

Execution tokens, Semantics, etc.

M. Anton Ertl, TU Wien

# Concepts

execution token

named word

`:noname`

execution semantics

interpretation semantics

compilation semantics

text interpreter

`immediate`

`execute`

`compile,`

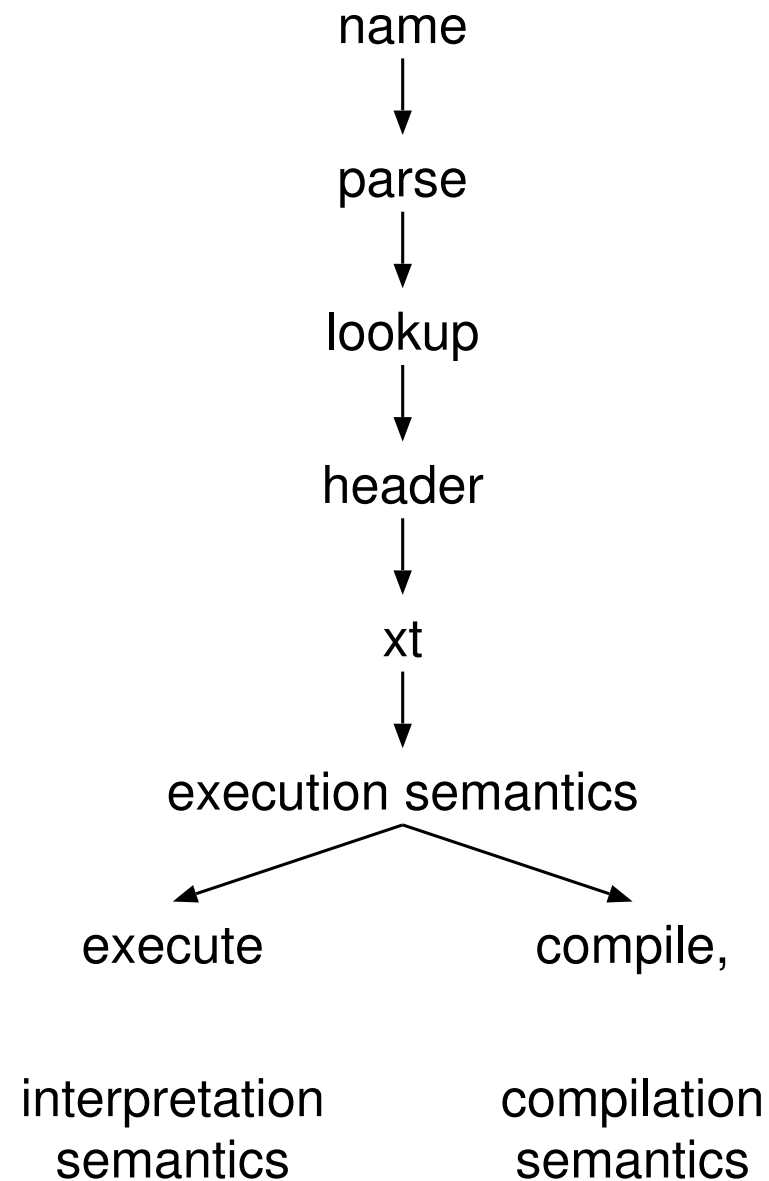
`postpone`

`, [']`

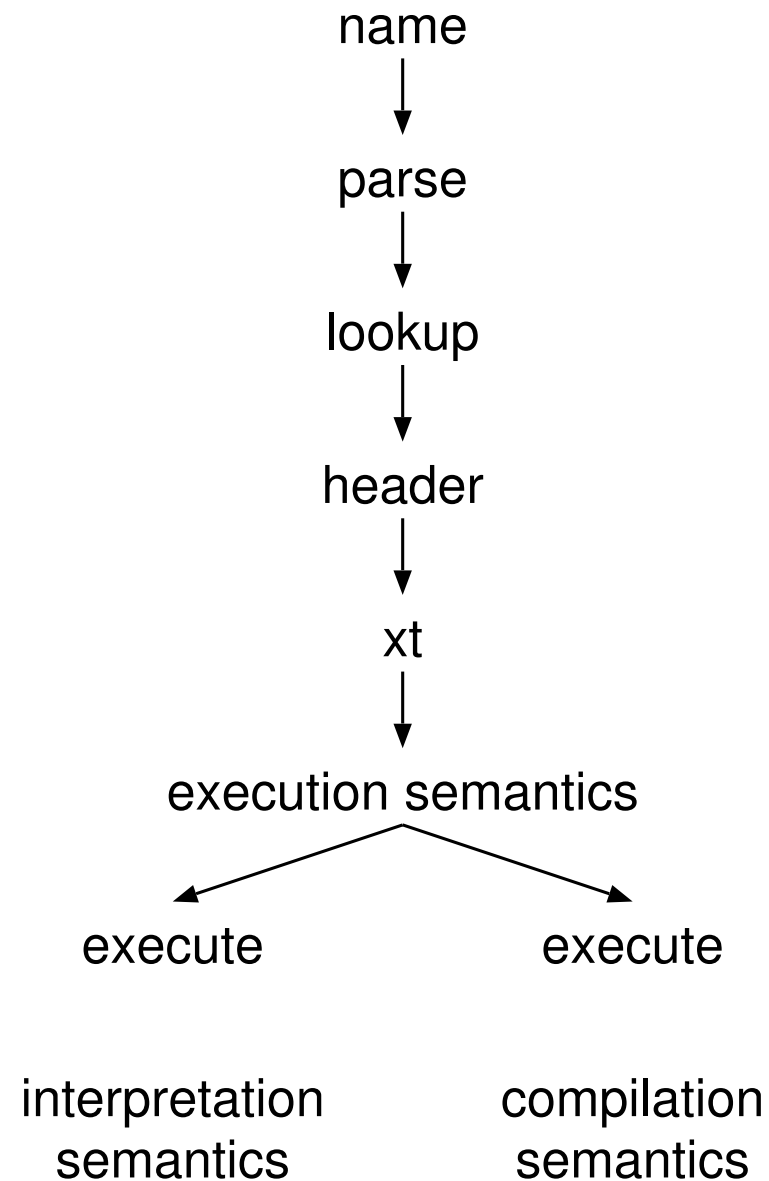
## User-defined text interpreter

```
...                               ( c-addr )
FIND ?DUP IF                     ( xt +-1 )
    STATE @ IF                   ( xt +-1 )
    0> IF EXECUTE ELSE COMPILE, THEN ( ??? )
    ELSE                         ( xt +-1 )
        DROP EXECUTE            ( ??? )
    THEN
ELSE                               ( c-addr )
    ( whatever you do for an undefined word )
THEN
...
```

# Normal words, traditional implementation



# Immediate words, traditional implementation



S'' , Forth-94

name

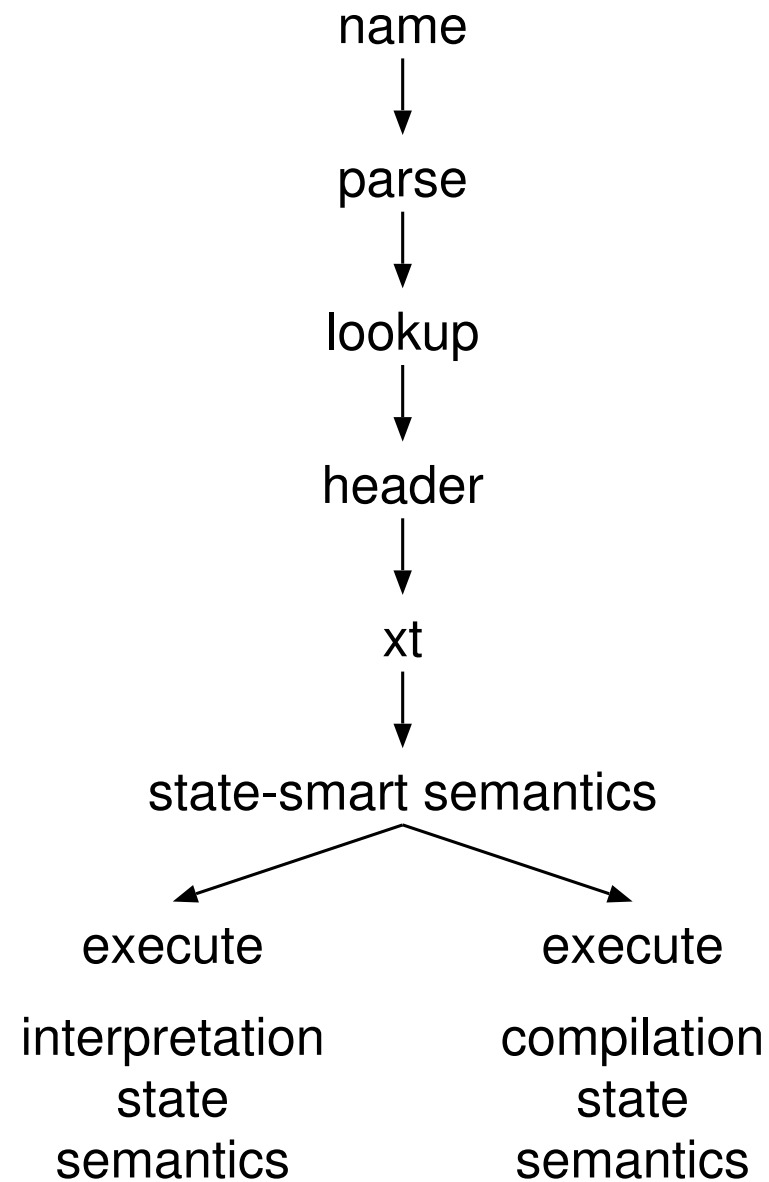
?

~~execution semantics~~

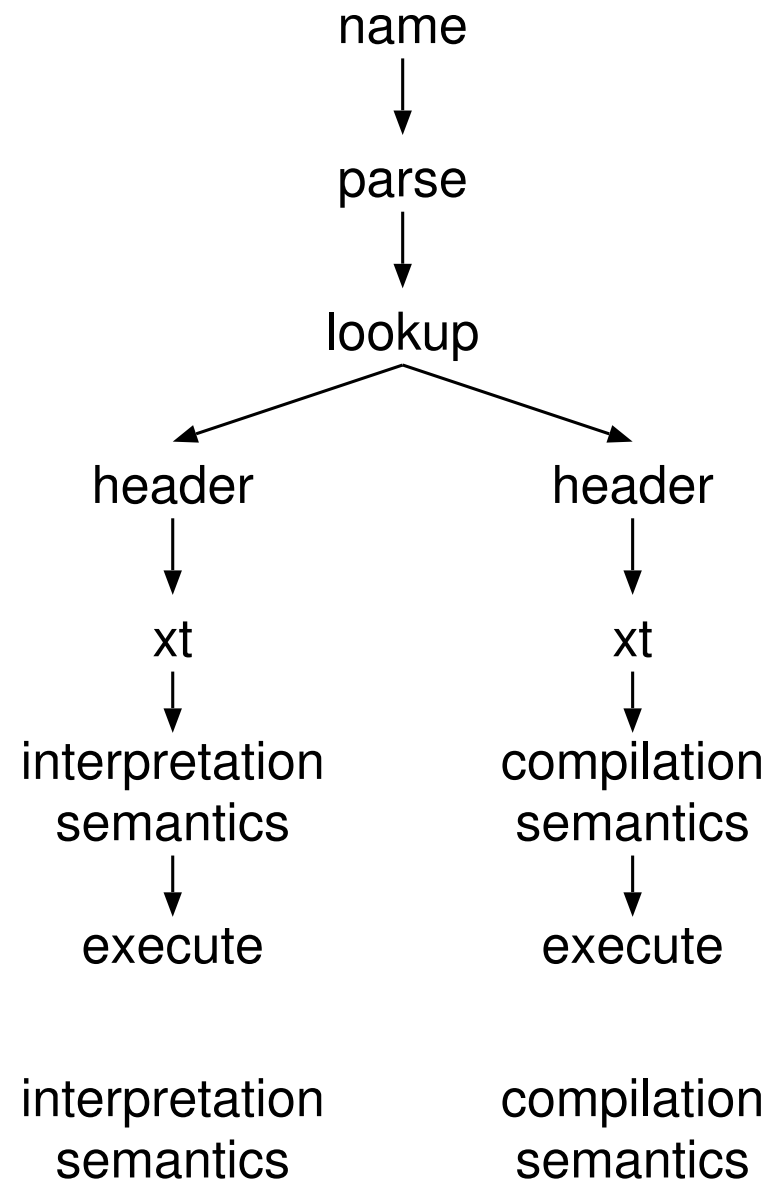
interpretation  
semantics

compilation  
semantics

$S''$ , state-smart

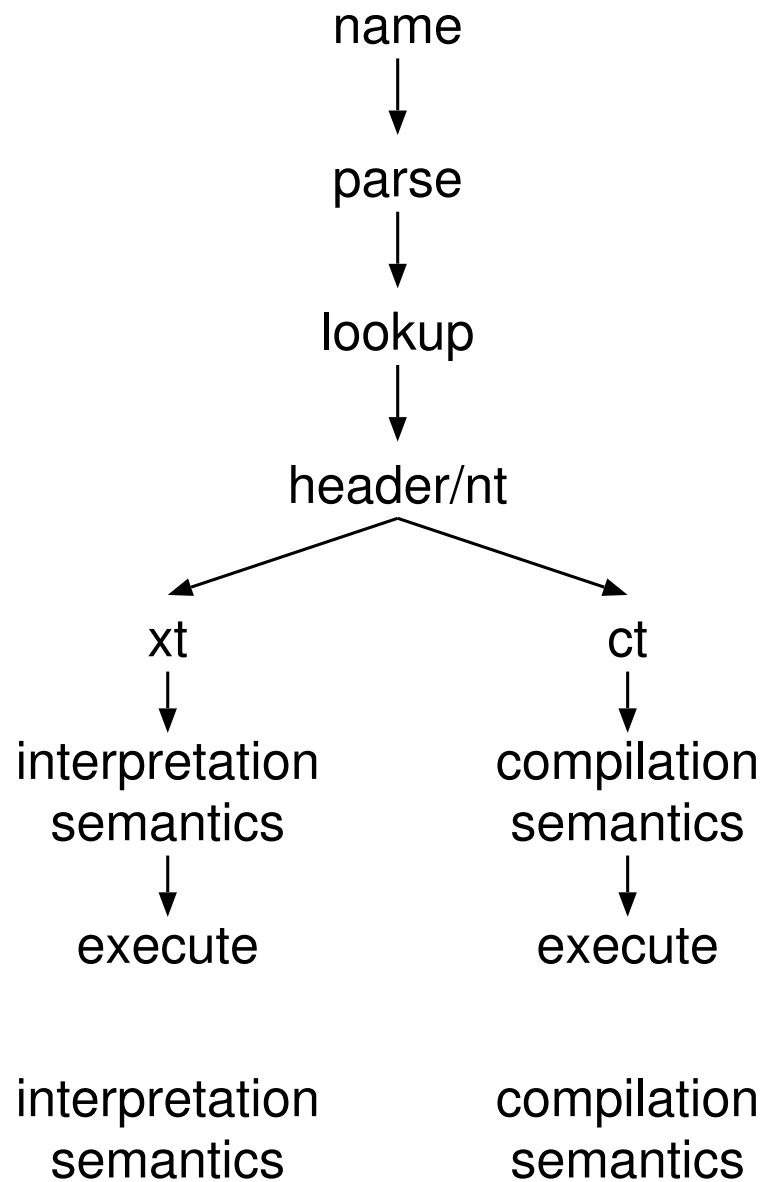


S'', cmForth





S'' , Gforth



$S''$ , VFX

name



parse



lookup



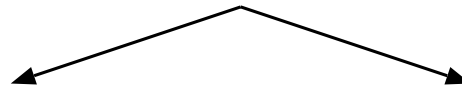
header



xt



? semantics



execute

compile,

interpretation  
semantics

compilation  
semantics

## Problems

- Text interpreter works on all systems  
includes user-defined text interpreter
- Problems with postpone ' [']

```
: ]execute[ ] execute postpone [ ;  
' S" ]execute[ bla" type  
\ should type "bla", breaks on state-smart
```

```
: foo [ ' S" compile, ] ;  
foo bla" type  
\ should type "bla", breaks on VFX
```

# Problems

- compile, should be equivalent to
- : compile, postpone literal postpone execute ;

```
: defer
  create ['] abort ,
does>
  @ execute ;
: defers
  ' >body @ compile, ; immediate
```

```
defer foo ' bar is foo
:noname defers foo ... ; is foo
```

\ From Mini-00F

```
: method ( m v -- m' v ) Create over , swap cell+ swap
DOES> ( ... o -- ... ) @ over @ + @ execute ;
: :: ( class "name" -- ) ' >body @ + @ compile, ;
```

## Solutions?

- Disallow ticking and postponing words like `s"`
- Use cmForth or Gforth implementation
- Get rid of words like `s"`  $\Rightarrow$  recognizers