



**The MSC Editor – Executive Summary**

A de-risk project by Viking Software

Bo Thorsen,   
CEO, Viking Software A/S  
[bo@vikingsoftware.com](mailto:bo@vikingsoftware.com)

# The MSC Editor Software Package

The TASTE software development tools have been developed by ESA over a series of years and provides tools for software developers to create and maintain high quality software source code.

This is the definition of TASTE on the ESA project website: “TASTE is a development environment dedicated to embedded, real-time systems and was created under the initiative of the European Space Agency back in 2008, after the completion of a FP6 project called ASSERT. TASTE can be used to design small to medium-size systems; it relies on formal languages and is based on the idea of building ‘correct by construction’ software. TASTE allows software designers to easily integrate heteregeneous pieces of code produced either manually (in C or Ada) or automatically by external modelling tools such as MATLAB Simulink, SCADE, or Real-Time Developer Studio. By using TASTE, a software system architect can easily produce a complete and homogeneous system.”

Viking Software has been working with the ESA team to create this first part of the new version of this system. In this de-risk project, Viking Software has implemented an MSC editor. This is one of the elements of TASTE, and is the first part in a major rewrite of the previous implementations of the TASTE elements.

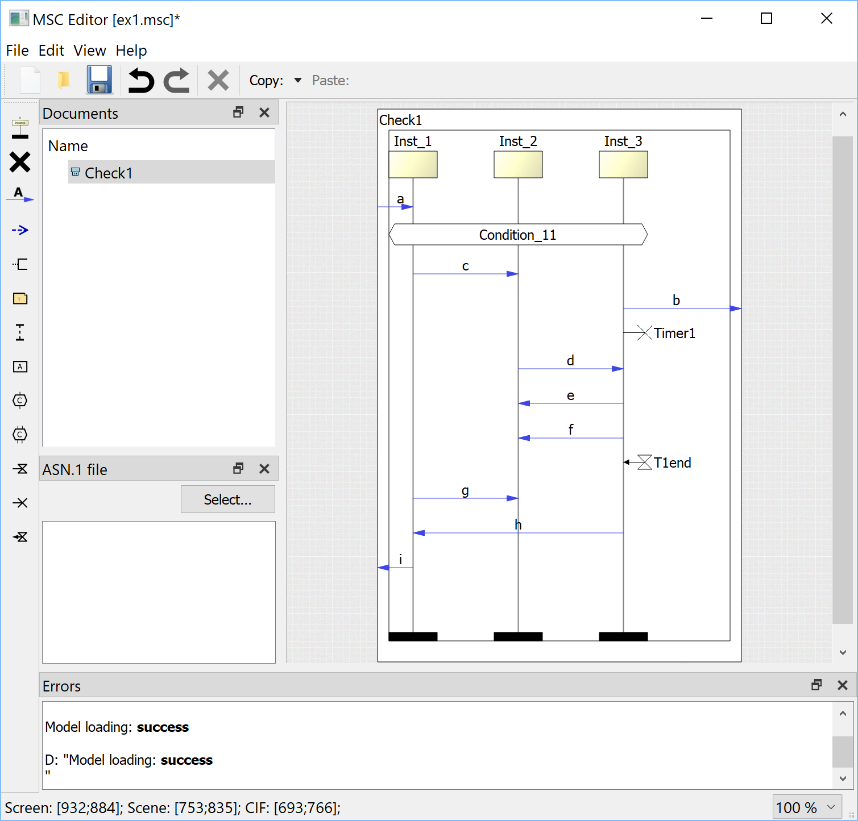
The MSC Editor is a software application for Message Sequence Chart diagrams. An MSC is a diagram that shows communications between system components (also called instances).  Each vertical line in an MSC represents a system components, and horizontal arrows represent messages from one component to another. It is a design-level notation, intended for and best suited for system design involving two or more subsystems; it is not particularly appropriate for system requirements.

During the project, a parser and writer for MSC files has been created. This is able to handle the necessary standard files, and is open for further expansion to other versions of the standard, should this need later arise. The parser is in a library of it’s own, so it is possible to reuse this library in other applications.

A graphical editor for the files has also been created. This uses the parser to read and write MSC files, and offers the graphical editing of MSC files.

MSC files are text files and can be written in a text editor. But a graphical tool provides an overview that is much easier to handle. The tool allows the user to edit the text of the MSC files and later load this in the graphical editor. Or a user can use the graphical editor to create the MSC file and modify it manually. This is not the intended use, but it is possible to do it.

Below is shown a screenshot of the application running on Windows. In the screenshot, some of the capabilities of the editor are shown.



In the application, all aspects of the MSC files can be handled graphically, and the resulting MSC file can be saved or inspected.

The application can also handle ASN.1 files, which are used to describe the contents of messages sent between instances.

These are separate files that can be linked to the MSC charts and modified in the application.

The application can run standalone on Windows, Linux or Mac, or it can run as a plugin to the Qt Creator IDE[[1]](#footnote-1). This allows projects using the Qt Creator IDE as the main development IDE to maintain the MSC files directly in the development environment.

Qt Creator is a general purpose IDE that can be used to create any kind of software. It is often used to work on software development projects with Qt as the basic framework.

Finally, a streaming mode has been implemented. This offers the possibility of sending commands from Python scripts to the application.

This is useful because the other TASTE tools can record the actual messages going back and forth in a running system, and save those as a Python script, and this script can be run in the application.

When doing this, the application shows the contents of the script like a movie being played back. Software developers use this for example for debugging purposes. And it can also be used for visualizing the messages happening in the system.

# Commercial Application

This de-risk project has resulted in an application that any person or company can use to create and maintain MSC files.

The commercial value of this tool comes from three areas:

* The existence of the tool should allow further use of MSC files in software development for future ESA missions as well as other software development projects.
* The use of this tool is free of charge to any interested party, lowering the cost of using MSC files in software development projects.
* With the rewrite of TASTE, there is a possibility to provide commercial training and support for the applications. Viking Software is going to provide this.
* Due to the open source license of the application, there is no vendor lock-in. This means any person or company is free to create a training or support program. Viking Software only has the advantage of experience over other potential vendors.

The full value of the MSC editor will not be present before the rewrite of the TASTE suite has been completed. But even on it’s own, it is already a substantially better tool than what is currently available. The value is already there, but it will increase when the complete new TASTE IDE is available for users.

The purpose of this de-risk project has not been to create a commercial program for Viking Software, ESA or any other entity. It has been to create the tool. That the tool exists, results in potential commercial value.

1. Integrated Development Environment, an application that software developers use to develop software in. [↑](#footnote-ref-1)