sound_dtc

COLLABORATORS

	TITLE :		
	sound_dtc		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		April 14, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

sound_dtc

1.1 sound_dtc.doc

```
sound.datatype()
```

1.2 sound.datatype/sound.datatype

```
   NAME
sound.datatype -- root data type for sounds.

   FUNCTION
The sound.datatype is the super-class for any sound related
classes.

   METHODS
OM_NEW -- Create a new sound object.

OM_GET -- Obtain the value of an attribute.

OM_SET -- Set the values of multiple attributes.

OM_UPDATE -- Update the values of multiple attributes.

OM_DISPOSE -- Dispose of a sound object.

GM_LAYOUT -- Layout the object and notify the application of the
    title and size.

GM_HITTEST -- Determine if the object has been hit with the
    mouse.

GM_HELPTEST -- Determine if the mouse pointer is over the object.

GM_GOACTIVE -- Tell the object to go active.  On SELECTDOWN, the
    sound will start playing.

GM_HANDLEINPUT -- Handle input.  Currently input (other than
```

SELECTDOWN) doesn't affect the sound.

GM_RENDER -- Cause the graphic to render.

GM_DOMAIN -- Ask the object about its min/nom/max size.

DTM_TRIGGER -- Cause an event to occur. Currently the only
    trigger events are STM_PLAY, STM_STOP and STM_PAUSE.

DTM_COPY -- Copy the entire sound to the clipboard as 8SVX.
        Starting v41 the datatype will create a 16sv or AIFF sample
    when a 16bit sample has been passed to it.

DTM_WRITE -- Write the entire sound to a file as 8SVX. See also
    DTM_COPY.

DTM_OBTAINDRAWINFO -- Get draw handle for DTM_DRAW. One of the
    following tags must be passed to the method:

  PDTA_Screen
  GA_DrawInfo

DTM_DRAW -- Draw the object into the specified rastport. The tapedeck
    controls won't be drawn (no GadgetInfo available).

DTM_RELEASEDRAWINFO -- Free handle obtained via DTM_OBTAINDRAWINFO.

DTM_SELECTED -- Calculate start- and endsample of the marked area.
    NOTE: Do not assume anything about the objects' selectbox!

DTM_CLEARSELECTED -- Clear previuously marked area.

  TAGS
SDTA_VoiceHeader (struct VoiceHeader *) -- Set and get the base
    information for the sound. VoiceHeader is defined in
    <datatypes/soundclass.h>.
    NOTE: sound.datatype totally ignores the information provided
    by the VoiceHeader. It's saved on disk at DTM_WRITE.

    Applicability is (ISG).

SDTA_Sample (UWORD *) -- Set and get the sound data. Starting
    V40 the sample data does not need to be in CHIP memory. For
    V41 the sample should ALWAYS reside in fast memory.
    NOTE: The memory must be allocated with MEMF_PUBLIC flag set.
    It'll be freed at OM_DISPOSE by FreeVec(). In case that NewDTObject
    fails, you must free the memory yourself. If you wish to use
    any other method than AllocVec to allocate the memory, the
    OM_DISPOSE method must be overriden by your class:

      ...

      case OM_DISPOSE:
  {
    struct SignalSemaphore *lock =
      &((struct DTSpecialInfo *)((struct Gadget *)o)->
      SpecialInfo)->si_Lock;

```
      /* Wait till any write in progress has been finished */
      ObtainSemaphore( lock );

      GetDTAttrs( o, SDTA_Sample, (ULONG) &Sample, TAG_DONE );
      SetDTAttrs( o, NULL, NULL, SDTA_Sample, NULL, TAG_DONE );
      /* now free the memory (in this example it was
      ** allocated by AllocMem) */
      FreeMem( Sample, sizeOfSample );

      ReleaseSemaphore( lock );
      /* fall through !!! */
    }
    default:
      DoSuperMethodA( cl, o, msg );

    ...
```

        Applicability is (ISG).

  SDTA_SampleLength (ULONG) -- Length of the sound data in sample frames.

        Applicability is (ISG).

  SDTA_Period (UWORD) -- Set and get the period of the sound.
        This attribute can be used to affect a playing sound.
        Starting v41 this attribute is obsolete - use SDTA_Frequency
        instead.

        Default for this tag is 394. Applicability is (ISG).

  SDTA_Volume (UWORD) -- Set and get the volume of the sound. This
        attribute can be used to affect a playing sound.

        Valid range is from 0 to 64. Default for this tag is 64.
        Applicability is (ISG).

  SDTA_Cycles (UWORD) -- Set and get the number of cycles the
        sound will be played.

        Default for this tag is 1. Applicability is (ISG).

  The following tags are new for V40.

  SDTA_SignalTask (struct Task *) -- Task to signal when the
        is complete, or if SDTA_Continuous is TRUE, when
        the next buffer is needed.

        Default for this tag is NULL. Applicability is (ISG).

  SDTA_SignalBit (BYTE) -- Signal bit to use with SDTA_SignalTask
        or -1 to disable.

        Default for this tag is -1. Applicability is (ISG).

  SDTA_Continuous (BOOL) -- Used to indicate that the sound
        datatype will be fed a continuous stream of data. Defaults

        to FALSE.

        Default for this tag is FALSE. Applicability is (ISG).

    The following tags are new for V41.

    SDTA_SampleType (UBYTE) -- Set and get sampletype. The following
        types are currently supported:

        SDTST_M8S: standard 8bit mono sample.
        SDTST_S8S: signed 8bit stereo sample (samplewise left/right).
        SDTST_M16S: signed 16bit mono sample.
        SDTST_S16S: signed 16bit stereo sample (samplewise left/right).

        Default for this tag is SDTST_M8S. Applicability is (ISG).

    SDTA_Panning (sposition) -- Set and get the stereofield. This
        attribute can be used to affect a playing sound. Examples:
        0x08000: centers the sample in the stereofield.
        0x10000: pans the sample all the way to the right.
        0x00000: pans the sample all the way to the left.

        Default for this tag is 0x8000. Applicability is (ISG).

    SDTA_Frequency (ULONG) -- Set and get sample frequency. This
        attribute can be used to affect a playing sound.

        Default for this tag is EClockFreq*5/394. Applicability
        is (ISG).

      BUGS
    SDTA_Continuous wasn't supported by v41 before 41.4 --
    GetDTAttrs(o, SDTA_Continuous, &val, TAG_DONE) returned 0L.