

File created: 27-Jan-2024 15:50:00 {DSK}<home>larry>il>loops>system>LOOPSSPEEDUP.;2

edit by: lmm

changes to: (MACROS \GETDTD)

previous date: 11-Mar-2022 19:19:08 {DSK}<home>larry>il>loops>system>LOOPSSPEEDUP.;1

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

(RPAQQ LOOPSSPEEDUPCOMS
(

;;; Need the following low-level stuff to make a datatype non-garbage-collectable. This saves us the effort of maintaining reference counts

```
(FNS Make-Not-Reference-Counted)
(DECLARE%: EVAL@COMPILE DONTCOPY (FILES (LOADCOMP)
                                         LLGC LLDATATYPE)
          (EXPORT (MACROS \GETDTD)))
```

;;; Datatype for in-line caches

```
(CONSTANTS (\IN-LINE-CACHE-BLOCK-SIZE 100))
(RECORDS \IN-LINE-CACHE-BLOCK)
(P (Make-Not-Reference-Counted '\IN-LINE-CACHE-BLOCK))
```

;;; IV Lookup cache stuff

```
(FNS FlushIVIndexCache \Make-IV-Cache-Entry)
[INITVARS (*IV-Cache-Block* (CREATE \IN-LINE-CACHE-BLOCK))
  (*IV-Cache-Index* 0)
  (*All-IV-Cache-Block* (LIST *IV-Cache-Block*))
  (*Global-IV-Cache-Block* (\ALLOCBLOCK (TIMES 4 1024)
    (GLOBALVARS *IV-Cache-Block* *IV-Cache-Index* *All-IV-Cache-Block* *Global-IV-Cache-Block*)
    (ADDVARS (GLOBALVARS *Global-IV-Cache-Block*))
```

;;; Method lookup caching stuff

```
(FNS FlushMethodCache \Make-Method-Cache-Entry)
[INITVARS (*Method-Cache-Block* (CREATE \IN-LINE-CACHE-BLOCK))
  (*Method-Cache-Index* 0)
  (*All-Method-Cache-Block* (LIST *Method-Cache-Block*))
  (*Global-Method-Cache* (\ALLOCBLOCK (TIMES 4 1024)
    (GLOBALVARS *Method-Cache-Index* *Method-Cache-Block* *All-Method-Cache-Block* *Global-Method-Cache*)
    (ADDVARS (GLOBALVARS *Global-Method-Cache*))
```

;;; Need the following low-level stuff to make a datatype non-garbage-collectable. This saves us the effort of maintaining reference counts

(DEFINEQ

(Make-Not-Reference-Counted

[LAMBDA (TYPENAME)

; Edited 9-Jun-87 07:09 by sml

;;; Make datatype TYPENAME not ref counted. Must NOT have allocated any yet!

; Thanks to bvm for this magic

```
(LET* ((DTD (\GETDTD (\TYPENUMBERFROMNAME TYPENAME)))
      (BITS (LOGOR 32768 (fetch DTDTYPEENTRY of DTD)))
      NEW)
  (COND
    ((NOT (EQP BITS (fetch DTDTYPEENTRY of DTD)))
     (change (fetch DTDTYPEENTRY of DTD)
              BITS)
     (SETQ NEW (fetch DTDFREE of DTD))
     (COND
      (NEW (\MAKEMDSENTRY (IPLUS (LLSH (\HILOC NEW)
                                     8)
                                (LRSH (\LOLOC NEW)
                                       8))
              BITS)))
```

; Type is ref-counted, so need to change it

; Have to fix type of already allocated page, if any

)

(DECLARE%: EVAL@COMPILE DONTCOPY

(FILESLOAD (LOADCOMP)
 LLGC LLDATATYPE)

;; FOLLOWING DEFINITIONS EXPORTED

```
{MEDLEY}<loops>system>LOOPSSPEEDUP.;1
```

```
(DECLARE%: EVAL@COMPILE
```

```
(PUTPROPS GETDTD MACRO ((typeNum)
  (ADDBASE \DTDSpaceBase (ITIMES typeNum 18))))
)
)
```

```
:: END EXPORTED DEFINITIONS
```

```
::: Datatype for in-line caches
```

```
(DECLARE%: EVAL@COMPILE
```

```
(RPAQQ \IN-LINE-CACHE-BLOCK-SIZE 100)
```

```
(CONSTANTS (\IN-LINE-CACHE-BLOCK-SIZE 100))
)
```

```
(DECLARE%: EVAL@COMPILE
```

```
(DATATYPE \IN-LINE-CACHE-BLOCK
  (NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL
    NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL
    NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL
    NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL))
)
```

```
(/DECLAREDATATYPE '\IN-LINE-CACHE-BLOCK
  '(POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
    POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
    POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
    POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
    POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
    POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
    POINTER POINTER POINTER)
  ;; ---field descriptor list elided by lister---
  '200)
```

```
(Make-Not-Reference-Counted '\IN-LINE-CACHE-BLOCK)
```

```
::: IV Lookup cache stuff
```

```
(DEFINEQ
```

```
(FlushIVIndexCache
```

```
[LAMBDA NIL
```

```
; Edited 11-Jun-87 12:46 by sml
```

```
::: Wipe out the IVIndex cache
```

```
[if *Global-IV-Cache-Block*
  then (\ZEROWORDS *Global-IV-Cache-Block* (\ADDBASE *Global-IV-Cache-Block*
    (SUB1 (TIMES 2 (TIMES 4 1024))
    (for block in *All-IV-Cache-Block* when block do (\ZEROWORDS block (\ADDBASE block (SUB1 (TIMES 2
      \IN-LINE-CACHE-BLOCK-SIZE
      ]))
    ]))
```

```
(Make-IV-Cache-Entry
```

```
[LAMBDA NIL
```

```
; Edited 11-Jun-87 12:46 by sml
```

```
::: Return a new method cache entry position
```

```
(UNINTERRUPTABLY
  (LET (CACHE-BLOCK)
    (SETQ CACHE-BLOCK (\ADDBASE *IV-Cache-Block* *IV-Cache-Index*))
    (add *IV-Cache-Index* 4)
    ;; If we have used up the current block, allocate another one
    (if (NOT (LESSP *IV-Cache-Index* (TIMES 2 \IN-LINE-CACHE-BLOCK-SIZE)))
      then (SETQ *IV-Cache-Block* (CREATE \IN-LINE-CACHE-BLOCK))
        (push *All-IV-Cache-Block* *IV-Cache-Block*)
        (SETQ *IV-Cache-Index* 0))
    CACHE-BLOCK)))
```

```
(RPAQ? *IV-Cache-Block* (CREATE \IN-LINE-CACHE-BLOCK))
```

```
(RPAQ? *IV-Cache-Index* 0)
```

```
(RPAQ? *All-IV-Cache-Block* (LIST *IV-Cache-Block*))
```

```
(RPAQ? *Global-IV-Cache-Block* (\ALLOCBLOCK (TIMES 4 1024)))
```

```
{MEDLEY}<loops>system>LOOPSSPEEDUP.;1
```

```
(DECLARE%: DOEVAL@COMPILE DONTCOPY
```

```
(GLOBALVARS *IV-Cache-Block* *IV-Cache-Index* *All-IV-Cache-Block* *Global-IV-Cache-Block*)
)
```

```
(ADDTOVAR GLOBALVARS *Global-IV-Cache-Block*)
```

```
::: Method lookup caching stuff
```

```
(DEFINEQ
```

```
(FlushMethodCache
```

```
  [LAMBDA NIL
```

```
    ; Edited 11-Jun-87 12:46 by sml
```

```
::: A method has changed, so clear out any cached information about messages.
```

```
  (CLEARCLISPARRAY '_Super
```

```
    'MACROS)
```

```
  (CLEARCLISPARRAY '(NIL _SuperFringe)
```

```
    'MACROS)
```

```
  [AND *Global-Method-Cache* (\ZEROWORDS *Global-Method-Cache* (\ADDBASE *Global-Method-Cache*
```

```
    (SUB1 (TIMES 2 (TIMES 4 1024]
```

```
  (for block in *All-Method-Cache-Block* when block do (\ZEROWORDS block (\ADDBASE block
```

```
    (SUB1 (TIMES 2
```

```
      \IN-LINE-CACHE-BLOCK-SIZE
```

```
    ])
```

```
(Make-Method-Cache-Entry
```

```
  [LAMBDA NIL
```

```
    ; Edited 11-Jun-87 12:45 by sml
```

```
::: Return a new method cache entry position
```

```
  (UNINTERRUPTABLY
```

```
    (LET (CACHE-BLOCK)
```

```
      (SETQ CACHE-BLOCK (\ADDBASE *Method-Cache-Block* *Method-Cache-Index*))
```

```
      (add *Method-Cache-Index* 4)
```

```
      ;; If we have run out, allocate another big block
```

```
      (if (NOT (LESSP *Method-Cache-Index* (TIMES 2 \IN-LINE-CACHE-BLOCK-SIZE)))
```

```
          then (SETQ *Method-Cache-Block* (CREATE \IN-LINE-CACHE-BLOCK))
```

```
              (push *All-Method-Cache-Block* *Method-Cache-Block*))
```

```
              (SETQ *Method-Cache-Index* 0))
```

```
      CACHE-BLOCK)))
```

```
)
```

```
(RPAQ? *Method-Cache-Block* (CREATE \IN-LINE-CACHE-BLOCK))
```

```
(RPAQ? *Method-Cache-Index* 0)
```

```
(RPAQ? *All-Method-Cache-Block* (LIST *Method-Cache-Block*))
```

```
(RPAQ? *Global-Method-Cache* (\ALLOCBLOCK (TIMES 4 1024)))
```

```
(DECLARE%: DOEVAL@COMPILE DONTCOPY
```

```
(GLOBALVARS *Method-Cache-Index* *Method-Cache-Block* *All-Method-Cache-Block* *Global-Method-Cache*)
```

```
)
```

```
(ADDTOVAR GLOBALVARS *Global-Method-Cache*)
```

FUNCTION INDEX

FlushIVIndexCache2 Make-Not-Reference-Counted1 \Make-Method-Cache-Entry3
FlushMethodCache3 \Make-IV-Cache-Entry2

VARIABLE INDEX

All-IV-Cache-Block2 *Global-IV-Cache-Block* .2 *IV-Cache-Block*2 *Method-Cache-Block*3
All-Method-Cache-Block 3 *Global-Method-Cache* ...3 *IV-Cache-Index*2 *Method-Cache-Index*3

RECORD INDEX

\IN-LINE-CACHE-BLOCK2

CONSTANT INDEX

\IN-LINE-CACHE-BLOCK-SIZE2

MACRO INDEX

\GETDTD2
