Question
When I submit my MADYMO calculation, the job aborts randomly without a clear reason. What could be the cause of this behaviour?

Answer
Despite the stringent QA process that TASS enforces on its products, there can be various reasons why calculations abort. The list below is a compilation of reasons for aborting calculations based on the history of TASS support. Please check if any of the following reasons may apply to your situation:

- A distinction shall be made between an 'abort' caught by the MADYMO Solver and echoed as 'MADYMO terminated abnormally' and cases where the solver aborts without message.
  - When the message 'MADYMO terminated abnormally' is given, the simulation has run into a problem that prohibits continuation of the calculation. The time integration may become unstable because of a too large time step, the velocity of finite element nodes may become too large, the number of separate contact detections becomes too large, etc. The reason is often found in modelling errors. In such cases, contact TASS support for assistance.
  - When the solver aborts without a message, a bug in the code may be the cause and you may want to contact TASS support to report the problem. In this case not all information of the system is echoed in the log file and it is highly appreciated when you share the entire screen output with the TASS support organization.
- Anti-aliasing is a feature, implemented for example in the Hybrid III series that requires more intermediate output for the filtering algorithms. The output is written in temporary files stored in the location set by the environment variable TMPDIR. If the directory set by TMPDIR is limited in its size, the MADYMO Solver cannot write the intermediate output and aborts.
- You may be trying to launch a 64-bit application on a 32-bit Operating System. This generally does not work. The reverse often works, i.e. a 32-bit application can be run on a 64-bit OS. An example of this case is given below in which MADYMO for 64-bit IBM AIX is started on a 32-bit system:

```
Running /appl/madymo6.4.1/madymo_641/ibmrs51/bin/bsaf 'a_frontalfc.xml'
'a_frontalfc.saf'
Could not load program bsaf:
Symbol resolution failed for bsaf because:
Symbol _GetCatName__FiPCc (number 135) is not exported from dependent
module /usr/vacpp/lib/libC.a[ansi_64.o].
Symbol __ct__Q2_3std8_LocinfoFPCci (number 204) is not exported from
dependent
module /usr/vacpp/lib/libC.a[ansi_64.o].
Examine .loader section symbols with the 'dump -Tv' command.

** ABORT ** ID=(INPI41/3D/CMDL3D)
Failed to parse the XML input file. Use the MADYMO Reference Manual to assist
with correction of the errors in the input deck.
MADYMO TERMINATED ABNORMALLY, because of ERRORS.
Error: Job Execution Failed.
```

- You may be trying to launch a Linux 2.6 kernel application on a Linux 2.4 kernel operating system. This generally does not work, i.e. there is no 'forward' compatibility between Linux kernel versions. In general, the Linux community does provide backwards compatibility, i.e. a Linux 2.4 kernel application runs on a Linux 2.6 kernel operating system. Error messages from the OS may be similar to:

```
/lib64/tls/libc.so.6: version `GLIBC_2.3.4' not found
```

Summarizing the last two bullets:

		MADYMO installer		
OS	kernel	Linux2.4-X86	Linux2.4-x86_64	Linux2.6-x86_64
32-bit	2.4	OK	Not OK	Not OK
	2.6	OK	Not OK	Not OK
64-bit	2.4	OK	OK	Not OK
	2.6	OK	OK	OK

- Calculations have been known to crash in MPP mode because of incorrect configuration of the cluster. Possible causes have been identified as:
  - Incorrect or incomplete information in the host files of the nodes of the cluster (often located in /etc/hosts). For example, the nodes may be named incorrectly causing a problem in looking up the IP address of the node.
  - Limits set to stack size or memory usage. Calculations done in MPP coupling may exceed the limits causing MPP processes to die without proper message of the cause, for example:

    ```
    MPI Application rank 0 exited before MPI_Finalize() with status 0
    ```

    Put the command **unlimit** in the .cshrc file when using a csh or tcsh shell to eliminate this problem.
    For bash shells the commands **ulimit -s unlimited** and **ulimit -v unlimited** should be put in the .bashrc file to eliminate limits on stack size and virtual memory size respectively.

Additional Comments

Error messages vary for different operating systems and their versions. When in doubt, contact your local MADYMO Support organization.