



MÄK Data Logger

Frequently Asked Questions

What is the MÄK Data Logger?

The MÄK Data Logger is an easy-to-use application for capturing and replaying simulation data. The Data Logger's intuitive user interface enables you to record HLA or DIS messages to a file and replay them to review, critique, analyze, or debug simulation exercises. The Logger Toolkit allows you to make the Logger an important component in a larger system such as an After Action Review (AAR) or Data Analysis system.

Does it work with any FOM?

Yes! The Logger can be configured to work with any FOM. While it is configured to support the RPR FOM by default, many of our customers are using the Logger with FOMs radically different than the RPR FOM.

Do I need to write any special code if I am not using the RPR FOM?

No, you don't have to; but, if you choose not to, some functionality in the Logger will not work well. When playing back a tape in non-real-time playback mode (i.e. at fast or slow speeds > 1 or < 1), the Time Space Positional Information (TSPI) will not be adjusted to adhere to the dead reckoning algorithm (if one is being used as in the RPR FOM). By using a small plug-in called a FOM Mapper, the logger can adjust the TSPI data (e.g., velocity and acceleration) to correctly adhere to the dead reckoning algorithm. These adjustments will prevent jerky visualizations.

What is a FOM Mapper?

A FOM Mapper is a plug-in (think '.dll', or '.so') which you can write to translate non-standard FOM data into something the Logger understands. For example, the Logger works with Euler angles for entity orientation. If your FOM uses Quaternions to represent entity orientation, a FOM Mapper would make it possible for the Logger to manipulate orientation data in non real-time playback. If you wish to build your own FOM Mapper you will need to own a VR-Link Developers license.

Does the Logger record custom PDUs?

Yes! The Logger will record all PDUs that have a standard DIS Header, regardless of the PDU Kind. No extra work or configuration is required.

How many entities can I record at the same time?

Many. The exact number depends entirely on how you define an *entity*. More importantly, it depends on what the entities are doing. In general the Logger is limited by the following factors:

- Updates per second (including object discovery criteria and interactions received) – You can think of this as a function of the total number of objects and how fast they are moving.
- Checkpointing policy (see: 'What is checkpointing?')

Our internal testing indicates that on a modern desktop computer with a modest checkpoint policy, you should be able to record and play back more than 6 thousand entities with 2-3 updates each per second with no problems.

What is checkpointing?

A checkpoint is the name for an event that occurs when the Logger saves the state of all known objects at a specific time to a file. Checkpoints are used to enable fast time jumping when playing back a file. When a user jumps to a given time in the file the Logger reads the last checkpoint, then it reads every packet up to the jump time to assemble an “up-to-date” list of all objects and their attributes before publishing any required updates to the network. If a very long file has no checkpoints, and a jump is requested near the end of the file, then the Logger is required to read all packets from the beginning of the file to the jump point, a process that may take quite a bit of time for large files.

How do I optimize checkpointing?

Most Logger users never need to worry about optimizing checkpointing. However, if you are using the Logger in a large (over 1k entities) exercise you may want to consider examining your policy.

To optimize the Logger’s checkpoint policy users must balance their need for volume of updates recorded (infrequent checkpoints) with their requirements for fast jump times (frequent checkpoints).

Frequent checkpointing allows for very fast time jumps while significantly increasing the total amount of work the Logger does during recording and playback. The more time the Logger spends writing checkpoints, the higher the probability the Logger will miss updates, or get bogged down. High checkpoint rates ultimately limit the total number of objects the Logger can successfully record.

On the other hand, infrequent checkpoints will increase the time the Logger takes to jump to a new time while playing back a file. In the extreme, with no checkpoints the Logger will need to read every packet up to the jump time before a jump completes. For recordings of multi-week exercises, this could be a considerable amount of time.

The Logger provides several checkpoint strategies to help you meet the needs of your exercise. You can decide to checkpoint after some period of time, or after a specific number of packets are received. For recordings with only several hundred entities, either checkpoint policy will work well. However, for recordings with several thousand moving objects, a time based policy may make better sense.

Can I extend it?

Absolutely! All MAK products come with a toolkit. The Logger Toolkit allows you to extend, or even embed the Logger in another application.

Do I need to buy VR-Link?

Only if you want to extend the Logger using the Logger Toolkit. VR-Link is not required to use the Logger as a stand-alone application. VR-Link is also not required if you have a pre-built FOM Mapper such as the MATREX FOM Mapper.

Do I need to buy Qt?

Only if you plan to modify the Logger using the Logger Toolkit. If you are not using the Logger Toolkit Qt is not required. Future versions of the MAK Logger will use Qt 4.5+ which is available under the LGPL License, which is compatible with commercial software and available for free.

Does the MAK Logger support DDM?

The logger does support DDM using the conveyed region information. It records the region information received with interactions and updates. It can then use that region information to send the interactions and updates using DDM. The logger also supports a customized DDM strategy where the packet information can be inspected and used to create regions and modify their extents. The customized approach requires the development of a logger plugin that performs the region manipulations.

Can I export to a SQL or Access database with the Logger?

Yes. Once you have finished recording your exercise you can export your file (or portions of it) to an ODBC source on your computer. Currently supported ODBC sources are Access and MySQL. You can also easily export to Excel using the Access data source.

Does SQL exporting work with any FOM?

Yes. However, it requires that you have a correctly formatted OMT file. The OMT file is a full description of the FOM as specified by the HLA Standard. The Logger uses the OMT file to figure out how to decode individual attributes and build a SQL relational database.

Can I export DIS to an SQL or Access database?

Yes. Most standard DIS PDUs are supported. See the MÄK Data Logger User's Guide for more information.

Does SQL recording work with nonstandard PDUs?

No, custom PDUs cannot be recorded to a database without writing a custom plug-in. This is because the Logger doesn't know how to decode the custom PDU into useful records in SQL tables.

Why doesn't the Logger record directly to SQL?

Performance. The MÄK Data Logger uses a highly optimized file format with an optimized disk access algorithm which yields performance superior to a generic relational database. While a SQL database is useful for data analysis, it is often an inferior record/playback choice.

If you are running a long exercise and wish to periodically dump sections of file to a database, please use the Record to Multiple Files feature. This feature will periodically save file segments which can then be opened in another Logger instance and exported to SQL.

How do I analyze the data after I've recorded it?

The Logger itself is not an analysis tool; rather, it is one component in an analysis system. Use the Logger to record your exercise, then either play it back later while running an analysis tool, or export the file to a SQL Database (see "Can I export to a SQL or Access database with the Logger?") and then use your favorite data analysis tool.

What about TENA?

The Logger does not support TENA at this time. We have no current plans to support it in the future. If this is a requirement for you please contact us and let us know.

Is there a 64 bit version?

The logger is delivered with 64 bit libraries for the VC10 compiler only and only 64 bit for this platform. The other platforms only support 32 bit executables and toolkit. However, the 32 bit Logger will work on both 32 bit and 64 bit operating systems. The only time 64 bits is required is if you are trying to link the logger into a 64 bit application. If you have a requirement for 64 bit on another platform, please let us know.

What about HLA Evolved?

HLA Evolved is the recently approved IEEE 1516-2010 standard. The logger supports the IEEE 1516-2010 standard.

Can I control the logger remotely?

Yes! Using the logger's remote control API, you can send message to the logger to perform most of the logger's functionality. The logger provides an example logger control console application that demonstrates the use of the remote control API.