DTrace for NextSoftware

1. General info

DTrace is a simple tracer that can be controlled at command line of the executable.

There are only two command-line parameters: tracing file (--TraceFile) and tracing rule (--TraceRule).

At initialization, DTrace-module parses the rule and set up the tracing context. The tracing context is stored in a global variable. All trace-outputs should use this global context.

2. Parameters

2.1 Tracing file (--TraceFile)

Just a filename for a text-file.

E.g.: --TraceFile="tracefile_rec.txt"

2.2 Tracing rule (--TraceRule)

Tracing rule describes when during a runtime a particular output should be activated.

Tracing rule consists of tracing channel(s) and tracing condition(s).

The construction of the rule is: "channel_1,channel_2,...:condition_1,condition_2,..."

Example for a tracing rule: --TraceRule="D_CABAC:poc==0"

Here, tracing channel is D_CABAC and condition is poc==0, which means CABAC tracing output is activated at POC 0 only. You can also use poc>=2 or set the range like poc>=0,poc<=3 for example.

2.2.1 Tracing channel

Channels are defined in dtrace_next.h. Users can add their own channels.

Just put the channel definition into enum-list AND finally add it to the next_channels-table in the function tracing_init().

2.2.2 Tracing condition

Currently only "poc" (binds to currently coded picture order count) and “final” (is set to ‘1’ if the encoder is currently processing final encoding information and to ‘0’ in the RD-search stage, always ‘1’ for the decoder) are supported. Feel free to add your own conditions.

NOTE: Conditions are added and updated during runtime through DTRACE_UPDATE(...). It updates the DTrace internal state, so channels can be activated at the right moment. If it's not updated properly (at right place in the code, e.g. too late), the trace output can become wrong.

For example, "poc"-condition should be updated at the start of the picture(AccesUnit).

Please look into source code for how the "poc"-condition is used.

3. Using of DTrace macros

The most used macro is DTRACE. It's like a printf-function with some additional parameters at the beginning.

Format:

DTRACE(tracing context, tracing channel, "...." [,params]);

Example:

DTRACE(g_trace_ctx, D_CABAC, "EP=%d \n", bin);

There are also macros for output of buffers, picture components or conditional-outputs available. Please have a look into dtrace_next.h.

4. DTrace as a replacement of ENC_DEC_TRACE

The old ENC_DEC_TRACE macro has been ported into the new structure. To generate the same information as with the ENC_DEC_TRACE macro, the application has to be compiled with the ENABLE_TRACING macro set to 1 and run with the following command line:

Encoder: --TraceFile=EncTrace.txt --TraceRule=D_SYNTAX,D_HEADER:final==1

Decoder: --TraceFile=DecTrace.txt --TraceRule=D_SYNTAX,D_HEADER:final==1