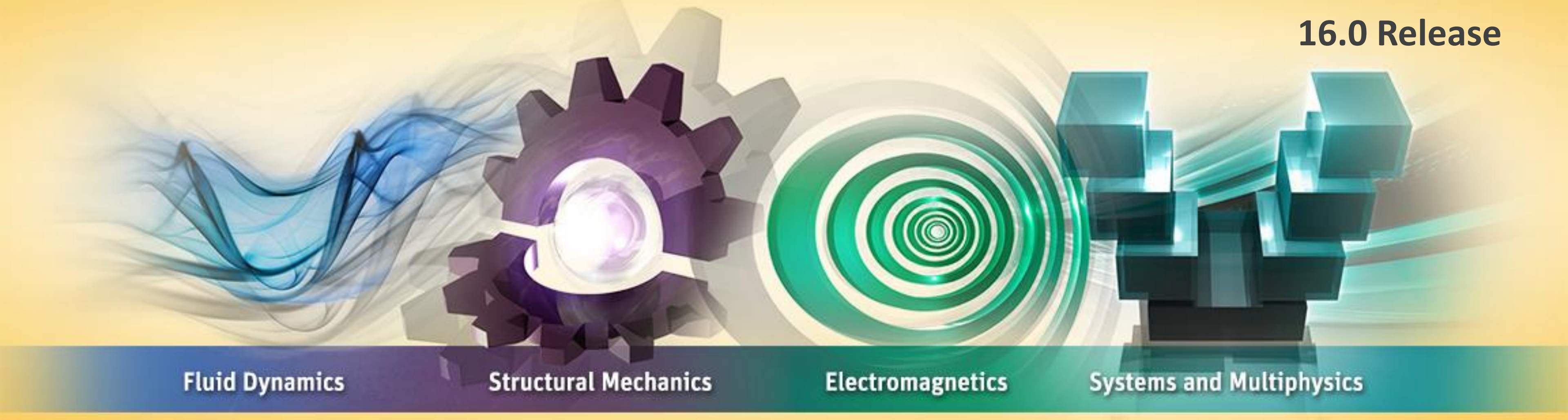
Lecture 13:
Additional Variables (AVs), CFX Expression Language (CEL) & CFX Command Language (CCL)

16.0 Release



Introduction to ANSYS CFX

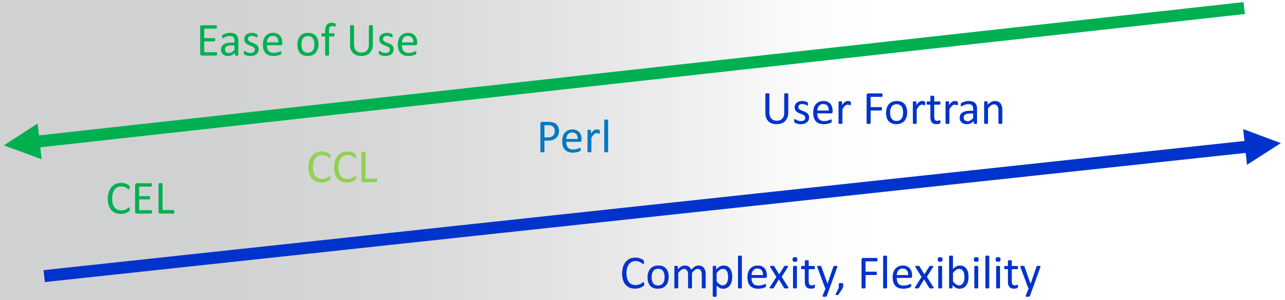
Introduction

- **Lecture Theme:**
  - It is possible to create user variables, expressions and functions with which to customize a CFD model, e.g. physical properties of fluids, physical models. With a knowledge of the underlying command language, it is possible to make changes more quickly.
- **Learning Aims:**
  - You will learn:
    - how to create Additional Variables (user variables)
    - how to set up equations and functions using CFX Expression Language (CEL)
    - the structure of CFX Command Language (CCL) and where it is used

so that you can extend your use of the software to deal with a greater variety of conditions and work more efficiently.

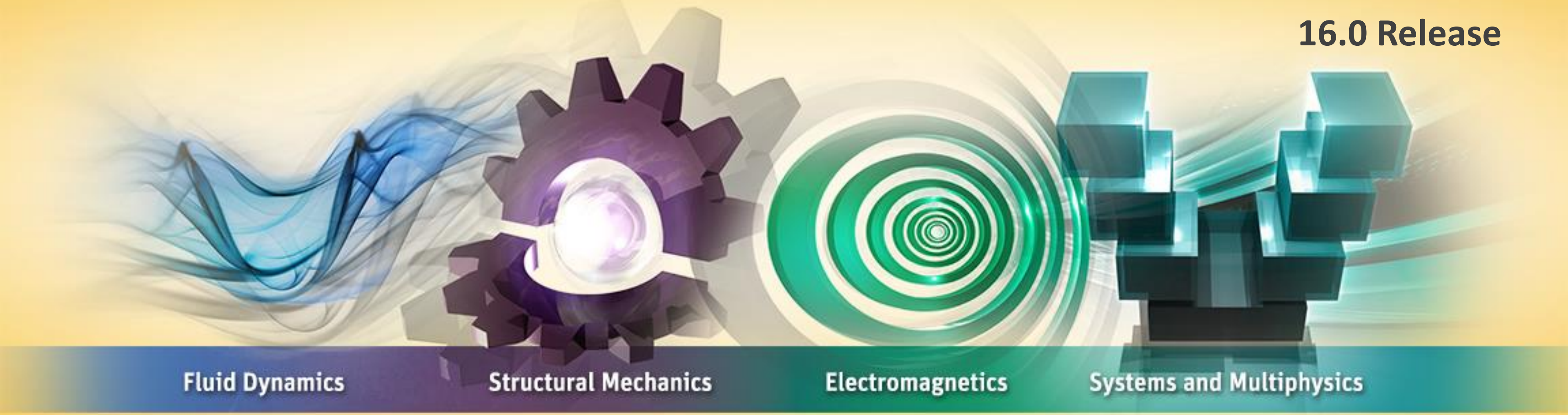
Customization in CFX

- **CFX can be customized using:**
  - Custom Variables (Additional Variables, or “AVs”)
  - CFX Expression Language (CEL)
  - CFX Command Language (CCL)
  - Perl scripts
  - User Fortran Routines



Additional Variables (AVs)

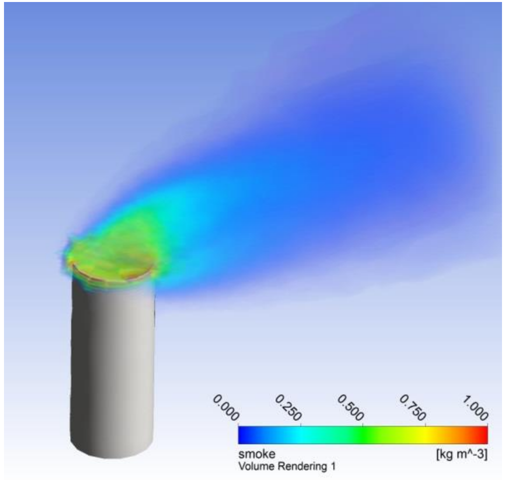
16.0 Release



Introduction to ANSYS CFX

Additional Variables

- Additional Variables (AVs) are non-reacting scalar components that do not directly influence the flow
- They can be solved for using a transport equation or an algebraic expression
  - Transported AVs require boundary and initial conditions
- Examples:
  - A tracer such as a dye or smoke
    - *Transport* Additional Variable. The AV is transported with the flow, but does not influence the flow
  - pH level
    - *Algebraic* Additional Variable. A function of other variables



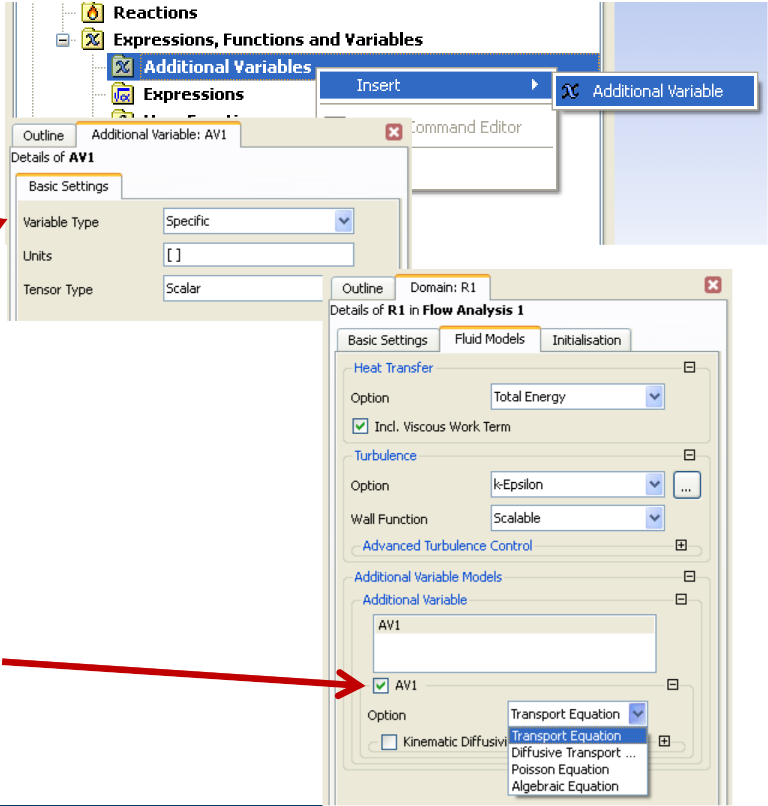
Additional Variables

- **Additional Variables can be used to work-around some limitations:**
  - In CFX-Pre integrated CEL functions, e.g. `areaAve()` cannot operate on an expression
    `areaAve(Velocity * Density)@Inlet` is not valid.
    Instead create an Algebraic AV equal to the expression and pass the AV to the function

- **An Additional Variable can be used to show the variation in age of the fluid in the domain to indicate stagnant regions, for example**
  - This is done by creating a transport AV "Age" with units of [s] and with Type = Volumetric (see next slide)
    - Inlet and initial values are zero
    - A source term with a value of 1 is set for the AV throughout the domain

Additional Variables

- **To create an AV right-click on Expressions, Functions and Variables > Additional Variables, or use the toolbar**
  - **Variable Type**
    - **Specific: solved on per-unit-mass basis**
    - **Volumetric: solved on per-unit-volume basis**
    - **Unspecified: defined by algebraic expression**
  - **Units: describe the additional variable**
  - **Tensor Type: Scalar or Vector as necessary**



- **The AV must then be switched on in the Domain**
  - **On Fluid Models or Solid Models tab**

Domain – Equation Options

- **Transport Equation**
  - **Kinematic Diffusivity - laminar diffusion**
  - **Turbulent diffusion always included**

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla \bullet (\rho U \phi) = \nabla \bullet (\rho D_{\Phi} \nabla \phi) + S_{\phi}$$

Transient | Advection | Diffusion | Sources

- **Diffusive Transport Equation**

$$\frac{\partial(\rho\phi)}{\partial t} = \nabla \bullet (\rho D_{\Phi} \nabla \phi) + S_{\phi}$$

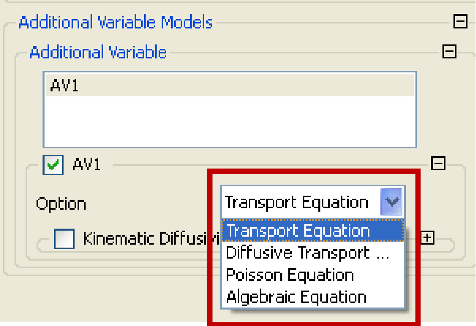
Transient | Diffusion | Sources

- **Poisson Equation**
  - **Used in electromagnetics**

$$0 = \nabla \bullet (\rho D_{\Phi} \nabla \phi) + S_{\phi}$$

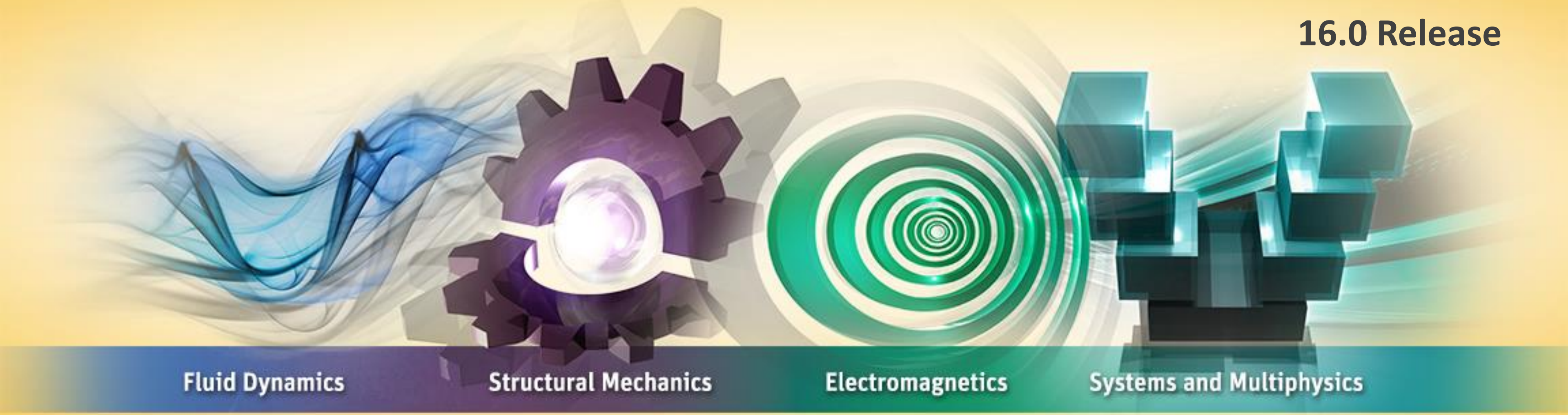
Diffusion | Sources

- **Algebraic or Vector Algebraic Equation**
  - **Vector - expression for each component**



CFX Expression Language (CEL)

16.0 Release

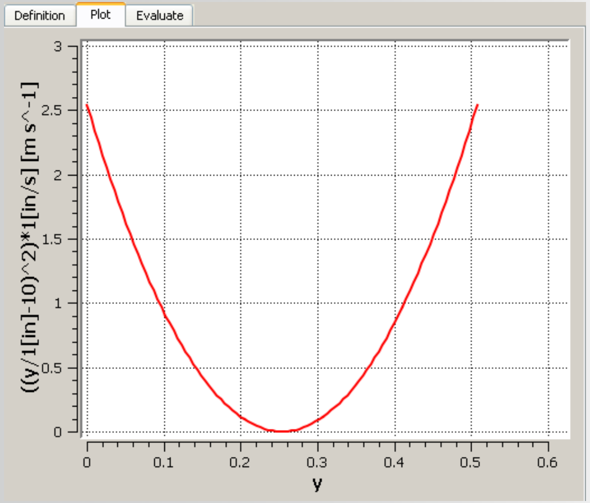


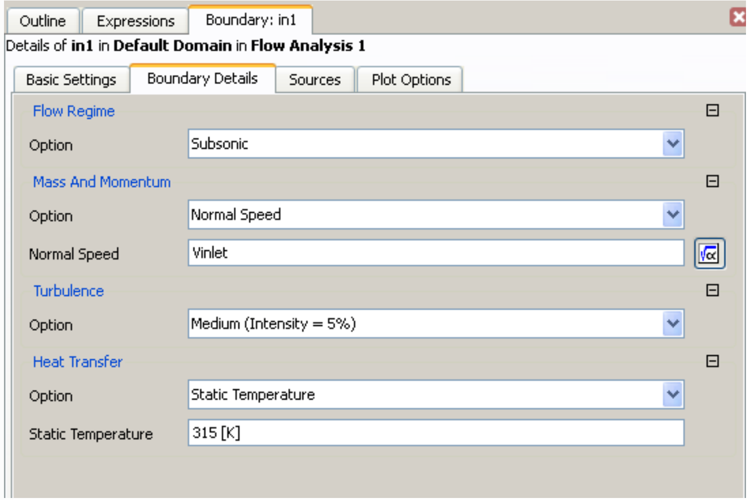
Introduction to ANSYS CFX

Realize Your Product Promise®

CEL

- **CEL - CFX Expression Language**
  - **Allows the user to create equations that can be functions of solution/system variables and can be used in CFX-Pre and CFD-Post**
  - **Example:** Vinlet ((y / 1 [in]-10) ^ 2)* 1 [in/s]





CEL Rules

- **The syntax rules are the same as those for conventional arithmetic.**
- **Operators are written as:**

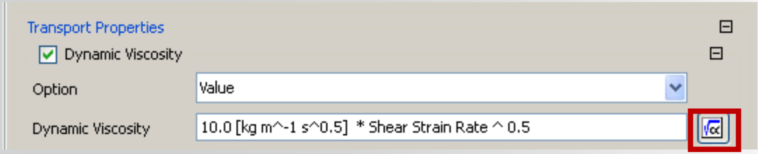
  + (addition) - (subtraction) * (multiplication)
  / (division) ^ (exponentiation)

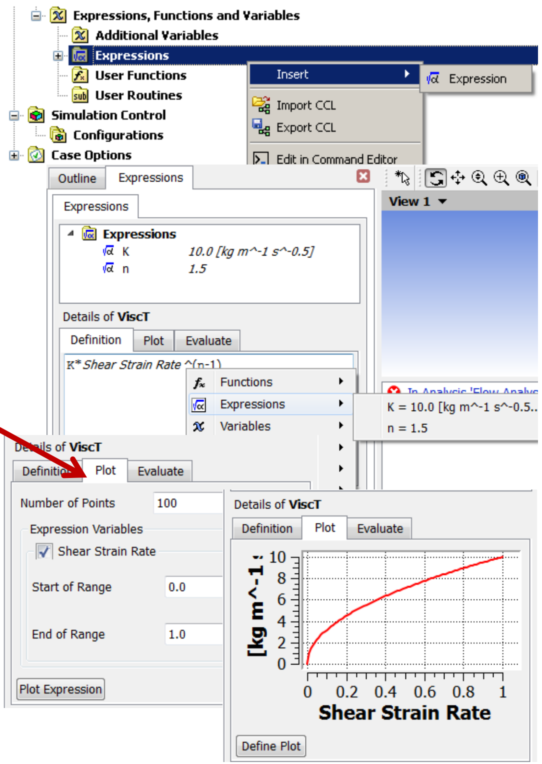
- **Variables and expressions are case sensitive, e.g. t vs T**
- **Availability of variables depends on physics, e.g. t only in transient models**
- **Can mix units but must be dimensionally consistent for addition and subtraction operations (example: *1.0 [mm] + 0.45 [yds]* is OK)**
- **Some constants are available in CEL, e.g.** `e, g, pi, R`

CEL - Creating Expressions

- **Expression Editor**
  - Create a library of expressions
  - Right-click in Definition window for drop-down lists of variables, functions, locations, constants…
  - Plot and evaluate expression to check behaviour

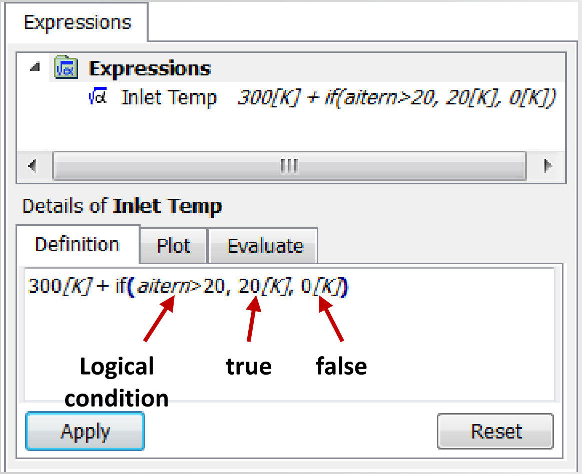
- **Alternatively, expressions can be entered directly where used**

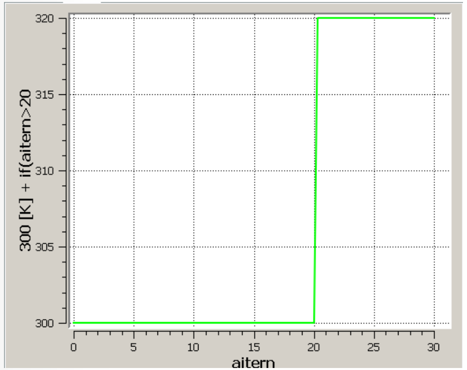




CEL – Conditional if Statement

- Using an "if" function
  - Example: set inlet temperature to 300 K for the first 19 iterations then raise it to 320 K after 20 iterations



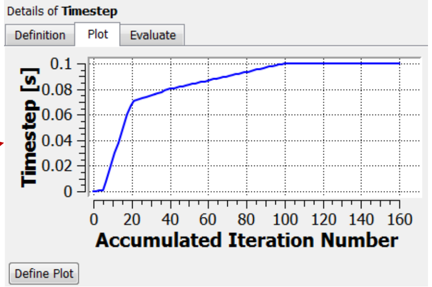


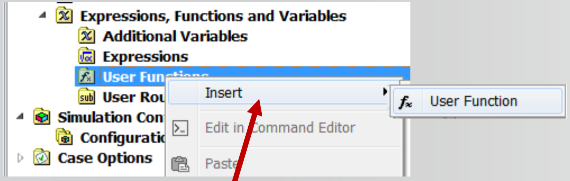
Note: On the 20th iteration inlet temp = 310 K

  - CEL Appendix contains some other useful functions

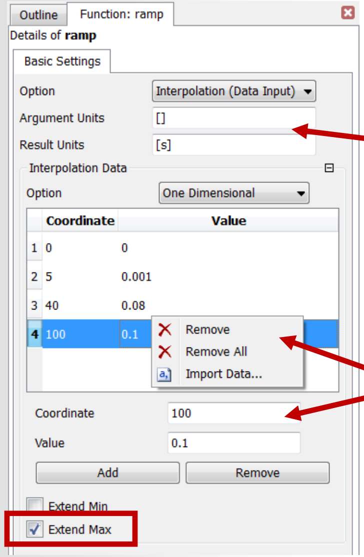
User Functions: Example

- 1D linear and 3D interpolation functions
- Example: Timescale a function of iteration number



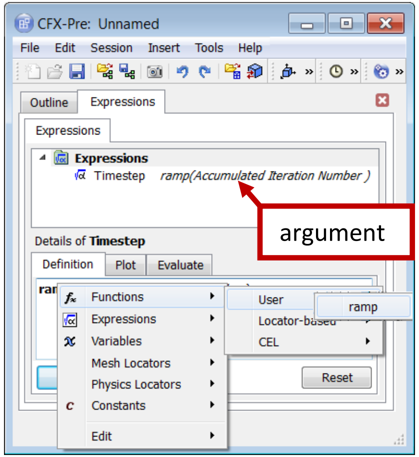


right-click to insert new function or use toolbar icon



set units for input & output

import data points or add manually



argument

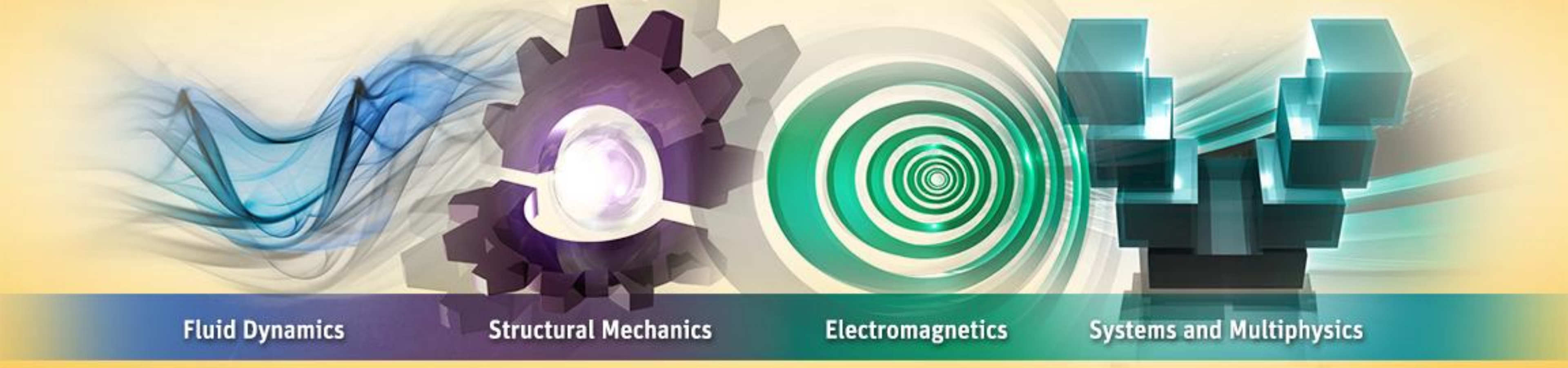
Introduction | AVs | CEL | CCL | Summary

Integrated Quantities

- Functions to evaluate a quantity on a location, e.g. *volumeInt(Density)@tank*
- Available in CFX-Pre and CFD-Post
  - in CFX-Pre the argument must be a variable and not an expression
- Some functions apply to a variable and some don't
  - area average of Cp on an isosurface: *areaAve(Cp)@iso1*
  - mass flow of particular fluid through a boundary: *oil.massFlow()@inlet*
- If location is a general mesh region the syntax is @REGION:<name>
- Phases/components: [<phase name>.][<component name>.]<function>@<locator>
  - *Air.Nitrogen.massFlow()@outlet*
- For vector functions a component and a local coordinate frame can be specified:
  - *area_x()@boundary* area projected in the x-direction
  - force_z_MyCoord()@wall z component of force on wall in coordinate frame "MyCoord"

CFX Command Language (CCL)

16.0 Release



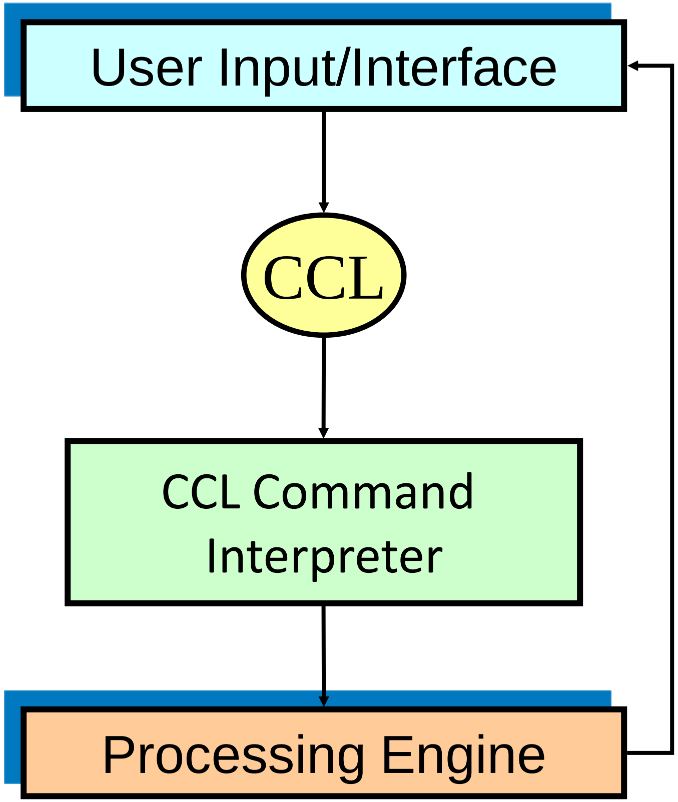
Introduction to ANSYS CFX

What is CCL?

- **CFX Command Language**
  - **object definition language underling all CFX products**
  - **object parameters define the “state” of an object**
  - **GUIs present object parameters in a contextual manner (parameters shown/hidden based on GUI selections)**

- **Allows for easy modifications**
  - **E.g. modifying an inlet velocity at a boundary**

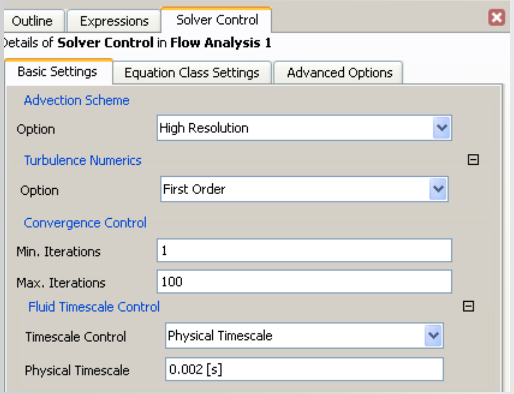
- **Advanced functionality (without complexity)**
  - **i.e. loops and logic**

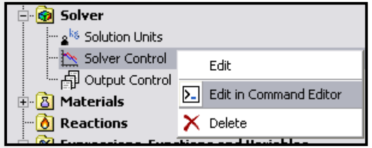


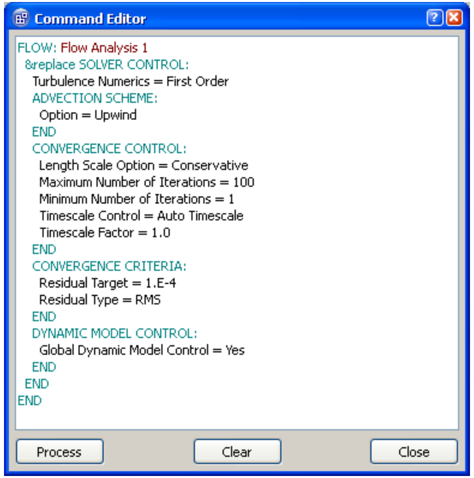
CCL Instruction Types

- CCL Statements can be classified into three categories:
  1. Object and parameter definitions
  2. Actions
  3. Power Syntax

1. Parameter Definition:



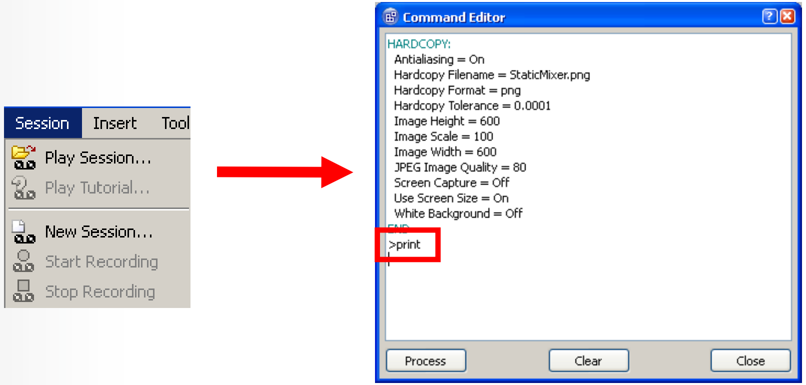




CCL Instruction Types

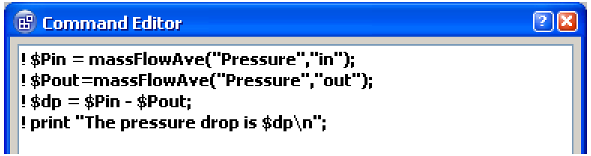
2. Action Command:

– E.g. a CFD-Post session file can include actions such as >print

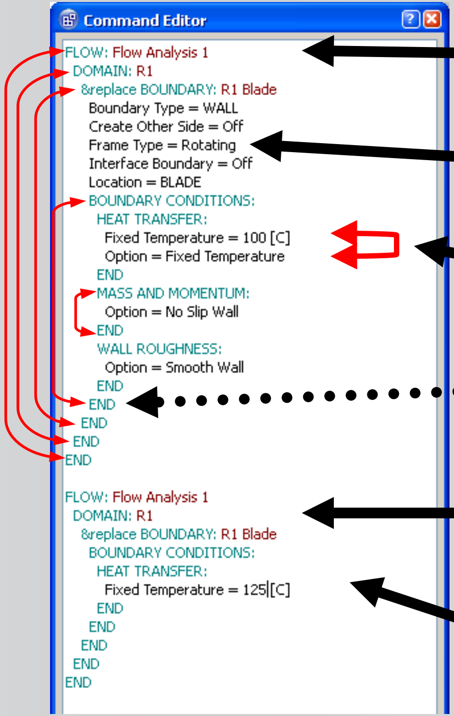


3. Power Syntax:

– Power Syntax commands are identified in CCL by the special character “!” at the start of each line.



Data Hierarchy



Terminology: The object *FLOW* is the parent of the object *DOMAIN*, and this is *Flow Analysis 1*

Terminology: Parameter Name = Parameter Value

As long as the parameters are of the same object, order is unimportant

An object started must be ended

CCL follows a data hierarchy. In order to specify the Temperature, the associated parents (Boundary, Domain, Flow) must be included

If data is set in one place and modified in another, the latter definition overrides the first

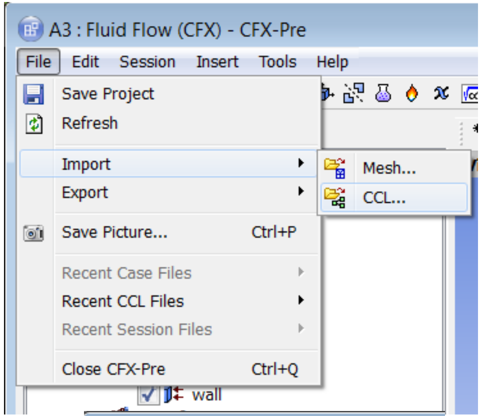
CCL Syntax

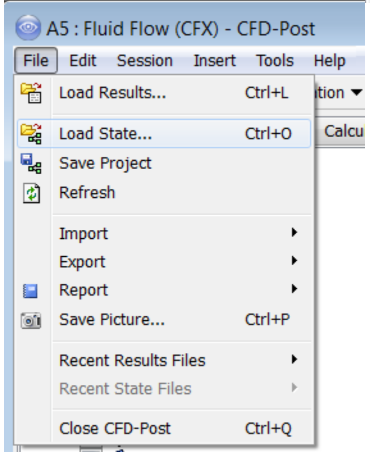
Similar rules to many programming languages. Some rules in the CCL Appendix and complete list in the Help documentation (advanced search “Simple Syntax Details” for the CFX application). Main points are:

- **Case Sensitivity**
  - Everything in the file is sensitive to case
- **Effect of spaces**
  - Spaces before or after a name are not part of the name
  - Single spaces inside a name are significant
  - Multiple spaces and tabs inside a name become a single space
- **CCL names definition**
  - First character must be alphabetic
  - Subsequent characters can be any number of be alphabetic, numeric , space

CCL Usage

- CCL can increase efficiency
  - Frequently used physics definitions can be saved to a text file and imported into CFX-Pre
  - Settings in CFD-Post can be saved to a state file for repeated use. A state file is created by default in Workbench
  - CCL Appendix shows further options





Introduction | AVs | CEL | CCL | Summary

Summary

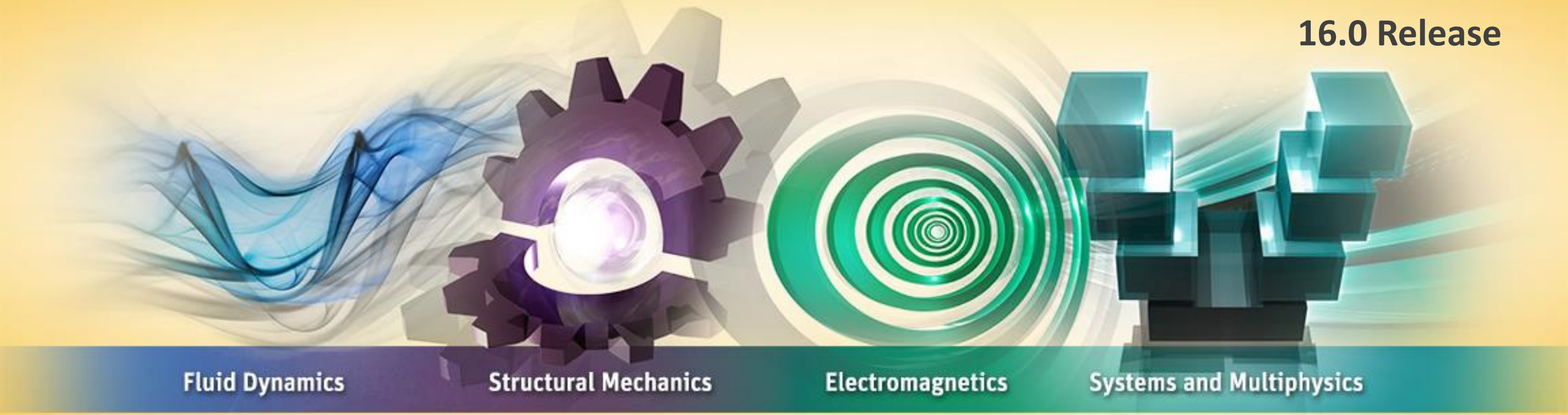
- **Customization is possible with the addition of user variables, expressions and functions**

- **CCL is the language which links the user with the software.**

- **Sometimes it is more efficient to bypass the GUI and manipulate the CCL manually**

Appendix CEL

16.0 Release



Introduction to ANSYS CFX

Useful Functions

The *inside()* function returns 1 when inside the specified location and 0 when outside

- Useful to limit the scope of a function to a subdomain or boundary

The *step()* function return 1 when the argument is positive and 0 when the argument is negative

- Useful as an on-off switch
- *if()* function can also be used as a switch

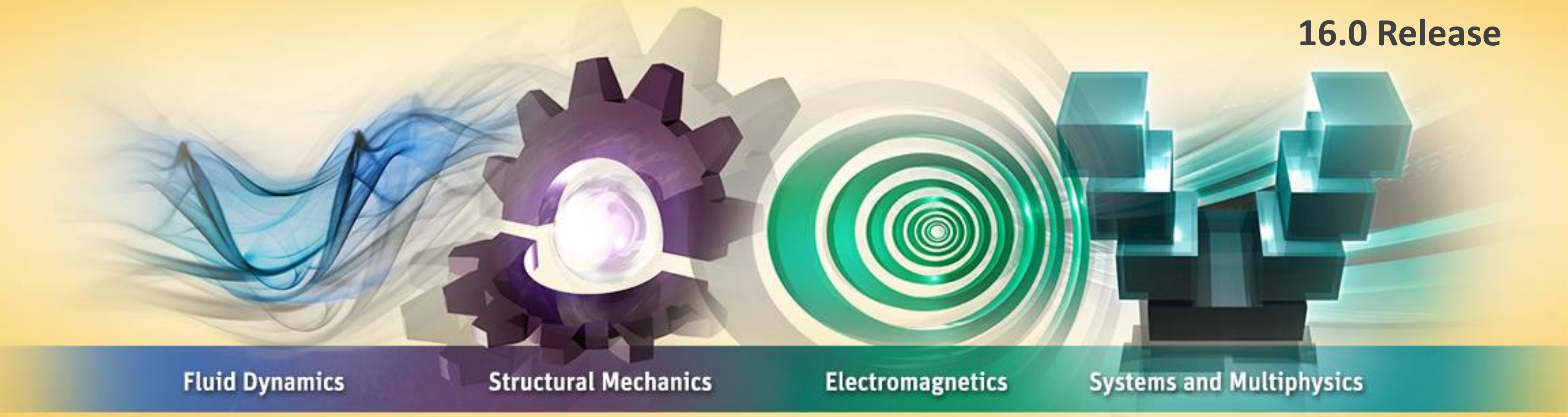
***areaAve()* and *massFlowAve()* are used to evaluate the average of a quantity on a location**

- *areaAve()* is an area-weighted average. It is usually used on wall boundaries and when the quantity is not "carried with the flow", e.g. Pressure at an outlet, Temperature on a wall
- *massFlowAve()* is an average weighted by the local mass flow. It is usually used to evaluate quantities that are "carried with the flow", e.g. Temperature at an outlet

ANSYS

Appendix CCL

16.0 Release



Introduction to ANSYS CFX

CCL Examples

CCL is used throughout CFX and other CFX products, a few examples are:

- **CFX-Pre: Physics definition, Session Files**
- **CFX-Solve: Command File (echoed in the OUT file)**
- **CFX-Post : State files, Session Files**
- **CFX-TurboGrid: State Files**

Syntax Rules

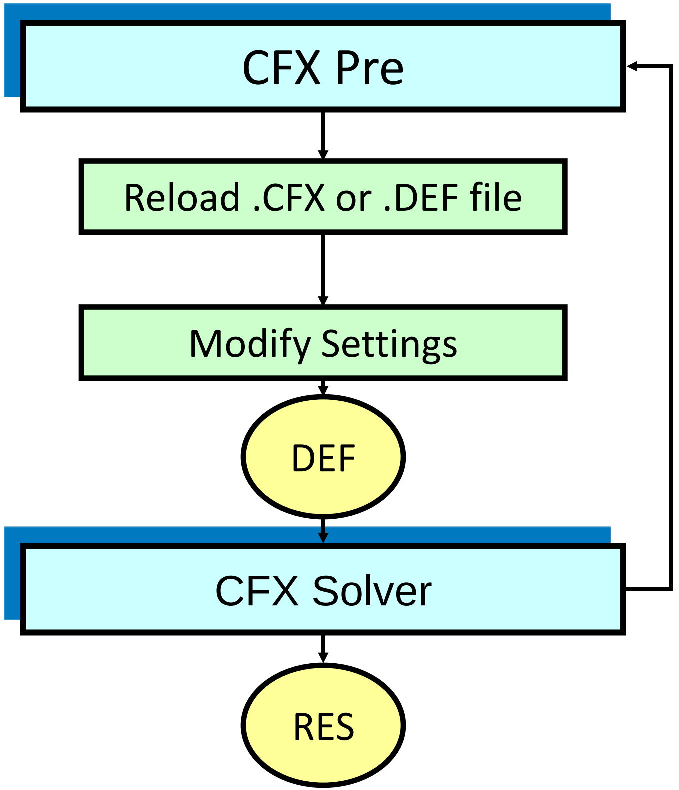
- **Parameter Values**
  - STRING
    - A string beginning with $ is a Power Syntax (Perl) variable
    - Following a $, the characters [,],{ and } terminate the preceding Perl variable name
    - A string beginning with # is a comment
  - STRING LIST
    - A list of string items separated by commas, e.g names = one, two, three, four
  - INTEGER
    - If a real is specified when an integer is needed, the real is rounded to the nearest integer.
  - REAL LIST
    - All items in the list must have the same dimensions
  - LOGICAL
    - YES/NO, TRUE/FALSE, 1/0 or ON/OFF are all accepted as are initial letter variants Y, T, N, F

- **Continuation character **

Quick Modifications

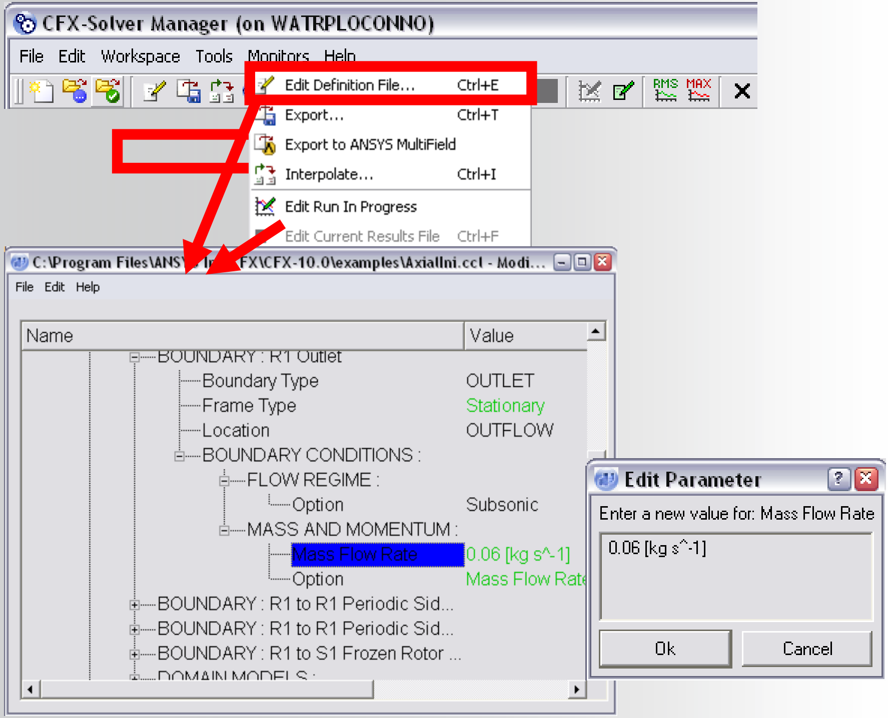
- Users can modify a .def file manually using CCL instead of using the CFX-Pre GUI

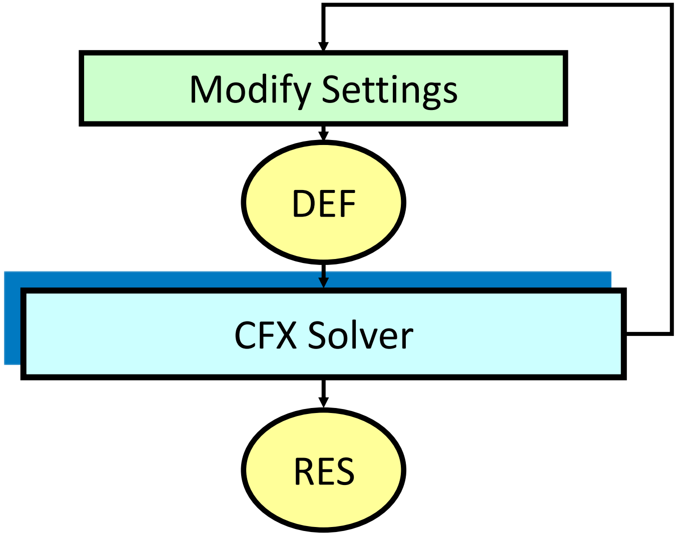
- Two alternate methods of modifying a DEF file:
  - Solver GUI
  - Command Lines



Quick Modifications

SOLVER METHOD





Quick Modifications

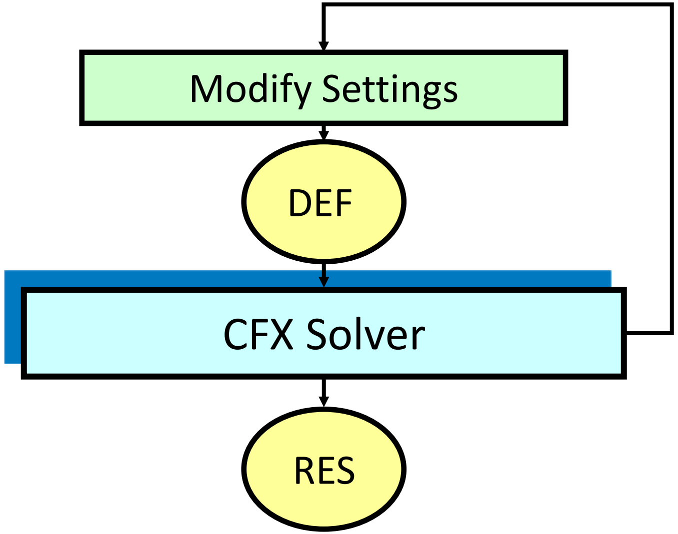
COMMAND LINE METHOD

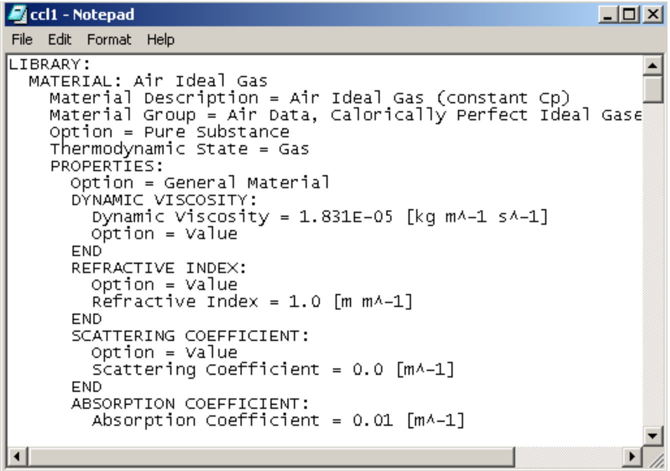
1. In the CFX-Launcher, click “Tools > Command Line”
2. In command screen type:

   cfx5cmds –read –def *filename*.def –text *ccl1*

3. Edit *ccl1* in notepad and save
4. In command screen type:

   cfx5cmds –write –def *filename*.def –text *ccl1*






```
LIBRARY:
  MATERIAL: Air Ideal Gas
    Material Description = Air Ideal Gas (constant Cp)
    Material Group = Air Data, Calorically Perfect Ideal Gase
    Option = Pure Substance
    Thermodynamic State = Gas
    PROPERTIES:
      Option = General Material
      DYNAMIC VISCOSITY:
        Dynamic Viscosity = 1.831E-05 [kg m^-1 s^-1]
        Option = Value
      END
      REFRACTIVE INDEX:
        Option = Value
        Refractive Index = 1.0 [m m^-1]
      END
      SCATTERING COEFFICIENT:
        Option = Value
        Scattering Coefficient = 0.0 [m^-1]
      END
      ABSORPTION COEFFICIENT:
        Absorption Coefficient = 0.01 [m^-1]
```


Quick Modifications

- **Using a CCL file**
  - Create a text file with modified CCL
  - “Save As” .ccl file, (e.g. “bc1.ccl”)
  - On the *Solver* tab of the Run Definition form in the Solver manager, enter an *Argument* as follows:

  -ccl bc1.ccl

  - Or, start the solution from the command line, using:

  cfx5solve –def run.def –ccl bc1.ccl

```
FLOW: Flow Analysis 1
 DOMAIN: R1
  &replace BOUNDARY: R1 Blade
   Boundary Type = WALL
   Create Other Side = Off
   Frame Type = Rotating
   Interface Boundary = Off
   Location = BLADE
   BOUNDARY CONDITIONS:
    HEAT TRANSFER:
     Fixed Temperature = 100 [C]
     Option = Fixed Temperature
    END
    MASS AND MOMENTUM:
     Option = No Slip Wall
    END
    WALL ROUGHNESS:
     Option = Smooth Wall
    END
   END
  END
 END
END
```