

***CHAPTER 6 - MSC.Dytran FILE MANAGEMENT, EXECUTIVE  
AND CASE CONTROL SECTIONS***



***MSC.Dytran***

## СОДЕРЖАНИЕ

- ❑ **Операторы File Management Section (FMS)**
- ❑ **Операторы Executive Control Section**
- ❑ **Операторы Case Control Section**
  - **Инициализация операторов Bulk Data Section**
  - **Управление процессом решения**
- ❑ **Управление выводом результатов**
- ❑ **Управление выводом данных для *рестарта***
- ❑ **Возможные проблемы**

## ОПЕРАТОРЫ FILE MANAGEMENT SECTION (FMS)

### ❑ Управление *рестартом*

- ✓ RESTART – оператор, указывающий, что это – *рестарт* ранее выполнявшегося задания
- ✓ RSTBEGIN – номер шага интегрирования, используемого в качестве исходной точки для *рестарта*
- ✓ RSTFILE – идентификатор файла *рестарта*
- Пример: *рестарт* с использованием файла DEMO\_0.RST, “точка” *рестарта* – 1000-й шаг интегрирования

RESTART

RSTBEGIN = 1000

RSTFILE = DEMO\_0.RST

### ❑ Подключение пользовательской подпрограммы

- ✓ USERCODE – оператор, иницирующий выполнение задания с использованием пользовательской подпрограммы и задающий имя файла с текстом этой подпрограммы

- Пример:

START

USERCODE=user.f

...

## ОПЕРАТОРЫ FILE MANAGEMENT SECTION (FMS)

- ❑ **Управление расчётом предварительно напряжённого состояния**
  - **PRESTRESS** – указание, что до моделирования переходного процесса необходимо провести расчёт предварительно напряжённого состояния
  - **NASTDISP** – задание имени файла, полученного с помощью MSC.Nastran и содержащего информацию о деформации конструкции при  $t=0$
  - **BULKOUT** – задание имени файла, в который будет выведены координаты узлов после расчёта преднапряжённого состояния
  - **SOLUOUT** – задание имени двоичного файла, в который выводятся все результаты расчёта преднапряжённого состояния
  - **NASINIT** – оператор раздела Bulk Data, используемый для управления расчётом преднапряжённого состояния
    - ✓ Пример инициализации расчёта преднапряжённого состояния (файл BLADE.DIS – результат расчёта деформаций в MSC.Nastran, файлы GRID.OUT и SOL.OUT – результаты расчёта преднапряжённого состояния:

**PRESTRESS**

**NASTDISP = BLADE.DIS**

**BULKOUT = GRID.OUT**

**SOLUOUT = SOL.OUT**

## ОПЕРАТОРЫ EXECUTIVE CONTROL SECTION

- ❑ **TIME** – задание максимальной величины затрат процессорного времени для проведения расчёта (в минутах)
  
- ❑ **CEND** – оператор - разделитель (символ конца раздела Executive Section)

## ОПЕРАТОРЫ CASE CONTROL SECTION (CCS)

### □ Три группы операторов:

- Операторы инициализации операторов раздела Bulk Data
  - ✓ Инициализация нагрузок
  - ✓ Инициализация начальных условий
  - ✓ Инициализация закреплений
- Операторы управления вычислениями
  - ✓ Критерий прекращения выполнения задания
- Операторы управления выводом данных
  - ✓ Управление выводом алфавитно-цифровой информации (типа сообщений и т.п.)
  - ✓ Управление выводом результатов
  - ✓ Управление выводом информации для возможного рестарта

## CASE CONTROL: ИНИЦИАЛИЗАЦИЯ ОПЕРАТОРОВ BULK DATA

### □ Назначение

- Инициализация операторов Bulk Data задания закреплений и нагрузок: только инициализированные операторы Bulk Data, принадлежащие к этой группе, будут “задействованы” при расчёте, остальные будут проигнорированы

### □ Инициализация операторов задания нагрузок

- TLOAD – инициализация операторов TLOAD1, описывающих зависимый от времени нагрузки

### □ Инициализация начальных условий

- TIC – инициализация операторов TICn, TICEL, TICGP и TICEUL

### □ Инициализация закреплений

- SPC – инициализация операторов SPCn (закрепление узлов)

## CASE CONTROL: ОПЕРАТОРЫ УПРАВЛЕНИЯ ВЫЧИСЛЕНИЯМИ

### ❑ Назначение

- Задание критерия прекращения выполнения задания

### ❑ Прекращение выполнения задания по достижении процессом заданного значения времени

- Отметим, что задаётся значение времени, при котором будет остановлен данный расчёт, после чего может быть выполнен рестарт (если он был предусмотрен)
  - ✓ Пример: `ENDTIME = 10.0E-3`
  - ✓ По умолчанию `ENDTIME = 0.0`

### ❑ Прекращение выполнения задания после выполнения заданного количества шагов интегрирования

- При выполнении заданного количества шагов выполнение задания будет остановлено
  - ✓ Пример: `ENDSTEP = 10000`
  - ✓ По умолчанию `ENDSTEP = 999999`



## УПРАВЛЕНИЕ ВЫВОДОМ ДАННЫХ

- **Спецификация выводимых данных (FMS + CCS)**
  1. **TYPE** – оператор задания типа файла (FMS)
  2. **SAVE** – оператор задания количества записей, выполняемых в файл перед тем как он будет *закрыт* (FMS)
  3. **STEPS/TIMES** – операторы задания частоты вывода результатов (CCS)
  4. **ELEMENTS/GRIDS/...** – операторы задания вида КЭ примитивов, для которых будут выводиться результаты расчёта (CCS)
  5. **ELOUT/GPOUT/...** – операторы задания типов выводимых результатов (CCS)
  
- **Пример спецификации вывода в “архивный” файл**

**TYPE (ARC) = ARCHIVE**  
**SAVE (ARC) = 10**  
**STEPS (ARC)=100, THRU, END, BY, 100**  
**ELEMENTS (ARC) = 22**  
**SET22 = 12, 100, THRU, 200**  
**ELOUT (ARC) = PRESSUPE, XVEL, YVEL, ZVEL**

## СПЕЦИКАЦИЯ ИНФОРМАЦИИ, ВЫВОДИМОЙ В ФАЙЛ ARC и THS

- ❑ В отдельный файл ARC или THS может выводиться информация только одного типа: для узлов, жёстких тел, материалов, какого-либо одного типа элементов и т.п.
- ❑ Виды примитивов, для которых могут выводиться результаты в файлы ARC или THS:

1D элементы  
 Объёмные *лагранжевы* элементы  
 Четырёхузловые оболочечные элементы  
 Трёхузловые оболочечные элементы  
 Трёхузловые мембранные элементы  
 Оболочечные элементы типа “Dummy”  
 Объёмные гидродинамические *эйлеровы* элементы  
 Объёмные *эйлеровы* элементы с жёсткостью сдвига  
 Объёмные “мульти-материал” *эйлеровы* элементы

Жёсткие тела  
 Жёсткие эллипсоиды  
 Материалы  
 Контактирующие поверхности  
 Поверхности взаимодействия  
 конструкция-жидкость  
 Поверхности надувных подушек безопасности  
 Поверхности  
 “Под”-поверхности  
 Сечения  
 Границы *эйлеровой* сетки  
 Узлов

- Для узлов и каждого вида элемента могут одновременно выводиться данные разных типов
- Список выводимых переменных – см. User Manual п. 3.9

## ЗАДАНИЕ ТИПА ВЫХОДНОГО ФАЙЛА

- Тип выходного файла (задаётся в операторе TYPE)
  1. ARCHIVE – “архивный” файл – двоичный файл для хранения результатов вычислений, анализируемых в графической форме
  2. TIMENIS – файл временных зависимостей – двоичный файл для хранения результатов вычислений, используемых для построения графиков
  3. RESTART – файл *рестарта* – двоичный файл для хранения данных, необходимых для рестарта
  4. STEPSUM – информация о выполненном шаге интегрирования (однострочная информация в OUT-файле)
  5. MATSUM – периодически выдаваемая в OUT-файл информация о параметрах материалов (масса, объём, энергия и т.п.)
  6. EBDSUM – периодически выдаваемая в OUT-файл информация о параметрах на границах *эйлеровой* сетки (интенсивность переноса массы и энергии через границу и т.п.)
  7. MRSUM – периодически выдаваемая в OUT-файл информация о переменных, характеризующих жёсткие (Rigid) материалы (массы, моменты количества движения, энергия и т.п.)

## ЗАДАНИЕ ЧАСТОТЫ ЗАКРЫТИЯ ФАЙЛА

- Оператором **SAVE** задаётся частота закрытия файла с результатами вычисления и создания нового файла, в который продолжится вывод информации
  - Пример: в файл с логическим именем **OUT1** будет выполняться **10** записей результатов вычислений, после чего он будет закрываться, а вывод информации продолжится во вновь созданный файл  
**SAVE (OUT1) = 10**
  - Для файлов типа **RESTART** возможно задание *отрицательного* интервала закрытия файла. В этом случае новый файл *рестарта* не будет создаваться, а вся необходимая для рестарта информация будет записываться в уже существующий файл на место предыдущей информации (делая “старую” информацию недоступной), например:  
**SAVE (RST) = -1**

## ЗАДАНИЕ ЧАСТОТЫ ВЫВОДА РЕЗУЛЬТАТОВ

- ❑ **Вывод в определённые моменты времени**
  - Список моментов времени, в которые будут выведены результаты  
TIMES (OUT1) = 0.2E-3, 0.6E-3, 0.8E-3
  - Вывод результатов в заданном промежутке (THRU) времени и с заданным интервалом (BY) времени  
TIMES (OUT1) = 0.2E-3, THRU, 1.0E-3, BY, 0.1E-3
  - Вывод результатов *до конца* моделируемого процесса  
TIMES (OUT1) = 0, THRU, END, BY, 0.1E-3
  
- ❑ **Вывод на определённых шагах интегрирования**  
STEPS (OUT1) = 1, THRU, END, BY, 100

## ПРИМЕРЫ СПЕЦИФИКАЦИИ ВЫВОДА РЕЗУЛЬТАТОВ

□ Узлы

GRIDS (OUT1) = 10  
SET 10 = 1 THRU 2000  
GROUT (OUT1) = XDIS, YDIS, ZDIS

□ Элементы

ELEMENTS (OUT1) = 20  
SET 20 = ALLELEMENTS  
ELOUT(OUT1) = TXX01, EFFSTS

□ Жёсткие тела

RIGIDS (OUT1) = 30  
SET 30 = 40, MR200, FR300  
RBOUT(OUT1) = XPOS, ZVEL

□ Жёсткие эллипсоиды

RELS(OUT1) = 40  
SETC 40 = HEAD, NECK, UT  
RELOUT(OUT1) = ZPOS

□ Материалы

MATS(OUT1) = 50  
SET 50 = 200, 300  
MATOUT(OUT1) = SIE

□ Контакты

CONTS (OUT1) = 70  
SET 70 = 1  
CONTOUT(OUT1) = XFORCE

## ПРИМЕРЫ СПЕЦИФИКАЦИИ ВЫВОДА РЕЗУЛЬТАТОВ

❑ Поверхности взаимодействия

CPLSURFS (OUT1) = 80

SET 80 = 25

CPLSOUT(OUT1) = PRESSURE

❑ Подушки безопасности

GBAGS (OUT1) = 90

SET 90 = 10

GBAGOUT(OUT1) = PRESSURE

❑ Поверхности

SURFACES (OUT1) = 100

SET 100 = 25

SURFOUT(OUT1) = PRESSURE

❑ “Под”-поверхности

SUBSURF (OUT1) = 110

SET 110 = 787

SURFOUT(OUT1) = MFLR-POR

❑ Сечения

CSECS (OUT1) = 120

SET 120 = 1

CSOUT(OUT1) = XFORCE

❑ Границы эйлеровой сетки

EBDS (OUT1) = 130

SET 130 = ALLEBDS

EBDOUT(OUT1) = PRESSURE

## УПРАВЛЕНИЕ *РЕСТАРТ*ОМ

- **Пример спецификации сохранения данных для *рестарта***
  - **Пример спецификации сохранения данных для *рестарта* в заданные моменты времени**  
TYPE (RST) = RESTART  
TIMES (RST) = 0, THRU, END, BY, 1.0E-3  
SAVE (RST) = 10
  - **Пример спецификации сохранения данных для *рестарта* через заданное количество шагов интегрирования**  
TYPE (RESTART) = RESTART  
STEPS (RESTART) = 100, THRU, END, BY, 100  
SAVE (RESTART) = -1

**Примечание:** при спецификации файла *рестарта* нет необходимости указывать какие данные в нём необходимо сохранять, так как в файле *рестарта* сохраняется вся информация для всех узлов и элементов



## ВОЗМОЖНЫЕ ПРОБЛЕМЫ ПРИ ПОСТРОЕНИИ CCS

- ❑ Указание слишком длинных списков выводимых данных
- ❑ Использование сразу двух опций TIME и STEP для задания частоты вывода одних и тех же данных
- ❑ Неполное описание всех опций, необходимых для организации вывода данных
- ❑ Использование одного и того же номера для операторов SET и SETC
- ❑ “Пустые” наборы данных: запрос вывода данных для примитива (узла, элемента и т.п.), не представленных в модели
- ❑ “Описки” (неправильный синтаксис). Например, при обнаружении программой буквы “O” вместо нуля в числе 10000 будет выдано следующее сообщение:

```
%E-P1008101-NAS_CONVERT_TOK_TO_INT,,,  
INPUT DECK: water_jet.dat  
LINE #16   : SAVE,EULER,10000  
MESSAGE    : Error converting token into integer  
            Token = 10000  
            Istat = -1
```