typedef enum TeTypeCode {

  TC\_REF\_TOMBSTONE          = 0x0,

  TC\_REF\_INT32              = 0x1, // 32-bit signed

  TC\_REF\_FLOAT64            = 0x2, // 64-bit float

  TC\_REF\_STRING             = 0x3,

  TC\_REF\_INTERNED\_STRING    = 0x4,

  TC\_REF\_FUNCTION           = 0x5, // TsBytecodeFunc

  TC\_REF\_HOST\_FUNC          = 0x6, // TsHostFunc

  TC\_REF\_UINT8\_ARRAY        = 0x7, // Byte buffer

  TC\_REF\_SYMBOL             = 0x8, // Reserved: Symbol

  /\* ------------ Container types ------------------ \*/

  TC\_REF\_DIVIDER\_CONTAINER\_TYPES,  // <--- Marker

  TC\_REF\_CLASS              = 0x9, // TsClass

  TC\_REF\_VIRTUAL            = 0xA, // Reserved: TsVirtual

  TC\_REF\_RESERVED\_1         = 0xB, // Reserved

  TC\_REF\_PROPERTY\_LIST      = 0xC, // TsPropertyList

  TC\_REF\_ARRAY              = 0xD, // TsArray

  TC\_REF\_FIXED\_LENGTH\_ARRAY = 0xE, // TsFixedLengthArray

  TC\_REF\_CLOSURE            = 0xF, // TsClosure

  /\* -------------- Value types -------------------- \*/

  TC\_VAL\_INT14              = 0x10,

  TC\_VAL\_UNDEFINED          = 0x11,

  TC\_VAL\_NULL               = 0x12,

  TC\_VAL\_TRUE               = 0x13,

  TC\_VAL\_FALSE              = 0x14,

  TC\_VAL\_NAN                = 0x15,

  TC\_VAL\_NEG\_ZERO           = 0x16,

  TC\_VAL\_DELETED            = 0x17,

  TC\_VAL\_STR\_LENGTH         = 0x18, // The string "length"

  TC\_VAL\_STR\_PROTO          = 0x19, // The string "\_\_proto\_\_"

  TC\_END,

} TeTypeCode;

typedef enum vm\_TeWellKnownValues {

  VM\_VALUE\_UNDEFINED     = 0x0001,

  VM\_VALUE\_NULL          = 0x0005,

  VM\_VALUE\_TRUE          = 0x0009,

  VM\_VALUE\_FALSE         = 0x000D,

  VM\_VALUE\_NAN           = 0x0011,

  VM\_VALUE\_NEG\_ZERO      = 0x0015,

  VM\_VALUE\_DELETED       = 0X0019,

  VM\_VALUE\_STR\_LENGTH    = 0X001D,

  VM\_VALUE\_STR\_PROTO     = 0x0021,

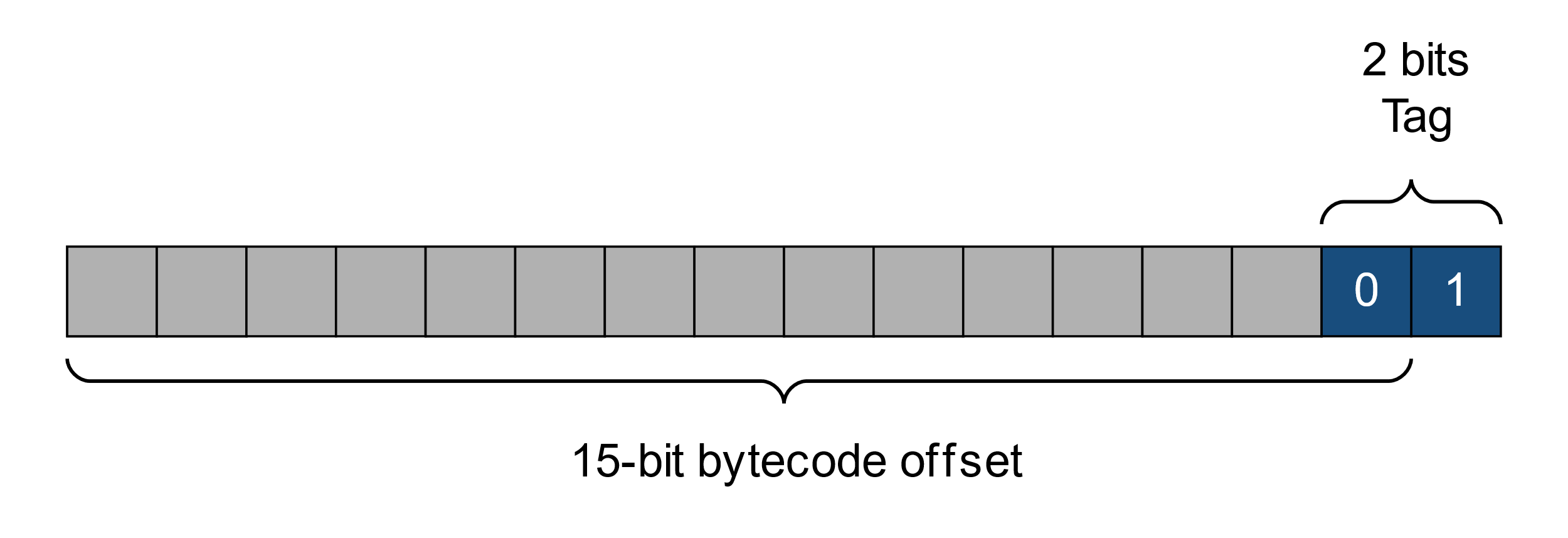
  VM\_VALUE\_WELLKNOWN\_END,

} vm\_TeWellKnownValues;

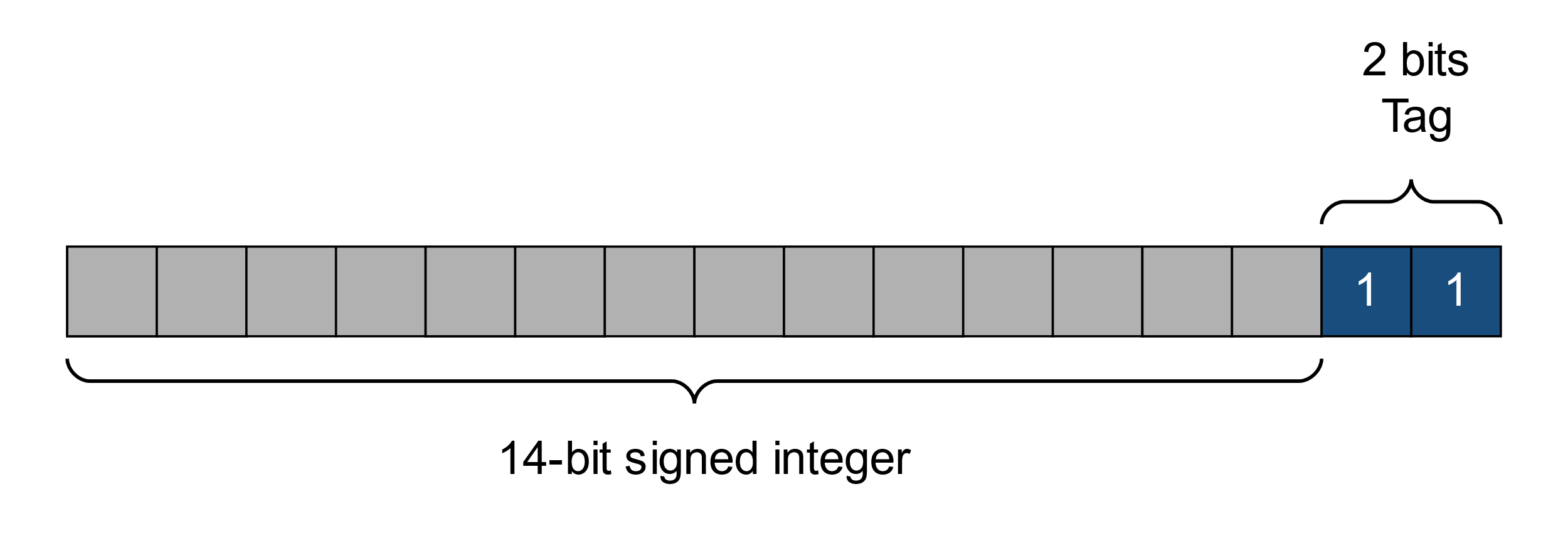
# Short Pointer

# 

# Bytecode Pointer



# Int14



# Header word

