TK_LPAREN chaine TK_RPAREN TK_SEMI /*ignore*/
| TK_IF boolfastexpr TK_THEN instructionlist suite /*ignore*/

suite :
TK_ENDIF
| TK_ELSE instructionlist TK_ENDIF

chaine :
id | TK_DOUBLEQUOTE idandspace TK_DOUBLEQUOTE

idandspace :
id | id idandspace

idlistbraq :
id | TK_LBRACE idlist TK_RBRACE

integ:
| TK_NUM

region :
regionterm /*terminal*/
| region TK_ANDAND region
| region TK_OROR region
| TK_EXCLA region
| TK_LPAREN region TK_RPAREN
| TK_PRE TK_MUL TK_LPAREN region TK_COMMA idlist TK_RPAREN /*pre
star, ignore*/
| TK_PRE TK_MUL TK_LPAREN region TK_COMMA idlist TK_COMMA integ TK_RPAREN /*ignore*/
| TK_POST TK_MUL TK_LPAREN region TK_COMMA idlist TK_RPAREN /*ignore*/
| TK_POST TK_MUL TK_LPAREN region TK_COMMA idlist TK_COMMA integ TK_RPAREN /*ignore*/

regionterm :
id | TK_LBRACE guard TK_RBRACE

```

```

boolfastexpr :
| id | TK_TRUE | TK_FALSE
| TK_STRATSUBSET TK_LPAREN region TK_COMMA region TK_RPAREN
| TK_STRATEQSET TK_LPAREN region TK_COMMA region TK_RPAREN
| TK_STRATISEMPTY TK_LPAREN region TK_RPAREN
| boolfastexpr TK_AND boolfastexpr
| boolfastexpr TK_OR boolfastexpr
| TK_EXCLA TK_LPAREN boolfastexpr TK_RPAREN
| TK_LPAREN boolfastexpr TK_RPAREN

/*guards are formula over numerical constraints*/
/*non primed formula*/
guard :
| TK_LPAREN guard TK_RPAREN
| guard TK_ANDAND guard
| guard TK_OROR guard
| guard TK_IMPLY guard
| TK_EXCLA TK_LPAREN guard TK_RPAREN
| term

term:
| TK_TRUE | TK_FALSE
| TK_STATE TK_EQUAL id /*bad state*/
| exprcompare

exprcompare:
| numexpr TK_GREATEREQUAL numexpr
| numexpr TK_GREATER numexpr
| numexpr TK_LESSEQUAL numexpr
| numexpr TK_LESS numexpr
| numexpr TK_EQUAL numexpr
| numexpr TK_NOTEQUAL numexpr

numexpr : /*numerical expressions*/
expr6

expr6:
expr7
| expr6 TK_PLUS expr7
| expr6 TK_MINUS expr7
expr7:
expr8
| expr7 TK_MUL expr8
| expr7 TK_DIV expr8
expr8:
| numcst
| id
| numcst id
| TK_LPAREN expr6 TK_RPAREN
| TK_MINUS expr8

```

```
numcst:
    TK_NUM | TK_RAT
```

## Token defs

```
(*keywords*)
| "model"          { TK_MODEL  }
| "var"           { TK_VAR    }
| "states"        { TK_STATES  }
| "transition"    { TK_TRANS   }
| "from"          { TK_FROM    }
| "to"            { TK_TO      }
| "guard"         { TK_GUARD   }
| "action"        { TK_ACTION   }
| "exists"        { TK_EXISTS   }
| "pre"           { TK_PRE     }
| "post"          { TK_POST    }
| "false"         { TK_FALSE   }
| "true"          { TK_TRUE    }
| "strategy"      { TK_STRATEGY }
| "Region"        { TK_REGION  }
| "init"          { TK_REGINIT  }
| "bad"           { TK_REGBAD  }
| "state"         { TK_STATE   }
| "Transitions"   { TK_STRATTRANS }
| "boolean"       { TK_BOOLEAN }
| "True"          { TK_STRATTRUE }
| "False"         { TK_STRATFALSE }
| "eqSet"         { TK_STRATSUBSET }
| "subSet"        { TK_STRATEQSET }
| "isEmpty"       { TK_STRATISEMPTY }
| "setMaxState"   { TK_SETMAXSTATE }
| "setMaxAcc"     { TK_SETMAXACC }
| "print"         { TK_PRINT   }
| "if"            { TK_IF     }
| "endif"         { TK_ENDIF  }
| "then"          { TK_THEN   }
| "else"          { TK_ELSE   }
| "and"           { TK_AND    }
| "or"            { TK_OR     }
```

```
(* Delimiters *)
| "," { TK_COMMA }
| ";" { TK_SEMI }
| ''' { TK_PRIME }
| "(" { TK_LPAREN }
| ")" { TK_RPAREN }
| "{" { TK_LBRACE }
| "}" { TK_RBRACE }
| "[" { TK_LBRACKET }
```

```

| "]" { TK_RBRACKET }
| ":" { TK_COLON }
| "\"" { TK_DOUBLEQUOTE }

(* Arithmetic operators *)
| "+" { TK_PLUS }
| "-" { TK_MINUS }
| "*" { TK_MUL }

(* Comparison functions *)
| ">" { TK_GREATER }
| "<" { TK_LESS }
| ">=" { TK_GREATEREQUAL }
| "<=" { TK_LESSEQUAL }
| "=" { TK_EQUAL }
| "!=" { TK_NOTEQUAL }

(*assignment*)
| ":=" { TK_ASSIGN }

(*boolean operators*)
| "&&" {TK_ANDAND}
| "||" {TK_OROR}
| "!" {TK_EXCLA}
| "=>" {TK_IMPLY}
| "<=>" {TK_EQUIV}

(*indet*)
| "?" {TK_INDET}

```