Release Guide
Documentation Feedback

At MSC Software, we strive to produce the highest quality documentation and welcome your feedback. If you have comments or suggestions about our documentation, write to us at: documentation-feedback@mscsoftware.com.

Please include the following information with your feedback:

- Document name
- Release/Version number
- Chapter/Section name
- Topic title (for Online Help)
- Brief description of the content (for example, incomplete/incorrect information, grammatical errors, information that requires clarification or more details and so on).
- Your suggestions for correcting/improving documentation

You may also provide your feedback about MSC Software documentation by taking a short 5-minute survey at: http://msc-documentation.questionpro.com.

Note: The above mentioned e-mail address is only for providing documentation specific feedback. If you have any technical problems, issues, or queries, please contact Technical Support.
Running Dytran Shared Memory Parallel (SMP) ................................................................. 32
Running Dytran Distributed Memory Parallel (DMP) .......................................................... 32
Running Dytran with ATB input files .................................................................................. 32
1 Introduction
Introduction

Dytran™ 2019 is the latest and most comprehensive version of Dytran released by MSC Software, providing you with new simulation technology and improved performance.

Dytran 2019 is available on Linux X8664 (Red Hat 7.1, Red Hat 7.3, SUSE ES 11 SP4 and SUSE ES 12 SP1) and Windows 64 bit (Windows 7 and Windows 10). Please see Chapter 3: System Information of the Dytran 2019 Release Guide.

Dytran 2019 includes new capabilities that are focused on two features:

- Dytran now supports User Defined Services (UDS).
- Automated Coupling Surface:
  - Dytran DMP can now be used when ACS is activated in the model.
  - The ACS coupling surface can now fail. Once an element of the structure has failed, it will be removed from the calculation and the gap will be considered a hole in the coupling surface.
  - A number of robustness issues have been addressed.
- Cohesive friction:
  - Cohesive friction can be defined locally on the surface interface.
- Dytran Explorer:
  - Now queuing and scheduling has been added.
  - Processing of UDS services has been added.

In addition, several critical software defects have been corrected in this release. The Dytran 2019 online documentation is available in PDF format on all platforms. The online documentation includes the Dytran Reference Manual, Dytran Theory Manual, Dytran User’s Guide, Dytran Example Problem Manual, Dytran Release Guide, and the Dytran Installation Instructions.

Dytran uses the Macrovision FLEXlm™ licensing system. If you already have an MSC.Dytran 2018 license, you will not need to obtain a new authorization code to activate Dytran 2019 on your computer. However, you will need to install the latest FLEXlm 11.13 license server. MSC Licensing software is now provided on its own Product Page on the Solution Download Center (SDC).

If you need assistance while installing Dytran 2019, please call the MSC Technical Support Hotline at 1-800-732-7284, or E-mail your support questions to mscdytran.support@mscsoftware.com.

For the latest information on supported platforms for upcoming releases of MSC products, please visit the following web site: http://www.mscsoftware.com/Support/Platform-Support/Default.aspx.
2 Enhancements

- SCA User Defined Services (UDS)  4
- Localized Cohesive Friction   8
- DMP for the Auto Coupling Approach  10
- Simulating Flow Through Failed Shells with the Auto Coupling Approach  11
- UDS Build Tool   15
- List of Software Corrections in Dytran 2019  21
- List of Known Software Defects in Dytran 2019  23
SCA User Defined Services (UDS)

In this version of Dytran, a number of UDS services have been reintroduced and a couple of new ones have been created. Below follows a short description of each one. Detailed information is available in the User Defined Services User’s Guide.

<table>
<thead>
<tr>
<th>Subroutine name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EOSUDS</td>
<td>Calculates the equation of state for Lagrangian and Eulerian solid elements</td>
</tr>
<tr>
<td>FAILUDS</td>
<td>User defined failure of Lagrangian and Eulerian materials.</td>
</tr>
<tr>
<td>FLOWUDS</td>
<td>Returns the velocity, pressure, density, and specific internal energy at a single material Eulerian user defined flow boundary.</td>
</tr>
<tr>
<td>TABLUDS</td>
<td>User defined table</td>
</tr>
<tr>
<td>TICEUDS</td>
<td>Defines an initial condition for Eulerian elements and/or grid points at the beginning of the analysis</td>
</tr>
<tr>
<td>PORUDS</td>
<td>The user subroutine defines a general porosity model for coupling surfaces</td>
</tr>
<tr>
<td>SHRUDS</td>
<td>The user subroutine defines the shear modulus, for Lagrangian and Eulerian elements with shear strength</td>
</tr>
<tr>
<td>YLDUDS</td>
<td>The user subroutine defines the yield stress for Lagrangian and Eulerian elements with shear strength</td>
</tr>
<tr>
<td>FORCUDS</td>
<td>The user subroutine defines the nodal velocities for Lagrangian elements</td>
</tr>
<tr>
<td>COMPUDS</td>
<td>The user subroutine defines the orthotropic failure criteria including degradation for Lagrangian shell elements</td>
</tr>
</tbody>
</table>

Example for a variety of UDS services can be found in the Dytran installation in the UDS-Examples directory.

**EOSEX**
- trivial/t11/n11020-c
- trivial/t11/n11020
- verif/smp-fe/rest_solids-c
- verif/smp-fe/rest_solids
- verif/user_rtns/n001-c
- verif/user_rtns/n001
- verif/user_rtns/n004-c
- verif/user_rtns/n004

**FAILEX**
- trivial/t11/n11024-c
- trivial/t11/n11024
Chapter 2: Enhancements

SCA User Defined Services (UDS)

- verif/roe/bnds
- verif/roe/bnds-c
- verif/roe/bnds_exflow
- verif/roe/bnds_exflow-c
- verif/roe/bnds_exflow
- verif/roe/bnds
- verif/roe/wig-c
- verif/roe/wig
- verif/tet4/n003
- verif/tet4/n003-c
- verif/user_rtns/exfail001-c
- verif/user_rtns/exfail001

**FAILEX1**

- trivial/materials/faildt_sol01-c
- trivial/materials/faildt_sol01-f
- verif/smp-fe/rest_solids-c
- verif/smp-fe/rest_solids

**FAILEX2**

- trivial/t11/n11018-c
- trivial/t11/n11018

**FLOWEX**

- trivial/t10/n10005-c
- trivial/t10/n10005
- trivial/t9/n9005-c
- trivial/t9/n9005
- verif/bnds/exflow3-tank-c
- verif/bnds/exflow3-tank
- verif/roe/bnds_exflow
- verif/roe/bnds_exflow-c
- verif/roe/bnds_exflow
- verif/roe/machstem-exflow-c
- verif/roe/machstem-exflow
- verif/viscosity/posuin-c
- verif/viscosity/posuin
- verif/viscosity/posuin-geo-c
- verif/viscosity/posuin-geo
- verif/viscosity/posuin-T11-c
- verif/viscosity/posuin-T11
- verif/viscosity/posuin-T9-c
- verif/viscosity/posuin-T9
- verif/viscosity/posuin-T9-geo-c
- verif/viscosity/posuin-T9-geo

**FLOWXDR**
- trivial/t11/n11015-c
- trivial/t11/n11015
- trivial/t11/n11015md-c
- trivial/t11/n11015md
- trivial/t9/n9005-c
- trivial/t9/n9005
- verif/bnds/exflow3-wave-c
- verif/bnds/exflow3-wave
- verif/bnds/exflow-dir
- verif/viscosity/posuin-md-c
- verif/viscosity/posuin-md
- verif/viscosity/posuin-T11-md-c
- verif/viscosity/posuin-T11-md
- verif/viscosity/posuin-T9-md-c
- verif/viscosity/posuin-T9-md

**FLOWXSQ**
- verif/bnds/exflow3-tank-c
- verif/bnds/exflow3-tank
- verif/bnds/exflow3-wave-c
- verif/bnds/exflow3-wave
- verif/bnds/exflow-sq
- verif/bnds/tankflow-c
- verif/bnds/tankflow

**TABLEEX**
- trivial/t4/n4003-c
Chapter 2: Enhancements
SCA User Defined Services (UDS)

- trivial/t4/n4003
- verif/corotspring/lelas04a

**TICEEX**
- trivial/t11/n11008-c
- trivial/t11/n11008
- verif/roe/bnds-c
- verif/roe/bnds
- verif/roe/bnds_exflow-c
- verif/roe/bnds_exflow
- verif/sloshing/exinit-c
- verif/sloshing/exinit

**TICGEX**
- verif/extra/n010
- verif/extra/n010-c

**POREX**
- trivial/t11/n11016-c
- trivial/t11/n11016
- trivial/t11/n11016md-c
- trivial/t11/n11016md
- verif/porosity/n011-c
- verif/porosity/n011md
- verif/porosity/n011md-c
- verif/porosity/n011
- verif/porosity/n012-c
- verif/porosity/n012
- verif/roe/porex-c
- verif/roe/porex

**SHREX**
- trivial/t11/n11010-c
- trivial/t11/n11010
- verif/user_rtns/n001-c
- verif/user_rtns/n001
Localized Cohesive Friction

In hydroplaning soil can stick to the wheel. This sticking can already be simulating by using PARAM, COHESION, but this PARAM applies to the whole wheel. In practice it can happen that mud sticks to only part of the wheel, while it slides off other parts. To simulate this, local cohesive friction has been added. It can be activated by using the new entries COUCOHF and COHFRIC.

The two figures below show an example model. Two wedges (black) are being pulled out of soil (red). The blue represent air. Here air and soil are modeled by a Euler mesh of 320x1x80 elements. For this model, getting a smooth interface between air and soil requires a fine Euler mesh.
DMP for the Auto Coupling Approach

For simulating fluid-structure interaction Sol 700 provides the fast coupling approach. It is activated by PARAM, FASTCOUP. It requires that the surface defining the interaction is a closed surface. The user has to close the surface by creating dummy surfaces and this can make the method less user friendly. To provide a user friendly approach, the auto coupling approach had been introduced in Nastran 2018. It supports interaction surfaces with holes or T-joints and is activated by PARAM, AUTOCOUP. In this release DMP has been enabled for this approach.

Cpu timings with dmp are shown for a simple test model. The model consists of a plate with hole. On the left side of the plate there is high pressure. On the other side there is low pressure. This gives the velocity field shown in the fringe plot below.

Running the model on 1, 2 and 4 CPUs gives:

<table>
<thead>
<tr>
<th>Number of CPUs</th>
<th>Time</th>
<th>Speed-up</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>273</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>176</td>
<td>1.56</td>
</tr>
<tr>
<td>4</td>
<td>135</td>
<td>2.0</td>
</tr>
</tbody>
</table>

The model uses 136161 Euler element.
Simulating Flow Through Failed Shells with the Auto Coupling Approach

Support of auto coupling to enable flow through failed segments has been added. It is activated by setting the option FAIL on the PARAM, AUTOCOUP entry. An example of a bird strike is shown below.

In the picture below a cylinder models the bird and impacts a shell shown in black. This causes failure of the shell and the bird flows through the failed segments. Only one Euler domain is needed to model the bird. Without the auto coupling approach two Euler domains would be needed. Therefore, auto coupling offers a more user friendly simulation method.
New Features of DytranExplorer

The following new features have been added to DytranExplorer:

1. Menu button to open Queue/Scheduler window.
2. Menu button to open UDS build tool.
3. Separate buttons for filtering Paraview PVD and VTU files in the output files panel.
4. Selector to activate UDS and browser to select the UDS libraries.

Setting Queue and Scheduler Parameters

“Scheduler” or “Queue” can activated in the “tools/options/Queue” menu. New jobs will not be submitted, but scheduled or added to the run queue instead.

Users can select between “queue” and “schedule” functionality and when “schedule” is active they can set the time and date on which the next job that is created should be started.

If “queue” is active, the job will be added to the end of the queue and if there are no other jobs waiting or running, it will be started immediately.
Manipulating Scheduled or Queued Jobs

Jobs added to the scheduler or queue can be listed and manipulated in the Scheduler window, activated by the menu button with the clock icon.

With the up and down arrows the priority of queued jobs can be changed, with the waste basket button jobs can be removed from the queue/scheduler, with the stop button running jobs can be killed and with the “Reschedule/queue” finished jobs can be entered to the queue/scheduler again.
UDS Build Tool

In order to do UDS builds, first the location of the SDK installation needs to be set in tools/options/folders:

Next the build tool can be opened
There are directory browsers to select the source and object trees, two list windows to set the services to be included in the build, a start button and a button to open the log file in a file browser when it's available.
List of Small Updates in Dytran 2019

Enhancing COSUBMAXT

During every cycle in a coupled Euler/Lagrange simulation the geometrical coupling surface computation is performed which can be very expensive. However, when the velocity of the structure is moderate it is not efficient to do these computations each cycle. For this reason, PARAM, COSUBMAX is already available to allow the solver to skip this geometric surface calculation a certain number of cycles, which is called subcycling. PARAM, COSUBMAXT is available to vary the magnitude of the subcycling during the simulation. However, this option did not allow to temporarily switch subcycling off. This switching off has now been added by allowing a value of zero in the table that is referenced by PARAM, COSUBMAXT.

For example:

```
PARAM,COSUBMAXT,501
TABLED1,501,,,,,,,,,+,
+,0.0,0.0,0.0025,0.0, 0.0025001, 2.0, 0.005,5.0,+ 
+,0.01,10
```

This model will show in the OUT-file:

```
NCYCLE NSUB TIME TOTAL-E NZ DLTH SSPD VELOCITY LMIN
831 0 2.496E-03 1.337E+04 C00032 3.000E-06 1.031E+03 5.777E+02 4.000E-02
832 0 2.499E-03 1.340E+04 C00032 3.000E-06 1.030E+03 5.776E+02 4.000E-02
833 0 2.502E-03 1.342E+04 C00032 3.000E-06 1.030E+03 5.773E+02 4.000E-02
834 1 2.505E-03 1.345E+04 C00032 3.000E-06 1.030E+03 5.767E+02 4.000E-02
835 0 2.508E-03 1.347E+04 C00032 3.000E-06 1.029E+03 5.759E+02 4.000E-02
836 0 2.511E-03 1.350E+04 C00032 3.000E-06 1.029E+03 5.751E+02 4.000E-02
837 2 2.514E-03 1.353E+04 C00032 3.000E-06 1.029E+03 5.741E+02 4.000E-02
```

Up to cycle 833 NSUB =0 and there is no subcycling. Up to cycle 833 the coupling surface computation is done each cycle. At cycle 834 subcycling is started with NSUB =1.

Element Summary for Failed Elements

Dytran allows for the simulation of failure of elements. However, after failure, the grid points of the failed elements can move unrestricted and this could sometimes lead to unrealistic values in the element summaries. A new parameter, PARAM,FALENEG is introduced that turns on a new algorithm to store the eroding kinetic, the eroding internal and the eroding hourglass energy separately from the energy summaries of non-failed elements. Then it can prevent from severe energy increase after element failure. By default this parameter is turned off.

Although PARAM,NZEROVEL which removes the nodes after failure also prevents from this behavior, it also removes the nodes and masses after failure. So in case a user wants to keep masses and nodes for other purpose and does not want the severe energy increase, the new parameter will help getting improved results.

For example a model has the following results for PARAM, FALENEG, OFF on the left and PARAM, FALENEG, ON on the right.
The total energy will not change whether the parameter is turned on or off:

**Figure 2-1**  1a: Total Energy (OFF)  
1b: Total Energy (ON)

Only the kinetic and internal energies of the non-failed elements are caught when the PARAM is turned on.

**Figure 2-2**  2a: Kinetic Energy (OFF)  
2b: Kinetic Energy (ON)

**Figure 2-3**  3a: Internal Energy (OFF)  
3b: Internal Energy (ON)
When PARAM, FALENEG, ON the following element histories are available.
The kinetic and internal energies of the failed elements:

![Figure 2-4 Kinetic Energy (ON)](image)

![Figure 2-5 Internal Energy (ON)](image)
When the PARAM is turned ON the OUT-file will show two additional values EROENGY-KN and EROENGY-IN in the material summary (MATSUM):

```
*****************************************************************************
* *
* MATERIAL SUMMARY OF CYCLE 1024 TIME= 5.5632E-04 *
* *
* NAME  MASS  VOLUME  X-MOMENTUM  Y-MOMENTUM  Z-MOMENTUM *
* 2  7.0650E-01  1.4927E-04  -3.7443E-02  -4.4439E+00  3.3558E-03 *
* TOTALS  7.0650E-01  1.4927E-04  -3.7443E-02  -4.4439E+00  3.3558E-03 *
* *
* NAME  ENERGY-TOT  ENERGY-KIN  ENERGY-INT  ENERGY-DIS  ENERGY-HRG *
* 2  1.5985E+03  1.4962E+02  3.9685E+02  3.9685E+02  4.0149E+01 *
* TOTALS  1.5985E+03  1.4962E+02  3.9685E+02  3.9685E+02  4.0149E+01 *
* *
* NAME  EROENGY-KN  EROENGY-IN *
* 2  5.3723E+02  4.7470E+02 *
* TOTALS  5.3723E+02  4.7470E+02 *
* *
*****************************************************************************
```
List of Software Corrections in Dytran 2019

The following software defects are fixed in this release:

<table>
<thead>
<tr>
<th>Defect</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYT-1815</td>
<td>A description of the Mohr Coulomb Yield model was missing from the Dytran Theory Manual. This was added.</td>
</tr>
<tr>
<td>DYT-1748</td>
<td>The combination of adaptive meshing and dmp did not support the use of the COUOPT entries. The COUOPT entries are used to define ambient pressure. Support for this combination has been added to Dytran.</td>
</tr>
<tr>
<td>DYT-1744</td>
<td>Ths2Txt could only process THS files with element and grid ids up to 7 digits (1 - 9,999,999), now also the 8th digit can be used (1 - 99,999,999).</td>
</tr>
<tr>
<td>DYT-1743</td>
<td>Activated &quot;delete&quot; option in the menu that pops up when RST files are right clicked.</td>
</tr>
<tr>
<td>DYT-1740</td>
<td>There is memory leak when MAT2 and other materials are used together. It was fixed.</td>
</tr>
<tr>
<td>DYT-1733</td>
<td>A new output file named &lt;jid&gt;_JOBID.INFO contains the process id of the job submitted by Dytran Explorer. This process ID can be used to kill the process with the Task Manager in situations where Dytran Explorer is &quot;not responding&quot; due to heavy workloads on the system.</td>
</tr>
<tr>
<td>DYT-1730</td>
<td>Incorrect conversion to lower case of generic name. Because the Windows OS is case insensitive, file names are converted to lower case before they are compared to the generic name derived from the name of the input file. This generic name needs to be converted to lower case as well before being used in the comparison.</td>
</tr>
<tr>
<td>DYT-1728</td>
<td>WALLS and WALLOUT has been added to contents table of Chapter 4.</td>
</tr>
<tr>
<td>DYT-1726</td>
<td>Archive files resulting from a run with adaptive meshing (MESH with OPTION=ADAPT) need to defined with SAVE(xxx)=1, otherwise the changed connectivity will not be included and post-processing will not be possible. A remark has been added to MESH entry in the Dytran Reference Guide.</td>
</tr>
<tr>
<td>DYT-1723</td>
<td>When selecting series of archive files that belong to the same set with only different cycle numbers in their names, Arc2Vtk included all files which names started with the base name of the files to be selected. For example when searching for BASE_NAME_nnn.ARC also files matching BASE_NAME_EXTRA_nnn.ARC were included. This has been solved.</td>
</tr>
<tr>
<td>DYT-1722</td>
<td>By default Dytran Explorer started Arc2Vtk with the option to use a TAB character as field separator when converting THS files to “CSV”, however ParaView needs a comma as field separator. The field separator has been changed from TAB to comma.</td>
</tr>
<tr>
<td>DYT-1718</td>
<td>All values were zero when EFFSTS was requested in an output request for Euler elements when the viscosity was used. This has been resolved. The Euler element variable EFFSTS is properly computed in case of viscosity.</td>
</tr>
<tr>
<td>Defect</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DYT-1704</td>
<td>An output request can be made on a selection of Euler elements. When DMP is used, and the model does not contain any structure, the output files that were generated contained all Euler elements in the model. This problem has been fixed.</td>
</tr>
<tr>
<td>DYT-1676</td>
<td>The description of VERSION in the CONTACT entry has been properly realigned.</td>
</tr>
<tr>
<td>DYT-1675</td>
<td>The description of PARAM,EULERPR has been updated to show that &quot;partitions&quot; are used instead of &quot;Euler cubes&quot; by this parameter.</td>
</tr>
<tr>
<td>DYT-1671</td>
<td>Dytran 2019 now supports User Defined Services by using the Software Development Kit. The user subroutine that used to cause failures in older versions of Dytran has been successfully tested with the new UDS functionality.</td>
</tr>
<tr>
<td>DYT-1540</td>
<td>BODYFR1 is not working with HYDRO solver. This is described in the Reference Manual, however, the Dytran solver did not properly trap the mistake in the input file when the user tries. A new error message has been added to trap this.</td>
</tr>
<tr>
<td>DYT-1148</td>
<td>After elements fail, the nodes will have large displacements that can grow out of control. Then all energy values increase dramatically. A change has been made to the code where eroding energies can be stored and other types of energies for failed elements are set to zero. This is controlled by a new parameter: PARAM, FALENEG. The default is OFF. When set to ON this enhancement is activated.</td>
</tr>
<tr>
<td>DYT-1114</td>
<td>Dytran Explorer used to show a command window. An update has been made to minimize this windows at start up.</td>
</tr>
<tr>
<td>DYT-704</td>
<td>The order of the directories on the PATH environment variable was incorrect, causing Dytran to select an incompatible DLL. Changing this order fixed the problem.</td>
</tr>
<tr>
<td>DYT-481</td>
<td>Dytran Explorer was enhanced by adding the options for DMP and SMP to queuing and scheduling of jobs.</td>
</tr>
<tr>
<td>DYT-328</td>
<td>KEYHOFF shell formulation is used in Dytran as the default formulation. Since it controls the warping behavior in shells, it is more useful than BELY shell formulation. However the transverse shear strain calculation gave stiffer results. So when the transverse shear strain was dominant case such as the large rotation, the results were not correct. The bug is fixed.</td>
</tr>
</tbody>
</table>
# List of Known Software Defects in Dytran 2019

The following are known software defects in this release:

<table>
<thead>
<tr>
<th>Defect</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYT-2078</td>
<td>Internally PWELD/CROD combination is changed into BJOIN definition. It is not allowed that the joint connection is used for connecting nodes to rigid bodies. So please do not use the combination to connect between rigid bodies and nodes. Instead of using PWELD/CROD combination, please use the tied contact between nodes and rigid bodies, RCONREL or please use the real beam elements. An error message will be added to the solver to properly catch this when it occurs in an input file.</td>
</tr>
<tr>
<td>DYT-2077</td>
<td>When running DMP, a CONM2 attached to a rigid body will be dropped from the simulation. The following message will appear: &quot;The CONM2 (id 9999999) addresses a non-existing grid point. Grid point number 99999 not found on partition 1. The entry will be ignored.&quot;</td>
</tr>
<tr>
<td>DYT-2026</td>
<td>The automatic coupling approach can sometimes give errors and core dumps for sloshing tanks. This may happen for very complicated shapes of the coupling surface. Solutions to these problems is actively being worked on.</td>
</tr>
<tr>
<td>DYT-1881</td>
<td>The materials RUBBER1/RUBBER2 are not implemented yet in the Dytran SMP solver. It is the limitation in the current Dytran version and an error message will be added to the solver to properly catch this when it occurs in an input file. Also it will be considered for as an enhancement request for a future version of Dytran.</td>
</tr>
<tr>
<td>DYT-1738</td>
<td>When installation Dytran on C:\Program Files it will not be able to execute. Work-around is to install Dytran on C:\MSC.Software\Dytran or another location that does not have spaces in the installation path.</td>
</tr>
<tr>
<td>DYT-1731</td>
<td>The automatic coupling approach can sometimes give errors and core dumps when Euler elements have no longer fluid inside (when they become empty).</td>
</tr>
<tr>
<td>DYT-1717</td>
<td>For models with adaptive meshing and with multiple Euler domains, the Euler archives will use the same element numbers for different Euler meshes. As a result these archives cannot be read in Patran simultaneously. The workaround is to read in each Euler archive separately into Patran.</td>
</tr>
<tr>
<td>DYT-1716</td>
<td>Although the ranges of the X- and Y-axis can be set manually when plotting time-history file data, the complete set of selected curves has to fit inside the range, otherwise the plot library used for drawing the curves will cause a crash of DytranExplorer. To prevent this, the ranges of the X- and Y-axis are checked against the ranges of the data and corrected automatically if necessary.</td>
</tr>
<tr>
<td>DYT-1504</td>
<td>Using a MESH entry with adaptive and a MESH entry with local coordinates together is not supported.</td>
</tr>
<tr>
<td>DYT-1142</td>
<td>DMP error occurred, that can run on single CPU mode. Model contains contact definitions on marker grid points.</td>
</tr>
<tr>
<td>DYT-1140</td>
<td>DMP error occurs when CONTACT and RCONN are used in the same model.</td>
</tr>
<tr>
<td>Defect</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>DYT-1130</td>
<td>MASS is getting increased from 2 FLOWDIR Boundary crossing.</td>
</tr>
<tr>
<td>DYT-607</td>
<td>Limitation of number of Euler material. When number is too high, the DMP simulation does not terminate with a proper error message.</td>
</tr>
<tr>
<td>DYT-606</td>
<td>When SET card is used incorrectly, the DMP simulation does not terminate with a proper error message.</td>
</tr>
</tbody>
</table>
3 System Information

- Software Installation  26
- Licensing  26
- Release Platforms  27
- Memory Requirements  28
Software Installation

Dytran 2019 on Windows and Linux platforms can be downloaded from MSC.Software's Solutions Download Center:

https://mscsoftware.subscribenet.com/

On the Windows platforms, Dytran 2019 can easily be installed as it uses the standard Windows Installation Wizard. On Linux platforms, the MSC Software standard installation script can be used to install the software on your system. Dytran 2019 is the successor of Dytran 2018.

Licensing

Dytran uses the FLEXlm license manager as the licensing system for nodelock and network licensing.

To run Dytran, you need an authorization code from MSC.Software Corporation. If you already have a license for MSC Dytran 2018, you will not need to obtain a new license for Dytran 2019. DMP capability is part of Dytran Standard and no additional licenses are needed to run DMP capability in Dytran 2019. However, in all cases, you do need a new installation of the license server software. Specifically, the FlexLM license server needs to be at level 10 or higher. For this purpose, an installation of FlexLM v11.9 is part of this release on all supported platforms. It is noted that Dytran 2019 is not able to check out licenses when the FlexLM server is lower than version 10.

On Windows and Linux computers, Dytran requires an Ethernet card on your computer even if your computer is not connected to a network. The FLEXlm licensing mechanism uses the Ethernet card to create the unique system identification encrypted in the license information file.
Release Platforms

Dytran 2019 was built and tested on the following hardware with the listed software installed as given in Table 3-1.

Table 3-1  Release Platforms

<table>
<thead>
<tr>
<th>Vendor</th>
<th>OS</th>
<th>FORTRAN Version</th>
<th>C Version</th>
<th>MPI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux X8664</td>
<td>RHEL 7.1</td>
<td>Intel 17.0.5.239</td>
<td>Intel 17.0.5.239</td>
<td>Intel MPI 2017.0.5.249 (Default)</td>
</tr>
<tr>
<td></td>
<td>RHEL 7.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SuSE 11 SP4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SuSE 12 SP1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Windows X8664</td>
<td>Windows 7</td>
<td>Intel 17.0.5.239</td>
<td>Visual Studio Professional 2017</td>
<td>Microsoft v8.1 (Default)</td>
</tr>
<tr>
<td></td>
<td>Windows 10</td>
<td></td>
<td></td>
<td>Intel MPI 2017 Update 2</td>
</tr>
</tbody>
</table>
Memory Requirements

In general, the size of the memory required by Dytran depends on the size of the engineering problem you wish to solve. The default memory size is calculated by the Dytran Solver. It makes an estimate based on the number of elements, number of grid points, boundary conditions, output requests, and others. Typically, the memory does not have to be adjusted anymore since the Dytran 2013 release, but it may still occasionally need adjustment.

On Linux platforms, the user has the command-line option (size=small/medium/large), or the user can add the MEMORY-SIZE definition in the input file. Dytran traces the usage of memory and prints a summary at the end of the output file of each analysis. The memory size listed in the summary is exact. It reflects the memory required for storing the model in core memory after one integration step. Additional memory required during the analysis is automatically allocated and de-allocated.

Under certain conditions, Dytran may stop and issue a message that it cannot allocate the required memory. Since the memory allocation in Dytran is dynamic, the system may require additional memory during an analysis. If the memory is available, it will be allocated and de-allocated when it is no longer needed. When the computer runs out of memory, the Dytran analysis may stop when it needs more memory to continue. The user can solve this problem by closing applications on the computer that are no longer needed, or the user can decrease the size of the core memory that Dytran allocates for the analysis if Dytran decided to use substantially more than the analysis requires. The information on the memory size requirements of the analysis can be found in the memory summary at the end of the analysis. MSC Software Corporation recommends using Dytran on a computer that has at least 4 GB of RAM.
4 Using Dytran

- Running Dytran on Linux  30
- Running Dytran on Windows  31
- User Defined Services  31
- Running Dytran Shared Memory Parallel (SMP)  32
- Running Dytran Distributed Memory Parallel (DMP)  32
- Running Dytran with ATB input files  32
Running Dytran on Linux

On Linux platform, you would use the command line interface like:

- `{Full path to Dytran installation}/2019/bin/dytran jid={jobid}

Alternatively, the user may choose to create the following alias:

- `% alias dytran {Full path to Dytran installation}/2019/bin/dytran

With this alias dytran jobs can be executed. For example:

- `dytran jid=xxx` (to submit a regular Dytran job)
- `dytran jid=xxx bat=no` (to submit Dytran in interactive mode)
- `dytran jid=xxx ncpus=2` (to submit Dytran using two cores for Shared Memory Parallel)
- `dytran jid=xxx udsdir=myuds scaobj=myobjs` (to submit a Dytran job with User Defined Services source files)
- `dytran jid=xxx scaobj=myobjs` (to submit a Dytran job with pre-built User Defined Services libraries)

To submit DMP jobs, you must specify the number of processors as well:

- `dytran jid=xxx dmp=yes ncpus=2` (to submit Dytran using two cores Distributed Memory Parallel)
- `dytran jid=xxx dmp=yes hlist=hostlist.txt bat=no` (to submit Dytran in a cluster using a host list)

**Note:** `xxx` should be replaced by the name of your input deck without the `.dat` extension.
Running Dytran on Windows

On Windows, analysis can be submitted with Dytran.Explorer, a graphical interface to control Dytran jobs and post process the results files. To start with Dytran.Explorer double-click the Dytran icon on your desktop after installing the Dytran 2019. Alternatively, you can use the start Menu to locate Dytran.Explorer under the Programs Folder. The Dytran.Explorer provides an on-line help system which includes online documentation. Basic post processing and animation tools are available by right-clicking on the results files displayed in Dytran Job window.

To submit Dytran jobs from the command line, open a DOS command window and type:

```
"{Full path to Dytran installation}\Dytran\2019\bin\dytran.exe" jid={job-name} dmp=yes ncpus={number of processors}
```

For instance:

1. Name input model is `bunker.dat`.
2. Dytran is installed at `C:\MSC.Software\Dytran`.
3. Model needs to run on four CPUs in DMP mode.

   The correct command would be:

   ```
   C:\MSC.Software\Dytran\2019\bin\dytran.exe jid=bunker dmp=yes ncpus=4
   ```

User Defined Services

Before Version 2018, the user had to compile individual user subroutine and create a new Dytran Executable to extend the functionality of Dytran. With the 2019 release the user had the possibility to create a dynamic library that was accessed and executed by the installed Dytran executable. In the 2019 release, this has been extended to the SCA based “User Defined Services” (UDS), also used by MSC Nastran. Just like with the 2019 release, both FORTRAN and C can be used now.

For User Defined Services, the correct version of the Intel Fortran compiler and Microsoft Visual Studio must be installed (See Chapter 3: System Information for detailed version information). Also a matching version of MSC Software’s “Software Development Kit” (SDK) needs to be installed on the system.

For UDS the source files need to be stored in directory trees with a strict layout. The easiest way to accomplish this is probably copying a complete tree from an example and make modifications to the appropriate routine(s). Building can be done by specifying the “udsdir” and “scaobj” parameters on the Dytran command line. On Windows also DytranExplorer can be used. When using the command line, also the following locations on the SDK installation need to be defined by environment variables:

- The directory holding the SCons executable (variable name SCATOOLS) and the location of the SCA system directory (variable name SCASYSTEM).

An example of using UDS with the Dytran command on Windows could be:

```bash
set SCATOOLS=C:\MSC.Software\SDK\2019\Tools
set SCASYSTEM=C:\MSC.Software\SDK\2019\SCAKernel
dytran.exe jid=xxx udsdir=yyy scaobj=zzz
```

and on Linux (C-shell)
setenv SCATOOLS <sdk-location>/SDK/2019/Tools
setenv SCASYSTEM <sdk-location>/SDK/2019/SCAKernel
dytran jid=xxx udsdir=yyy scaobj=zzz

With <sdk-location> the path to the directory where the SDK has been installed, xxx the name of the input file without “.dat” extension, yyy the path to the root of the directory tree with the modified UDS service(s) and zzz the (empty) directory that will be used as root of the directory tree with the generated UDS libraries. yyy is the directory that contains the four SCons files (SConopts, SConopts.delivery, SConscript and SConstruct) and can be relative to the working directory (the one that contains the input file(s)).

In both the Linux and Windows example, the Dytran command is assumed to be in the search path defined by the PATH environment variable.

Running Dytran Shared Memory Parallel (SMP)

To run Dytran 2019 with SMP enabled:

<DYTRANDIR>/bin/dytran.exe jid=xxxx dmp=no ncpus=<n>

with <DYTRANDIR> the root directory of the Dytran installation, something like C:\MSC.Software\Dytran\2019, xxxx the name of the input file without .dat extension and <n> the number of CPUs to be used. Must be equal to or smaller than the actual number of CPUs on the system.

Running Dytran Distributed Memory Parallel (DMP)

To run Dytran 2019 with DMP enabled on the local system:

<DYTRANDIR>/bin/dytran.exe jid=xxxx dmp=yes <mpi_vers>=yes ncpus=<n>

with <DYTRANDIR> the root directory of the Dytran installation C:\MSC.Software\Dytran\2019, xxxx the name of the input file without .dat extension, <mpi_vers> one of the supported MPI versions (“hpmpi” or “intelmpi”, if not specified the default Microsoft MPI will be used) and <n> the number of CPUs to be used, which must be equal to or smaller than the actual number of CPUs on the system.

When both SMP and DMP are going to be used together, the value of the dmp parameter has to be set to “both” and the number of processors to be used for dmp is defined with the “nproc” parameter:

<DYTRANDIR>/bin/dytran.exe jid=xxxx dmp=both <mpi_vers>=yes ncpus=<n> nproc=<m>

To run Dytran on a cluster, first a host list needs to be prepared to define the nodes and the numbers of cpus to be used on each node. Dytran can be submitted with the command:

<DYTRANDIR>/bin/dytran.exe jid=xxxx dmp=yes hlist=hostlist.txt

Running Dytran with ATB input files

In order to run Dytran 2019 with ATB, make sure the ATB file is in the same directory as the input files and issue the command:

<DYTRANDIR>/bin/dytran.exe jid=xxxx atb=yyyy
with atb the name of the ATB input file without the extension .ain. In case you already have the “.lin” file from a previous run, you can also use:

<DYTRANDIR>/bin/dytran.exe jid=xxxx lin=yyyy

with yyy the name of the “.lin” file without the extension .lin.