

Plasma Module

User's Guide

Plasma Module User's Guide

© 1998–2016 COMSOL

Protected by U.S. Patents listed on www.comsol.com/patents, and U.S. Patents 7,519,518; 7,596,474; 7,623,991; 8,219,373; 8,457,932; 8,954,302; 9,098,106; 9,146,652; and 9,323,503. Patents pending.

This Documentation and the Programs described herein are furnished under the COMSOL Software License Agreement (www.comsol.com/comsol-license-agreement) and may be used or copied only under the terms of the license agreement.

COMSOL, the COMSOL logo, COMSOL Multiphysics, Capture the Concept, COMSOL Desktop, LiveLink, and COMSOL Server are either registered trademarks or trademarks of COMSOL AB. All other trademarks are the property of their respective owners, and COMSOL AB and its subsidiaries and products are not affiliated with, endorsed by, sponsored by, or supported by those trademark owners. For a list of such trademark owners, see www.comsol.com/trademarks.

Version: COMSOL 5.2a

Contact Information

Visit the Contact COMSOL page at www.comsol.com/contact to submit general inquiries, contact Technical Support, or search for an address and phone number. You can also visit the Worldwide Sales Offices page at www.comsol.com/contact/offices for address and contact information.

If you need to contact Support, an online request form is located at the COMSOL Access page at www.comsol.com/support/case. Other useful links include:

- Support Center: www.comsol.com/support
- Product Download: www.comsol.com/product-download
- Product Updates: www.comsol.com/support/updates
- COMSOL Blog: www.comsol.com/blogs
- Discussion Forum: www.comsol.com/community
- Events: www.comsol.com/events
- COMSOL Video Gallery: www.comsol.com/video
- Support Knowledge Base: www.comsol.com/support/knowledgebase

Part number: CM021401

C o n t e n t s

Chapter 1: Introduction

About the Plasma Module	14
Why Plasma is Used in Modeling	14
How the Plasma Module Helps Improve Your Modeling	15
Plasma Module Nomenclature	16
The Plasma Module Physics Interface Guide.	18
Common Physics Interface and Feature Settings and Nodes.	19
Where Do I Access the Documentation and Application Libraries?	20
 Overview of the User's Guide	 24

Chapter 2: Data Required for Plasma Modeling

Data Requirements	28
Electron Impact Reactions	28
Reaction	30
Surface Reaction	30
Species	30
 Importing Collision Cross-Section Data	 33
Cross-Section Data File Format	33
The File Format	34
References for the Plasma Module Cross-Section Data Requirements . . .	36

Chapter 3: AC/DC Interfaces

The Electrostatics Interface	38
Domain, Boundary, Edge, Point, and Pair Nodes for the Electrostatics Interface	40
Charge Conservation	43

Initial Values.	44
Space Charge Density	44
Zero Charge	44
Ground	44
Electric Potential	45
Surface Charge Density	46
External Surface Charge Accumulation	47
Electric Displacement Field	47
Periodic Condition	48
Thin Low Permittivity Gap	48
Surface Charge Accumulation	49
Dielectric Shielding	50
Terminal	50
Floating Potential	52
Distributed Capacitance	53
Line Charge.	54
Line Charge (on Axis)	54
Line Charge (Out-of-Plane)	55
Point Charge	55
Point Charge (on Axis)	56
Charge Cross Section	57
Charge Thickness (Out-of-Plane).	57
Charge Conservation, Piezoelectric.	58
Metal Contact	58
Dielectric Contact	59
Electrostatic Point Dipole	60

The Electric Currents Interface 61

Domain, Boundary, Edge, Point, and Pair Nodes for the Electric Currents Interface	63
Current Conservation	65
Initial Values.	67
External Current Density.	67
Current Source	68
Electric Insulation	68
Floating Potential	69
Boundary Current Source	70
Normal Current Density	70

Distributed Impedance	71
Terminal	72
Contact Impedance	75
Sector Symmetry	76
Line Current Source	77
Line Current Source (on Axis)	78
Point Current Source	78
Point Current Source (on Axis)	79
Piezoresistive Material	79

The Electrical Circuit Interface 81

Ground Node	83
Resistor	83
Capacitor	84
Inductor	84
Voltage Source	85
Current Source	86
Voltage-Controlled Voltage Source	87
Voltage-Controlled Current Source	87
Current-Controlled Voltage Source	88
Current-Controlled Current Source	88
Subcircuit Definition	89
Subcircuit Instance	89
NPN BJT	89
PNP BJT	90
n-Channel MOSFET	90
p-Channel MOSFET	91
Mutual Inductance	92
Transformer	92
Diode	93
External I vs. U	93
External U vs. I	94
External I-Terminal	95
SPICE Circuit Import	96
SPICE Circuit Export	96

Theory for the Electrostatics Interface 98

Charge Relaxation Theory	98
------------------------------------	----

Electrostatics Equations	100
The Electrostatics Interface in Time Dependent or Frequency Domain Studies	101
Theory for the Electric Currents Interface	102
Electric Currents Equations in Steady State	102
Dynamic Electric Currents Equations	103
Theory for the Electrical Circuit Interface	104
Electric Circuit Modeling and the Semiconductor Device Models.	104
Bipolar Transistors	105
MOSFET Transistors	108
Diode	111
Reference for the Electrical Circuit Interface	113
Connecting to Electrical Circuits	114
About Connecting Electrical Circuits to Physics Interfaces	114
Connecting Electrical Circuits Using Predefined Couplings	115
Connecting Electrical Circuits by User-Defined Couplings	115
Solving.	116
Postprocessing.	117
Spice Import	118
SPICE Import	118
SPICE Export	119
Reference for SPICE Import and Export	119

Chapter 4: Fluid Flow Interface

The Single-Phase Flow, Laminar Flow Interface	122
The Laminar Flow Interface	122
Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow	125
Fluid Properties	126
Volume Force	127
Initial Values.	127
Wall.	128

Inlet	130
Outlet	133
Symmetry	134
Open Boundary	135
Boundary Stress	136
Vacuum Pump	137
Periodic Flow Condition	138
Pipe Connection	139
Flow Continuity	139
Pressure Point Constraint	140
Point Mass Source	140
Line Mass Source	141
Gravity	142

Theory for the Single-Phase Flow Laminar Interface 143

General Single-Phase Flow Theory	144
Compressible Flow	146
Weakly Compressible Flow	146
The Mach Number Limit	146
Incompressible Flow	147
The Reynolds Number	148
The Boussinesq Approximation	148
Theory for the Wall Boundary Condition	149
Prescribing Inlet and Outlet Conditions	152
Laminar Inflow	154
Laminar Outflow	154
Mass Flow	155
No Viscous Stress	156
Normal Stress Boundary Condition	157
Pressure Boundary Condition	157
Vacuum Pump Boundary Condition	159
Mass Sources for Fluid Flow	160
Numerical Stability—Stabilization Techniques for Fluid Flow	162
Solvers for Laminar Flow	164
Pseudo Time Stepping for Laminar Flow Models	166
Discontinuous Galerkin Formulation	168
Particle Tracing in Fluid Flow	168
References for the Single-Phase Flow, Laminar Flow Interfaces	169

Chapter 5: The Boltzmann Equation, Two-Term Approximation Interface

The Boltzmann Equation, Two-Term Approximation Interface	172
Domain, Boundary, and Pair Nodes for the Boltzmann Equation, Two-Term Approximation Interface	175
Boltzmann Model.	176
Initial Values.	177
Collision	178
Boundary Settings	179
Zero Energy Flux.	179
Zero Probability	179
 Theory for the Boltzmann Equation, Two-Term Approximation Interface	 180
The Electron Energy Distribution Function	180
Boltzmann Equation, Two-Term Approximation	181
References for the Boltzmann Equation, Two-Term Approximation Interface.	187

Chapter 6: The Drift Diffusion Interface

The Drift Diffusion Interface	190
Domain, Boundary, and Pair Nodes for the Drift Diffusion Interface . . .	192
Drift Diffusion Model	193
Initial Values.	195
Electron Production Rate.	196
General Power Deposition	196
Velocity	196
Inductive Power Deposition.	197
Microwave Power Deposition	197
Insulation.	197
Electron Density and Energy	197
Wall.	198
Flux.	199

Electron Outlet	200
Theory for the Drift Diffusion Interface	201
Introduction to Drift Diffusion Theory	201
Electron Transport Theory	203
Source Coefficients	205
Stabilization	206
Tensor Transport Properties	206
The Wall Boundary Condition	207
The Insulation Boundary Condition	208
The Electron Density and Energy Boundary Condition	209
Reference for the Drift Diffusion Interface	209

Chapter 7: The Heavy Species Transport Interface

The Heavy Species Transport Interface	212
Domain, Boundary, and Pair Nodes for the Heavy Species Transport Interface.	216
Cross Section Import	217
Diffusion and Migration	217
Electron Impact Reaction.	218
Reaction	222
Species	223
Surface Reaction	227
Surface Species	228
Flux	229
Inlet.	229
Outflow	230
Symmetry	231
Theory for the Heavy Species Transport Interface	232
About the Transport Equations	232
Multicomponent Diffusion Equations	233
Additional Quantities of Interest	235
Auxiliary Relations	236
Source Coefficients	236

Thermodynamic Properties	237
Species Transport Properties	238
Mixture Transport Properties	240
Scaling	241
Stabilization	242
Surface Reactions	242
Flux Matching	243
About Surface Species	245
Limitations of the Heavy Species Transport Equations	246
References for the Heavy Species Transport Interface	246

Chapter 8: Plasma Interfaces

The Plasma Interface	250
Domain, Boundary, Edge, Point, and Pair Nodes for the Plasma Interface.	256
Plasma Model	257
Initial Values.	259
 The Inductively Coupled Plasma Interface	 261
Plasma Conductivity Coupling	262
Electron Heat Source	262
Collisionless Heating	262
 The Microwave Plasma Interface	 264
 Plasma Reactors Theory	 265
 Capacitively Coupled Plasmas	 268
Computing the DC Bias	269
 DC Discharges	 271
Introduction to DC Discharge Theory.	271
DC Discharge Boundary Conditions	272
Tips for Modeling DC Discharges	273

Theory for Inductively Coupled Plasmas	274
Introduction to Inductively Coupled Plasmas	274
Domain Equations for the Inductively Coupled Plasma Interface	276
 Theory for Microwave Plasmas	 278
Introduction	278
Wave Heated Discharge Theory	278
Plasma Conductivity in the Presence of a Static Magnetic Field.	285
References	286

Chapter 9: Equilibrium Discharges Interfaces

The Equilibrium DC Discharge Interface	288
Equilibrium Discharge Heat Source	290
Flow Coupling	292
Equilibrium Discharge Boundary Heat Source	293
 The Combined Inductive/DC Discharge Interface	 295
Static Current Density Component	298
Induction Current Density Component	299
Lorentz Force	300
 The Equilibrium Inductively Coupled Plasma Interface	 302
 Equilibrium Discharges Theory	 306
 Summary of the Equilibrium Discharge Interfaces Features	 312
References	312

Introduction

This guide describes the Plasma Module, an optional add-on package that extends the COMSOL Multiphysics® modeling environment with customized physics interfaces and functionality optimized for the analysis of non-equilibrium discharges.

This chapter introduces you to the capabilities of this module. A summary of the physics interfaces and where you can find documentation and model examples is also included. The last section is a brief overview with links to each chapter in this guide.

- [About the Plasma Module](#)
- [Overview of the User's Guide](#)

About the Plasma Module

The Plasma Module provides a set of user interfaces which are used to define the physics, boundary conditions and material properties which collectively constitute a plasma. Electron transport is modeled with a pair of drift-diffusion equations, one for the electron density and another for the electron energy. Motion of the non-electron species is governed by a modified form of the Maxwell-Stefan equations. Poisson's equation is solved to compute the plasma potential. Additional heating mechanisms, such as induction heating or wave heating can be included by solving Ampere's law or the wave equation.

In this section:

- [Why Plasma is Used in Modeling](#)
- [How the Plasma Module Helps Improve Your Modeling](#)
- [Plasma Module Nomenclature](#)
- [The Plasma Module Physics Interface Guide](#)
- [Common Physics Interface and Feature Settings and Nodes](#)
- [Where Do I Access the Documentation and Application Libraries?](#)



[The Physics Interfaces and Building a COMSOL Multiphysics Model in the *COMSOL Multiphysics Reference Manual*](#)

Why Plasma is Used in Modeling

Low-temperature plasmas represent the amalgamation of fluid mechanics, reaction engineering, physical kinetics, heat transfer, mass transfer, and electromagnetics. The Plasma Module is a specialized tool for modeling non-equilibrium discharges which occur in a wide range of engineering disciplines. There are specialized physics interfaces for the most common types of plasma reactors including inductively coupled plasmas (ICP), DC discharges, wave heated discharges (microwave plasmas), and capacitively coupled plasmas (CCP).

Modeling the interaction between the plasma and an external circuit is an important part of understanding the overall characteristics of a discharge. This module has tools to add circuit elements directly to a 1D, 2D, or 3D model Component node, or import an existing SPICE netlist into the model. The plasma chemistry is specified by loading

in sets of collision cross sections from a file or by adding reactions and species to the Model Builder. The complicated coupling between the different physics that constitute a plasma is automatically handled by the included multiphysics interfaces.

How the Plasma Module Helps Improve Your Modeling

The Plasma Module is based on a series of scientific publications on numerical modeling of non-equilibrium discharges. These papers are referenced throughout the documentation. The physics interfaces can model low temperature, non-equilibrium discharges such as:

- Inductively coupled plasmas (ICP)
- Capacitively coupled plasmas (CCP)
- Microwave plasmas
- Light sources
- Electrical breakdown
- Space thrusters
- DC discharges
- Chemical vapor deposition (CVD)
- Plasma-enhanced chemical vapor deposition (PECVD)
- Dielectric barrier discharges (DBD)
- Electron cyclotron resonance (ECR)
- Reactive gas generators

The complexity of plasma modeling lies in the fact that it combines elements of reaction engineering, statistical physics, fluid mechanics, physical kinetics, heat transfer, mass transfer, and electromagnetics. The net result is a true multiphysics problem involving complicated coupling between the different physics. The module is designed to simplify the process of setting up a self-consistent model of a low-temperature plasma.

The physics interfaces include all the necessary tools to model plasma discharges, beginning with a Boltzmann Equation, Two-Term Approximation solver that computes the electron transport properties and source coefficients from a set of electron impact collision cross sections. This physics interface makes it possible to determine many of the interesting characteristics of a discharge by providing input properties such as the electric field and the electron impact reactions that make up the plasma chemistry, without solving a space-dependent problem.

For space-dependent models, the reactions and species which make up the plasma chemistry are conveniently managed in the Model Builder. When the fluid velocity and gas temperature are of interest, there are physics interfaces available for laminar flow and heat transfer. There are several options available when coupling the charged species

transport to the electromagnetic fields. Poisson's equation for the electrostatic potential is always solved. Modeling inductively coupled plasmas where induction currents are responsible for sustaining the plasma requires the AC/DC Module. For wave-heated discharges (microwave plasmas), the RF Module is required.

Plasma Module Nomenclature

The Plasma Module has numerous variables and expressions used throughout the documentation and in COMSOL Multiphysics. [Table 1-1](#) lists the variable, its description and its SI unit.

TABLE 1-1: NOMENCLATURE USED THROUGHOUT THE PLASMA MODULE DOCUMENTATION.

VARIABLE	DESCRIPTION	SI UNIT
ρ	Gas density	kg/m ³
p	Pressure	Pa
T	Temperature	K
M_k	Molecular weight, species k	kg/mol
M_n	Mean molar mass	kg/mol
w_k	Mass fraction, species k	l
W_k	Log of the mass fraction of species k	-
x_k	Mole fraction, species k	l
\mathbf{V}_k	Multicomponent diffusion velocity, species k	m/s
\mathbf{j}_k	Diffusive flux vector, species k	kg/(m ² ·s)
D_{kj}	Binary diffusion coefficient	m ² /s
$D_{k,m}$	Mixture averaged diffusion coefficient	m ² /s
D_k^T	Multicomponent thermal diffusion coefficient	kg/(m·s)
$D_{k,f}$	Diffusion coefficient for species k	m ² /s
Z_k	Charge, species k	l
$\mu_{k,m}$	Mixture averaged mobility, species k	m ² /(V·s)
V	Electric potential	V
\mathbf{E}	Electric field	V/m
R	Universal gas constant	J/(mol·K)
c_k	Molar concentration, species k	mol/m ³
r_j	Reaction rate, reaction j	mol/(m ³ ·s)
k_j	Rate coefficient, reaction j	m ³ /(mol·s)

TABLE 1-1: NOMENCLATURE USED THROUGHOUT THE PLASMA MODULE DOCUMENTATION.

VARIABLE	DESCRIPTION	SI UNIT
R_k	Rate expression, species k	$\text{kg}/(\text{m}^3 \cdot \text{s})$
n_k	Number density, species k	l/m^3
v'_{kj}	Stoichiometric matrix, forward component	l
v''_{kj}	Stoichiometric matrix, reverse component	l
v_{kj}	Total stoichiometric matrix	l
q	Unit charge	C
k_B	Boltzmann constant	J/K
$R_{\text{surf},k}$	Surface rate expression for species k	$\text{mol}/(\text{m}^2 \cdot \text{s})$
γ_f	Forward sticking coefficient	l
γ_i	Secondary emission coefficient	l
ε_i	Mean energy of secondary electrons	V
ρ_k	Mass concentration, species k	kg/m^3
\mathbf{u}	Mass averaged fluid velocity vector	m/s
F	Faraday's constant	$\text{s} \cdot \text{A}/\text{mol}$
N_A	Avogadro constant	l/mol
\mathbf{v}^T	Thermal drift velocity	m/s
u	Mass averaged velocity field, x/r-component	m/s
v	Mass averaged velocity field, y/z-component	m/s
w	Mass averaged velocity field, z-component	m/s
σ	Surface charge density	C/m^2
Γ_e	Electron flux	$\text{l}/(\text{m}^2 \cdot \text{s})$
q_j	Surface reaction rate, reaction j	$\text{mol}/(\text{m}^2 \cdot \text{s})$
A_j	Frequency factor, reaction j	$\text{m}^3/(\text{mol} \cdot \text{s})$
β_j	Temperature exponent, reaction j	l
E_j	Activation energy, reaction j	J/mol
$f(\varepsilon)$	Electron energy distribution function (EEDF)	$\text{V}^{-3/2}$
ε	Electron energy	V
$\bar{\varepsilon}, \varphi$	Mean electron energy	V
μ_e	Electron mobility	$\text{m}^2/(\text{V} \cdot \text{s})$
N_n	Neutral number density	l/m^3
$\mu_e N_n$	Reduced electron mobility	$\text{l}/(\text{V} \cdot \text{m} \cdot \text{s})$

TABLE I-1: NOMENCLATURE USED THROUGHOUT THE PLASMA MODULE DOCUMENTATION.

VARIABLE	DESCRIPTION	SI UNIT
θ	Out-of-plane direction	rad
I_p	Plasma current	A

The Plasma Module Physics Interface Guide












The Plasma Module extends the functionality of the physics interfaces of the base package for COMSOL Multiphysics. The details of the physics interfaces and study types for the Plasma Module are listed in the table. The functionality of the COMSOL Multiphysics base package is given in the *COMSOL Multiphysics Reference Manual*.



In the *COMSOL Multiphysics Reference Manual*:

- [Studies and Solvers](#)
- [The Physics Interfaces](#)
- For a list of all the core physics interfaces included with a COMSOL Multiphysics license, see [Physics Interface Guide](#).

PHYSICS INTERFACE	ICON	TAG	SPACE DIMENSION	AVAILABLE PRESET STUDY TYPE
AC/DC				
Electrical Circuit		cir	Not space dependent	stationary; frequency domain; time dependent; small signal analysis, frequency domain
Electrostatics ¹		es	all dimensions	stationary; time dependent
Fluid Flow				
Single-Phase Flow				
Laminar Flow ¹		spf	3D, 2D, 2D axisymmetric	stationary; time dependent


PHYSICS INTERFACE	ICON	TAG	SPACE DIMENSION	AVAILABLE PRESET STUDY TYPE
 Plasma				
Drift Diffusion		dd	all dimensions	time dependent
Heavy Species Transport		hs	all dimensions	time dependent
Boltzmann Equation, Two-Term Approximation		be	1D	frequency domain; mean energies; reduced electric fields
Plasma		plas	all dimensions	time dependent
Inductively Coupled Plasma ^{2, 4}		—	3D, 2D, 2D axisymmetric.	frequency-transient
Microwave Plasma ^{3, 4}		—	3D, 2D, 2D axisymmetric	frequency-transient
 Equilibrium Discharges				
Equilibrium DC Discharge ⁴		—	3D, 2D, 2D axisymmetric	stationary; time dependent
Combined Inductive/ DC Discharge ^{2, 4}		—	3D, 2D, 2D axisymmetric	frequency-transient
Equilibrium Inductively Coupled Plasma ^{2, 4}		—	3D, 2D, 2D axisymmetric	frequency-transient
¹ This physics interface is included with the core COMSOL package but has added functionality for this module. ² 3D is available with the addition of the AC/DC Module. ³ Requires the addition of the RF Module. ⁴ This physics interface is a predefined multiphysics coupling that automatically adds all the physics interfaces and coupling features required.				

Common Physics Interface and Feature Settings and Nodes

There are several common settings and sections available for the physics interfaces and feature nodes. Some of these sections also have similar settings or are implemented in


the same way no matter the physics interface or feature being used. There are also some physics feature nodes that display in COMSOL Multiphysics.

In each module’s documentation, only unique or extra information is included; standard information and procedures are centralized in the *COMSOL Multiphysics Reference Manual*.

	In the <i>COMSOL Multiphysics Reference Manual</i> see Table 2-3 for links to common sections and Table 2-4 to common feature nodes. You can also search for information: press F1 to open the Help window or Ctrl+F1 to open the Documentation window.
---	---

Where Do I Access the Documentation and Application Libraries?

A number of internet resources have more information about COMSOL, including licensing and technical information. The electronic documentation, topic-based (or context-based) help, and the application libraries are all accessed through the COMSOL Desktop.


	If you are reading the documentation as a PDF file on your computer, the blue links do not work to open an application or content referenced in a different guide. However, if you are using the Help system in COMSOL Multiphysics, these links work to open other modules (as long as you have a license), application examples, and documentation sets.
---	--



THE DOCUMENTATION AND ONLINE HELP


The *COMSOL Multiphysics Reference Manual* describes the core physics interfaces and functionality included with the COMSOL Multiphysics license. This book also has instructions about how to use COMSOL Multiphysics and how to access the electronic Documentation and Help content.

Opening Topic-Based Help


The Help window is useful as it is connected to many of the features on the GUI. To learn more about a node in the Model Builder, or a window on the Desktop, click to highlight a node or window, then press F1 to open the Help window, which then


displays information about that feature (or click a node in the Model Builder followed by the **Help** button (). This is called *topic-based* (or *context*) *help*.

Win	<p>To open the Help window:</p> <ul style="list-style-type: none">• In the Model Builder, Application Builder, or Physics Builder click a node or window and then press F1.• On any toolbar (for example, Home, Definitions, or Geometry), hover the mouse over a button (for example, Add Physics or Build All) and then press F1.• From the File menu, click Help ().• In the upper-right corner of the COMSOL Desktop, click the Help () button.
-----	--

Mac	<p>To open the Help window:</p> <ul style="list-style-type: none">• In the Model Builder or Physics Builder click a node or window and then press F1.• On the main toolbar, click the Help () button.• From the main menu, select Help>Help.
Linux	

Opening the Documentation Window









Win	<p>To open the Documentation window:</p> <ul style="list-style-type: none">• Press Ctrl+F1.• From the File menu select Help>Documentation ().
-----	---

Mac	<p>To open the Documentation window:</p> <ul style="list-style-type: none">• Press Ctrl+F1.• On the main toolbar, click the Documentation () button.• From the main menu, select Help>Documentation.
Linux	

THE APPLICATION LIBRARIES WINDOW

Each application includes documentation with the theoretical background and step-by-step instructions to create a model application. The applications are available in COMSOL as MPH-files that you can open for further investigation. You can use the step-by-step instructions and the actual applications as a template for your own modeling and applications. In most models, SI units are used to describe the relevant properties, parameters, and dimensions in most examples, but other unit systems are available.

Once the Application Libraries window is opened, you can search by name or browse under a module folder name. Click to view a summary of the application and its properties, including options to open it or a PDF document.

	The Application Libraries Window in the <i>COMSOL Multiphysics Reference Manual</i> .
<p><i>Opening the Application Libraries Window</i></p> <p>To open the Application Libraries window ():</p>	
	<ul style="list-style-type: none">From the Home toolbar, Windows menu, click () Applications Libraries.From the File menu select Application Libraries. <p>To include the latest versions of model examples, from the File>Help menu, select () Update COMSOL Application Library.</p>
 	<p>Select Application Libraries from the main File> or Windows> menus.</p> <p>To include the latest versions of model examples, from the Help menu select () Update COMSOL Application Library.</p>

CONTACTING COMSOL BY EMAIL

For general product information, contact COMSOL at info@comsol.com.

To receive technical support from COMSOL for the COMSOL products, please contact your local COMSOL representative or send your questions to

support@comsol.com. An automatic notification and case number is sent to you by email.

COMSOL WEBSITES

COMSOL website	www.comsol.com
Contact COMSOL	www.comsol.com/contact
Support Center	www.comsol.com/support
Product Download	www.comsol.com/product-download
Product Updates	www.comsol.com/support/updates
Discussion Forum	www.comsol.com/community
Events	www.comsol.com/events
COMSOL Video Gallery	www.comsol.com/video
Support Knowledge Base	www.comsol.com/support/knowledgebase

Overview of the User's Guide

The *Plasma Module User's Guide* gets you started with modeling using COMSOL Multiphysics. The information in this guide is specific to this module. Instructions how to use COMSOL in general are included with the *COMSOL Multiphysics Reference Manual*.



As detailed in the section [Where Do I Access the Documentation and Application Libraries?](#) this information can also be searched from the COMSOL Multiphysics software **Help** menu.

TABLE OF CONTENTS, GLOSSARY, AND INDEX

To help you navigate through this guide, see the [Contents](#), [Glossary of Terms](#), and [Index](#). See also [Plasma Module Nomenclature](#) in this chapter.

DATA REQUIREMENTS FOR PLASMA MODELING

The [Data Required for Plasma Modeling](#) chapter describes the [Data Requirements](#) to model low-temperature plasmas. It gives an overview of the data you need to assemble before attempting to model a plasma. It also includes the section [Importing Collision Cross-Section Data](#).

AC/DC INTERFACES

[AC/DC Interfaces](#) chapter describes the two physics interfaces available with this module under the AC/DC branch when adding a physics interface. Use the Electrostatics interface to compute the electrostatic field in the plasma caused by separation of space charge between the electrons and ions. This module enhances the physics interface included with the basic COMSOL Multiphysics license. Use the Electrical Circuit interface to add an external electrical circuit to the plasma model. Also see [The Magnetic Fields Interface](#) in the *COMSOL Multiphysics Reference Manual* for information about this physics interface and its feature node settings.

FLUID FLOW INTERFACES

[Fluid Flow Interface](#) describes the Laminar Flow interface, which has a few additional features available for this module.

BOLTZMANN EQUATION, TWO-TERM APPROXIMATION INTERFACE

[The Boltzmann Equation, Two-Term Approximation Interface](#) chapter describes the physics interface, which computes the electron energy distribution function (EEDF) from a set of collision cross sections for some mean discharge conditions. Additionally, electron source coefficients and transport properties can be computed. 1D components only. The underlying theory is included at the end of the chapter.

DRIFT DIFFUSION INTERFACE

[The Drift Diffusion Interface](#) chapter describes the underlying electron transport theory for the Drift Diffusion interface and details the available features. Use the Drift Diffusion interface to compute the electron density and mean electron energy for any type of plasma. A wide range of boundary conditions are available to handle secondary emission, thermionic emission, and wall losses. The underlying theory is included at the end of the chapter.

THE HEAVY SPECIES INTERFACE

[The Heavy Species Transport Interface](#) chapter describes the physics interface, which is a mass balance interface for all non-electron species. This includes charged, neutral, and electronically excited species. The physics interface also allows you to add electron impact reactions, chemical reactions, surface reactions, and species via the Model Builder. The underlying theory is included at the end of the chapter.

THE PLASMA INTERFACES

[Plasma Interfaces](#) chapter describes the following multiphysics interfaces. [The Plasma Interface](#) can be used to study discharges that are sustained by time-varying electrostatic fields or static electric fields in conjunction with secondary electron emission. Rate or Townsend coefficients can be used to define the collisional processes occurring in the plasma. When rate coefficients are used, cross section data can be imported, and the rate coefficient will be automatically computed. [The Inductively Coupled Plasma Interface](#) is for studying discharges that are sustained by induction currents. The induction currents are solved for in the frequency domain and all other variables in the time domain. The electron heating due to the induction currents is handled with two multiphysics features. The 3D components require the AC/DC Module. The underlying theory for each physics interface is included at the end of the chapter. [The Microwave Plasma Interface](#) requires the RF Module, and is used to study discharges that are sustained by electromagnetic waves (wave-heated discharge). Heating of the electrons due to their interaction with the electromagnetic waves is also handled using multiphysics features. The electromagnetic waves are solved for in the frequency domain. The underlying theory is included at the end of the chapter.

THE EQUILIBRIUM DISCHARGE INTERFACES

[Equilibrium Discharges Interfaces](#) chapter describes physics interfaces that can be used to model plasma in or in close local thermodynamic equilibrium (LTE), that is, where the electrons and heavy particles temperature are approximately equal. Close to LTE conditions are usually observed at high pressure and/or under small values of electrical fields.

Data Required for Plasma Modeling

Modeling of low-temperature plasmas is not only difficult because of the large number of physical processes occurring but also because of the amount of data that needs to be supplied to a model. This chapter gives an overview of the data you need to assemble before attempting to model a plasma.

In this chapter:

- [Data Requirements](#)
- [Importing Collision Cross-Section Data](#)

Data Requirements

This section gives an overview of the data to be assembled before attempting to model a plasma.

One of the main difficulties is to find a complete and physically correct chemical mechanism for the plasma of interest. This can involve just a handful of reactions and species (as for argon), or in the case of molecular gases there can be hundreds of reactions and tens of hundreds of species. Often the chemical mechanism can be found from a literature search, but if it is a unique plasma chemistry then the chemical mechanism is likely unknown. The only option is to find a plasma chemistry that exhibits similar properties to your specific application and use that as a reference.

The following sections guide you through the data need for each of the features which make up a plasma model.

- [Electron Impact Reactions](#)
- [Reaction](#)
- [Surface Reaction](#)
- [Species](#)

Electron Impact Reactions

For each *electron impact reaction*, the reaction formula is the first piece of information that is required. Most published papers involving the numerical modeling of plasmas include a table of the reactions which make up the plasma chemistry.

Once the formula is entered you need to supply some data to indicate the dependency of the reaction rate on the electron energy. The most common way of accomplishing this is to specify *cross-section data* for each of the electron impact reactions. The cross-section data can be hard to find or not even exist. Cross-section data is available for most of the common gases. Some useful resources for cross-section data are available at the following links:

- <http://fr.lxcat.net/home/>

- http://jila.colorado.edu/~avp/collision_data/electronneutral/ELECTRON.TXT



Cross-section data must be in a specific format in order to be imported into COMSOL Multiphysics. See [Importing Collision Cross-Section Data](#).

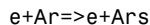
If cross-section data is not available then the rate coefficient can be specified with a constant value, an Arrhenius expression, or by a lookup table. In the case of an Arrhenius expression, the Arrhenius coefficients must be specified. If you want to use lookup tables for the rate coefficient then you need to load in a table of rate or Townsend coefficient versus mean electron energy.

Depending on the type of electron impact collision, the following data is also required:

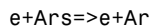
- The mass ratio of the electron to the target species for elastic collisions.
- The energy loss (in eV) for inelastic collisions.
- The ratio of statistical weights between the target and produced species if a detailed balance is required.

DETAILED BALANCE

If a detailed balance is requested then cross-section data for the reverse reaction is created. So, say you have an electronic excitation reaction like:



and you also want to include the reverse reaction in the model:



then the cross-section data for the forward reaction can be reused to compute the cross-section data for the reverse reaction. In order to construct a detailed balance, the ratio of the statistical weights between the target and product species is required. The formula used to compute the cross-section data for the reverse reaction by first shifting the electron energy data, ϵ , by the excitation energy, $\Delta\epsilon$, $\epsilon \rightarrow \epsilon + \Delta\epsilon$, then scaling the cross-section data, σ , by the ratio of statistical weights, g , $\sigma \rightarrow \sigma/g$. This forms a new look-up table of collision cross section versus energy for the reverse reaction. In practice, the value used should not make much difference to the overall discharge characteristics. Data for some gases can be found in [Ref. 2](#).

Reaction

For gas phase reactions, the forward rate constant using either Arrhenius coefficients or numeric value data is required. If the reaction is reversible then enter the reverse rate constant in the same way.

Surface Reaction

For surface reactions the following data is required:

- The forward sticking or rate coefficient, which can either be a numeric value or specified in terms of Arrhenius parameters as a function of the surface temperature.
- For first-order reactions, the total surface site concentration is required.
- For surface reactions that result in emission of secondary electrons, the secondary emission coefficient and mean energy of the secondary electrons is required.

Species

For each of the heavy species the following information is required:

- The molecular weight of the species
- The potential characteristic length of the species; required to compute the correct diffusivity and mobility
- The potential energy minimum of the species; required to compute the correct diffusivity and mobility

In cases where you are interested in solving for the neutral gas temperature, the following additional information is required:

- The species molar enthalpy, entropy, and specific heat, which can be entered directly as a function of temperature or by using *NASA polynomials*
- In the case of electronically excited species or ions, the properties of the ground state species can be entered, and an additional contribution to the species enthalpy can also be entered

The Plasma Module provides predefined transport and thermodynamic data for the following species:

TABLE 2-1: TABLE OF GASES WITH PREDEFINED TRANSPORT AND THERMODYNAMIC DATA

GAS	FORMULA
Helium	He
Neon	Ne
Argon	Ar
Krypton	Kr
Xenon	Xe
Carbon	C
Carbon hydride	CH
Methylene	CH ₂
Methyl radical	CH ₃
Methane	CH ₄
Carbon monoxide	CO
Carbon dioxide	CO ₂
Atomic fluorine	F
Fluorine	F ₂
Sulfur hexafluoride	SF ₆
Silicon tetrafluoride	SiF ₄
Atomic hydrogen	H
Hydrogen	H ₂
Water	H ₂ O
Hydrogen peroxide	H ₂ O ₂
Nitroxyl	HNO
Hydroperoxyl	HO ₂
Hydroxyl	OH
Atomic nitrogen	N
Nitrogen	N ₂
Nitrous oxide	N ₂ O
Amino radical	NH
Amino radical	NH ₂
Ammonia	NH ₃

TABLE 2-1: TABLE OF GASES WITH PREDEFINED TRANSPORT AND THERMODYNAMIC DATA

GAS	FORMULA
Nitric oxide	NO
Nitrogen dioxide	NO2
Atomic oxygen	O
Oxygen	O3
Ozone	O3
Silane	SiH4
Silane radical	SiH2
Disilane	Si2H6

The following surface species properties are also predefined:

TABLE 2-2: TABLE OF SURFACE SPECIES WITH PREDEFINED PHYSICAL AND THERMODYNAMIC DATA

NAME	FORMULA
Silicon	Si
Silicon hydride	SiH
Silicon dioxide	SiO2

Importing Collision Cross-Section Data

To facilitate the modeling process, all multiphysics interfaces and the Boltzmann Equation, Two-Term Approximation interface can import sets of collision cross sections from file. This section details the file format required and has these topics:

- [Cross-Section Data File Format](#)
- [The File Format](#)
- [References for the Plasma Module Cross-Section Data Requirements](#)

Cross-Section Data File Format

IMPORTING CROSS SECTIONS

The file format used to import cross-section data is almost the same as the format proposed by Hagelaar in [Ref. 1](#). The best way to get the cross-section data in the correct format is to modify the file `Ar_xssecs.txt` which is available in the Plasma Module Applications Libraries folder.

For each electron impact reaction, the *lookup table* of electron energy (eV) versus collision cross section (m^2) allows electron transport properties and source coefficients to be computed. As an example, a set of collision cross sections for oxygen are shown

in Figure 2-1.

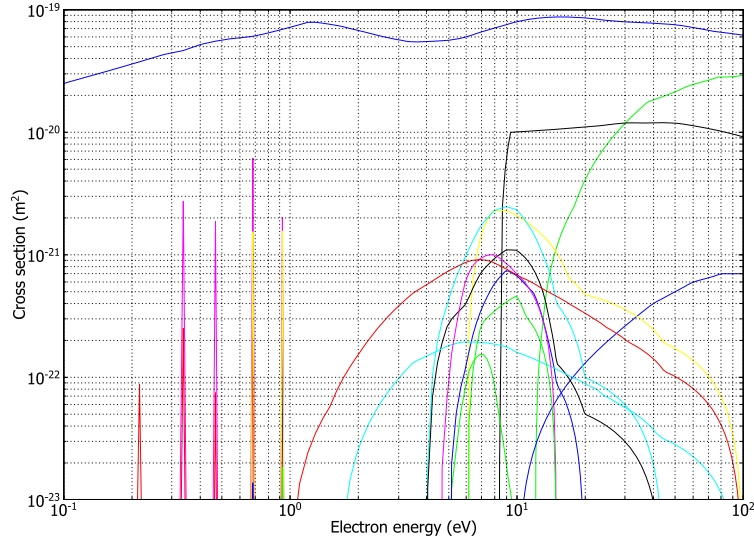


Figure 2-1: Plot of a set of collision cross sections for molecular oxygen. There are 16 collisions in total.

The File Format

Cross-section data must be specified using the following format:

```
1st line
2nd line
3rd line
4th line
```

```
-----
energy data (eV) Collision cross section (m2)
-----
```

- 1 1st line: Keyword in capitals indicating the type of the collision. Possible collision types are ELASTIC, EXCITATION, IONIZATION, and ATTACHMENT. Note the use of capital letters.
- 2 2nd line: Reaction formula. This formula carries over to the COMSOL Multiphysics GUI so that whatever is entered here appears in the **Formula** field. The formula must be separated by a => sign.
- 3 3rd line: For elastic collisions, the ratio of the electron mass to the target particle mass. For attachment, the 3rd line is 0 (zero). For ionization, the threshold energy

in eV. For excitation there are three (3) numbers separated by a space. The first number is the excitation energy (eV). The second number is the ratio of statistical weights of the final state to the initial state. The third number is a flag that determines whether or not a detailed balance is used. Use a 1 to construct a detailed balance and 0 otherwise.

- 4 4th line: Two unit conversion factors by which the columns of data are multiplied.
- 5 Table of the cross section as a function of energy. The table starts and ends by a line of dashes '-----' (at least five), and has otherwise two numbers per line: the energy in eV and the cross section in m^2 . The table does not need to follow the other lines immediately; that is, user comments such as reference information can be included above the table.

Here are some examples for the types of collisions available:

```
ELASTIC
e+02=>e+02
1.71e-5
1.0 1.0
-----
0.000000e+000 3.500000e-021
.
.
.
-----
```

```
ATTACHMENT
e+02=>0+0-
0
1.0 1.0
-----
5.800000e-002 2.220446e-036
.
.
.
-----
```

```
EXCITATION
e+02=>e+02b1s
1.6270 5 1
1.0 1.0
-----
1.627000e+000 2.220446e-036
.
.
.
-----
```

```


IONIZATION
e+02=>2e+02+
12.0600
1.0 1.0
-----
1.206000e+001 2.220446e-036
.
.
.
-----

```

References for the Plasma Module Cross-Section Data Requirements

1. G.J.M. Hagelaar and L.C. Pitchford, “Solving the Boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models,” *Plasma Sources Science and Technology*, vol. 14, pp. 722–733, 2005.
2. W.L. Wiese, M. W. Smith, and B.M. Glennon, “Atomic Transition Probabilities,” www.nist.gov/data/nsrds/NSRDS-NBS4.pdf, 1966.



AC/DC Interfaces

This chapter describes the physics interfaces found under the **AC/DC** branch ().

In this chapter:

- [The Electrostatics Interface](#)
- [The Electric Currents Interface](#)
- [The Electrical Circuit Interface](#)
- [Theory for the Electrostatics Interface](#)
- [Theory for the Electric Currents Interface](#)
- [Theory for the Electrical Circuit Interface](#)
- [Connecting to Electrical Circuits](#)
- [Spice Import](#)

The Electrostatics Interface

The **Electrostatics (es)** interface (), found under the **AC/DC** branch () when adding a physics interface, is used to compute the electric field, electric displacement field, and potential distributions in dielectrics under conditions where the electric charge distribution is explicitly prescribed. The formulation is stationary except for use together with other physics interfaces. Eigenfrequency, frequency-domain, small-signal analysis, and time-domain modeling are supported in all space dimensions.

The physics interface solves Gauss' Law for the electric field using the scalar electric potential as the dependent variable.

Charge Conservation is the main node, which adds the equation for the electric potential and has a Settings window for defining the constitutive relation for the electric displacement field and its associated properties such as the relative permittivity.

When this physics interface is added, these default nodes are also added to the **Model Builder**—**Charge Conservation**, **Zero Charge** (the default boundary condition), and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and space charges. You can also right-click **Electrostatics** to select physics features from the context menu.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (`_`) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is **es**.

CROSS-SECTION AREA (1D COMPONENTS)

For 1D components, enter a default value for the **Cross-section area** A (SI unit: m^2). The default value of 1 is typically not representative for a thin domain. Instead it describes a unit thickness that makes the 1D equation identical to the equation used for 3D components. See also [Change Cross Section](#).

THICKNESS (2D COMPONENTS)

For 2D components, enter a default value for the **Out-of-plane thickness** d (SI unit: m). The default value of 1 is typically not representative for a thin dielectric medium, for example. Instead it describes a unit thickness that makes the 2D equation identical to the equation used for 3D components. See also [Change Thickness \(Out-of-Plane\)](#).

PHYSICS-CONTROLLED MESH

Select the **Enable** check box to allow the physics interface to control the meshing process. Information from the physics, such as the presence of an infinite elements domain or periodic condition, will be used to set up automatically an appropriate mesh sequence.

TERMINAL SWEEP SETTINGS

Enter a **Reference impedance** Z_{ref} (SI unit: Ω). The default is 50 Ω .

Select the **Activate terminal sweep** check box to switch on the sweep and invoke a parametric sweep over the terminals. Enter a **Sweep parameter name** to assign a specific name to the variable that controls the terminal number solved for during the sweep. The **Sweep parameter name** must also be declared as a model parameter. The default is PortName.

The generated lumped parameters are in the form of capacitance matrix elements. The terminal settings must consistently be of either fixed voltage or fixed charge type.

The lumped parameters are subject to **Touchstone file export**. Enter a file path or **Browse** for a file. Select a **Parameter format (value pairs)** for the Touchstone export—**Magnitude and angle (MA)** (the default), **Magnitude in dB and angle (DB)**, or **Real and imaginary parts (RI)**. Select an option from the **If file exists** list—**Overwrite** or **Create new**. Create new is useful when the model is solved multiple times with different settings. Select a **Parameter to export**—**Z** (the default), **Y**, or **S**.

From the **Home** toolbar add a **Parameters** node and enter the chosen name and assign to it a temporary **Expression** of unity into the **Parameters** table. You can also right-click **Global Definitions** to add the node.







Only a temporary expression needs to be entered at this stage. During the analysis process, the solver assigns a proper value to this parameter. The generated lumped parameters are in the form of capacitance matrix elements. The terminal settings must consistently be of either fixed voltage or fixed charge type.

DEPENDENT VARIABLES

The dependent variable is the **Electric potential** V . You can change its name, which changes both the field name and the variable name. If the new name coincides with the name of another electric potential field in the model, the physics interfaces shares degrees of freedom. The new name must not coincide with the name of a field of another type or with a component name belonging to some other field.

DISCRETIZATION

To display this section, click the **Show** button () and select **Discretization**.

	In the <i>COMSOL Multiphysics Reference Manual</i> , see Table 2-3 for links to common sections and Table 2-4 to common feature nodes. You can also search for information: press F1 to open the Help window or Ctrl+F1 to open the Documentation window.
	<ul style="list-style-type: none">• Domain, Boundary, Edge, Point, and Pair Nodes for the Electrostatics Interface• Theory for the Electrostatics Interface
	<i>Electric Sensor</i> : Application Library path COMSOL_Multiphysics/Electromagnetics/electric_sensor

Domain, Boundary, Edge, Point, and Pair Nodes for the Electrostatics Interface

The [Electrostatics Interface](#) has these domain, boundary, edge, point, and pair nodes available.

ABOUT THE BOUNDARY CONDITIONS

The relevant physics interface condition at interfaces between different media is

$$\mathbf{n}_2 \cdot (\mathbf{D}_1 - \mathbf{D}_2) = \rho_s$$

In the absence of surface charges, this condition is fulfilled by the natural boundary condition

$$\mathbf{n} \cdot [(\epsilon_0 \nabla V - \mathbf{P})_1 - (\epsilon_0 \nabla V - \mathbf{P})_2] = -\mathbf{n} \cdot (\mathbf{D}_1 - \mathbf{D}_2) = 0$$

AVAILABLE NODES

These nodes, listed in alphabetical order, are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users). Also see [Table 3-1](#) for a list of interior and exterior boundary conditions, including edge, point, and pair availability.



In general, to add a node, go to the **Physics** toolbar no matter what operating system you are using. Subnodes are available by clicking the parent node and selecting it from the **Attributes** menu.

- [Change Cross Section](#)
- [Change Thickness \(Out-of-Plane\)](#)
- [Charge Conservation](#)
- [Charge Conservation, Piezoelectric](#)¹
- [Dielectric Contact](#)
- [Dielectric Shielding](#)
- [Distributed Capacitance](#)
- [Electric Displacement Field](#)
- [Electric Potential](#)
- [Electrostatic Point Dipole](#)
- [External Surface Charge Accumulation](#)
- [Floating Potential](#)
- [Ground](#)
- [Initial Values](#)
- [Line Charge](#)
- [Line Charge \(on Axis\)](#)
- [Line Charge \(Out-of-Plane\)](#)
- [Metal Contact](#)
- [Periodic Condition](#)
- [Point Charge](#)
- [Point Charge \(on Axis\)](#)
- [Space Charge Density](#)
- [Surface Charge Accumulation](#)
- [Surface Charge Density](#)
- [Terminal](#)
- [Thin Low Permittivity Gap](#)
- [Zero Charge](#) (the default boundary condition)

¹ This feature is available with the Piezoelectric Devices interface, which requires the Acoustics Module, MEMS Module, or the Structural Mechanics Module.

Table 3-1 lists the interior and exterior boundaries available with this physics interface. It also includes edge, point, and pair availability.

TABLE 3-1: INTERIOR AND EXTERIOR BOUNDARY CONDITIONS (INCLUDING EDGE, POINT, AND PAIR AVAILABILITY) FOR THE ELECTROSTATICS INTERFACE

NODE	INTERIOR	EXTERIOR	ALSO AVAILABLE FOR
Change Cross Section	x	x	pairs
Change Thickness (Out-of-Plane)	x	x	pairs
Dielectric Shielding	x	x	pairs
Distributed Capacitance	x	x	pairs
Electric Displacement Field	x	x	pairs
Electric Potential	x	x	edges, points, and pairs
External Surface Charge Accumulation		x	pairs
Floating Potential	x	x	pairs
Ground	x	x	edges, points, and pairs
Periodic Condition		x	not applicable
Surface Charge Accumulation	x		not applicable
Surface Charge Density	x	x	pairs
Terminal	x	x	domains
Thin Low Permittivity Gap	x		not applicable
Zero Charge (the default)	x	x	pairs



For axisymmetric components, COMSOL Multiphysics takes the axial symmetry boundaries (at $r = 0$) into account and automatically adds an **Axial Symmetry** node to the model that is valid on the axial symmetry boundaries only. There are also **Line Charge (on Axis)** and **Point Charge (on Axis)** available.



In the *COMSOL Multiphysics Reference Manual*, see [Table 2-3](#) for links to common sections and [Table 2-4](#) to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

Charge Conservation

The **Charge Conservation** node adds the equations for charge conservation according to Gauss' law for the electric displacement field. It provides an interface for defining the constitutive relation and its associated properties such as the relative permittivity.

MATERIAL TYPE

The **Material type** setting decides how materials behave and how material properties are interpreted when the mesh is deformed. Select **Solid** for materials whose properties change as functions of material strain, material orientation, and other variables evaluated in a material reference configuration (material frame). Select **Non-solid** for materials whose properties are defined only as functions of the current local state at each point in the spatial frame, and for which no unique material reference configuration can be defined. Select **From material** to pick up the corresponding setting from the domain material on each domain.

ELECTRIC FIELD

Select a **Constitutive relation** to describe the macroscopic properties of the medium (relating the electric displacement \mathbf{D} with the electric field \mathbf{E}) and the applicable material properties, such as the relative permittivity. Select:

- **Relative permittivity** (the default) to use the constitutive relation $\mathbf{D} = \epsilon_0 \epsilon_r \mathbf{E}$. Then the default is to take the **Relative permittivity** ϵ_r (dimensionless) values **From material**. For **User defined**, select **Isotropic**, **Diagonal**, **Symmetric**, or **Anisotropic** and enter values or expressions in the field or matrix. The default is 1.
- **Polarization** to use the constitutive relation $\mathbf{D} = \epsilon_0 \mathbf{E} + \mathbf{P}$. Then enter the components based on space dimension for the **Polarization** vector \mathbf{P} (SI unit: C/m²). The defaults are 0 C/m².
- **Remanent electric displacement** to use constitutive relation $\mathbf{D} = \epsilon_0 \epsilon_r \mathbf{E} + \mathbf{D}_r$, where \mathbf{D}_r is the remanent displacement (the displacement when no electric field is present). Then the default is to take the **Relative permittivity** ϵ_r (dimensionless) values **From material**. For **User defined**, select **Isotropic**, **Diagonal**, **Symmetric**, or **Anisotropic** and enter values or expressions in the field or matrix. Then enter the components based on space dimension for the **Remanent electric displacement** \mathbf{D}_r (SI unit: C/m²). The defaults are 0 C/m².

Initial Values

The **Initial Values** node adds an initial value for the electric potential V that can serve as an initial condition for a transient simulation or as an initial guess for a nonlinear solver.

INITIAL VALUES

Enter a value or expression for the initial value of the **Electric potential** V (SI unit: V). The default value is 0 V.

Space Charge Density

The **Space Charge Density** node adds a space charge density ρ , which appears on the right-hand side of the equation that the physics interface defines.

SPACE CHARGE DENSITY

Enter a value or expression for the **Space charge density** ρ_v (SI unit: C/m³). The default is 0 C/m³.

Zero Charge

The **Zero Charge** node adds the condition that there is zero charge on the boundary so that $\mathbf{n} \cdot \mathbf{D} = 0$. This boundary condition is also applicable at symmetry boundaries where the potential is known to be symmetric with respect to the boundary. This is the default boundary condition at exterior boundaries. At interior boundaries, it means that no displacement field can penetrate the boundary and that the electric potential is discontinuous across the boundary.

Ground

The **Ground** node implements ground (zero potential) as the boundary condition $V = 0$.

Ground means that there is a zero potential on the boundary. This boundary condition is also applicable at symmetry boundaries where the potential is known to be antisymmetric with respect to the boundary.

For some physics interfaces, also select additional **Ground** nodes from the **Edges** (3D components) or **Points** (2D and 3D components) submenus. For 2D axisymmetric components, it can be applied on the Symmetry axis.

BOUNDARY, EDGE, OR POINT SELECTION



Beware that constraining the potential on edges or points in 3D or on points in 2D usually yields a current outflow that is mesh dependent.

CONSTRAINT SETTINGS

To display this section, click the **Show** button () and select **Advanced Physics Options**.

Electric Potential

The **Electric Potential** node provides an electric potential V_0 as the boundary condition $V = V_0$.

Because the electric potential is being solved for in the physics interface, the value of the potential is typically defined at some part of the geometry. For some physics interfaces, also select additional **Electric Potential** nodes from the **Edges** (3D components) or **Points** (2D and 3D components) submenus. For 2D axisymmetric components, it can be applied on the symmetry axis.

BOUNDARY, EDGE, OR POINT SELECTION



Beware that constraining the potential on edges or points in 3D or on points in 2D usually yields a current outflow that is mesh dependent.

ELECTRIC POTENTIAL

Enter the value or expression for the **Electric potential** V_0 (SI unit: V). The default is 0 V.

CIRCUIT SETTINGS



Capacitively Coupled Plasmas

Select the **Circuit type**—**None** (the default), **Series RC circuit**, **Ballast resistor**, or **Blocking capacitor**. If **None** is selected, no external circuit is added.

Series RC Circuit

If **Series RC circuit** is selected, enter a value or expression for the **Ballast resistor** R_b (SI unit: Ω) and **Blocking capacitance** C_b (SI unit: F). The defaults are 1000 Ω and 1 pF, respectively. This option is typically used for metallic electrodes in DC discharges. The presence of the external circuit prevents the discharge from arcing. If arcing begins to occur the solver tends to fail and the ballast resistor value should be increased. The presence of the blocking capacitor also helps when solving by ramping up the applied potential from zero over a finite period of time.

Ballast Resistor

If **Ballast resistor** is selected, enter a value or expression for the **Ballast resistor** R_b (SI unit: Ω). The default is 1000 Ω . The presence of the ballast resistor reduces the likelihood that the discharge arcs. However, it is recommended that the **Series RC circuit** be chosen in preference to this one.

Blocking Capacitor

If **Blocking capacitor** is selected, enter a value or expression for the **Blocking capacitance** C_b (SI unit: F). The default is 1 pF. For 1D components, enter a value for the **Electrode surface area** A_s (SI unit: m^2). The default is 0.01 m^2 . This value is used to compute the total amount of electrical current flowing into the electrode due to the time varying displacement field and flow of charged particles. This option is typically used when modeling capacitively coupled plasmas, since these typically operate with a blocking capacitor between the power supply and the electrodes.

CONSTRAINT SETTINGS

To display this section, click the **Show** button () and select **Advanced Physics Options**.

Surface Charge Density

The **Surface Charge Density** node provides the following surface-charge boundary condition for exterior boundaries (left) and interior boundaries (right):

$$-\mathbf{n} \cdot \mathbf{D} = \rho_s, \quad \mathbf{n} \cdot (\mathbf{D}_1 - \mathbf{D}_2) = \rho_s$$

Specify the surface charge density ρ_s at an outer boundary or at an interior boundary between two nonconducting media.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or click **Harmonic Perturbation** on the **Physics** toolbar. For more information, see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

SURFACE CHARGE DENSITY

Enter the value or expression for the **Surface charge density** ρ_s (SI unit: C/m²).

External Surface Charge Accumulation

The **External Surface Charge Accumulation** node implements the boundary condition

$$-\mathbf{n} \cdot \mathbf{D} = \rho_s$$

where ρ_s is the solution of the following distributed ODE on the boundary:

$$\frac{d\rho_s}{dt} = \mathbf{n} \cdot \mathbf{J}_i + \mathbf{n} \cdot \mathbf{J}_e$$

where $\mathbf{n} \cdot \mathbf{J}_i$ is the normal component of the total ion current density on the wall and $\mathbf{n} \cdot \mathbf{J}_e$ is the normal component of the total electron current density on the wall, which are feature inputs.

MATERIAL TYPE

The **Material type** setting decides how materials behave and how material properties are interpreted when the mesh is deformed. Select **Solid** for materials whose properties change as functions of material strain, material orientation, and other variables evaluated in a material reference configuration (material frame). Select **Non-solid** for materials whose properties are defined only as functions of the current local state at each point in the spatial frame, and for which no unique material reference configuration can be defined. Select **From material** to pick up the corresponding setting from the domain material on each domain.

EXTERNAL SURFACE CHARGE ACCUMULATION

Enter values or expressions for the **Normal ion current density** $\mathbf{n} \cdot \mathbf{J}_i$ (SI unit: A/m²) and the **Normal electron current density** $\mathbf{n} \cdot \mathbf{J}_e$ (SI unit: A/m²).

Electric Displacement Field

The **Electric Displacement Field** node adds the following electric-displacement boundary condition:

$$\mathbf{n} \cdot \mathbf{D} = \mathbf{n} \cdot \mathbf{D}_0$$

It specifies the normal component of the electric displacement field at a boundary.

ELECTRIC DISPLACEMENT FIELD

Enter the coordinates of the **Boundary electric displacement field \mathbf{D}_0** (SI unit: C/m²).

Periodic Condition

The **Periodic Condition** node defines periodicity or antiperiodicity between two boundaries. If required, activate periodic conditions on more than two boundaries, in which case the Periodic Condition tries to identify two separate surfaces that can each consist of several connected boundaries. For more complex geometries, it might be necessary to use the **Destination Selection** subnode. With this subnode the boundaries which constitute the source and destination surfaces can be manually specified. The **Destination Selection** subnode is available from the context menu (right-click the parent node) as well as from the **Physics** toolbar, **Attributes** menu.



When this feature is used in conjunction with a [Sector Symmetry](#) feature on connected boundaries, the same periodic condition feature cannot be used on both sides of where the sector symmetry boundaries connect with the periodic boundaries. At least two periodic condition features are required for the model to compute correctly.

BOUNDARY SELECTION

When using nonconforming meshes on the source and destination of a periodic boundary pair, for numerical stability, a finer mesh should be applied on the destination side. Use conforming meshes if possible.

PERIODIC CONDITION

Select a **Type of periodicity**—**Continuity** (the default) or **Antiperiodicity**.

CONSTRAINT SETTINGS

To display this section, click the **Show** button () and select **Advanced Physics Options**.

For information about the Orientation of Source section, see [Orientation of Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

Thin Low Permittivity Gap

Use the **Thin Low Permittivity Gap** node

$$\mathbf{n} \cdot \mathbf{D}_1 = \frac{\epsilon_0 \epsilon_r}{d_s} (V_1 - V_2)$$

$$\mathbf{n} \cdot \mathbf{D}_2 = \frac{\epsilon_0 \epsilon_r}{d_s} (V_2 - V_1)$$

to model a thin gap of a material with a small permittivity compared to the adjacent domains. The layer has the thickness d_s and the relative permittivity ϵ_r . The indices 1 and 2 refer to the two sides of the boundary.

THIN LOW PERMITTIVITY GAP

The default is to take the **Relative permittivity** ϵ_r (dimensionless) values **From material**. For **User defined**, enter a different value or expression. Enter a **Surface thickness** d_s (SI unit: m). The default is 5 mm.

Surface Charge Accumulation

On interior boundaries, the **Surface Charge Accumulation** node implements the boundary condition

$$\mathbf{n} \cdot (\mathbf{D}_1 - \mathbf{D}_2) = \rho_s$$

where ρ_s is the solution of the following distributed ODE on the boundary:

$$\frac{d\rho_s}{dt} = \mathbf{n} \cdot \mathbf{J}_i + \mathbf{n} \cdot \mathbf{J}_e$$

where $\mathbf{n} \cdot \mathbf{J}_i$ is the normal component of the total ion current density on the wall and $\mathbf{n} \cdot \mathbf{J}_e$ is the normal component of the total electron current density on the wall.

SURFACE CHARGE ACCUMULATION

Enter the values or expressions for the **Normal ion current density** $\mathbf{n} \cdot \mathbf{J}_i$ (SI unit: A/m²) and the **Normal electron current density** $\mathbf{n} \cdot \mathbf{J}_e$ (SI unit: A/m²).



Dielectric Barrier Discharge: Application Library path **Plasma_Module/**
Direct_Current_Discharges/argon_dbd_1d

Dielectric Shielding

The **Dielectric Shielding** node adds dielectric shielding as a boundary condition. It describes a thin layer with thickness d_s and a bulk relative permittivity; ϵ_r that shields the electric field:

$$\mathbf{n} \cdot \mathbf{D} = -\nabla_t \cdot \epsilon_0 \epsilon_{rs} d_s \nabla_t V$$

Use this boundary condition when approximating a thin domain with a boundary to reduce the number of mesh elements.

MATERIAL TYPE

The **Material type** setting decides how materials behave and how material properties are interpreted when the mesh is deformed. Select **Solid** for materials whose properties change as functions of material strain, material orientation and other variables evaluated in a material reference configuration (material frame). Select **Non-solid** for materials whose properties are defined only as functions of the current local state at each point in the spatial frame, and for which no unique material reference configuration can be defined. Select **From material** to pick up the corresponding setting from the domain material on each domain.

ELECTRIC SHIELDING

The default is to take the **Relative permittivity** ϵ_r (dimensionless) values **From material**. It takes it from the adjacent domains if not explicitly defined. For **User defined** select **Isotropic**, **Diagonal**, **Symmetric**, or **Anisotropic** and enter values or expressions in the field or matrix. Enter a **Surface thickness** d_s (SI unit: m) of the shielding. The default is 1 m.

Terminal

The **Terminal** node provides a boundary or domain condition for connection to external circuits, to transmission lines, or with a specified voltage or charge. By specifying zero charge, a floating potential condition is obtained. When the **Terminal** node is applied on the domain level, the electric potential is set to constant on the entire domain selection.

The **Terminal** node also enables the computation of the lumped parameters of the system, such as capacitance.

TERMINAL

Specify the terminal properties. Enter a unique **Terminal name** that will be used to identify the global variables (such as the terminal voltage or current) that are defined by the terminal. The **Terminal name** can contain alphanumeric characters, but it should be numeric when performing terminal sweeps.

Select a **Terminal type**—**Charge** (the default), **Voltage**, **Circuit**, or **Terminated**. Select:

- **Charge** to prescribe the total charge deposited on the terminal boundaries. Enter the total charge Q_0 . The default is 0 C, which corresponds to a uncharged floating electrode.
- **Voltage** to specify a fixed electric potential at the terminal. Enter an electric potential V_0 . The default is 1 V.
- **Circuit** to specify a terminal connected to an Electrical Circuit physics interface that models an external circuit. The Terminal node provides a current-voltage characteristic to the circuit.



The [Electrical Circuit Interface](#) requires a current from the Terminal, so the **Circuit** terminal type can only be used in Time Dependent or Frequency Domain studies, in which the current can be computed as the time derivative of the charge.

The **Circuit** type cannot be used together with a terminal sweep.

- **Terminated** to connect the terminal to an impedance that might represent a load or a transmission line. When **Terminated** is selected, the scattering parameters (S-parameters) are computed. The **Terminated** setting can only be used in frequency-domain studies.
 - Enter a **Terminal power** P_0 to specify the input power at this terminal from the transmission line. This excitation can be toggled using a port sweep.
 - Select an option from the **Characteristic impedance from** list to define the value of the impedance—**Physics interface** or **User defined**. For **Physics interface** the **Reference impedance** Z_{ref} defined on the **Settings** window for the physics under Sweep Settings is used. For **User defined** enter a specific impedance Z_{ref} for this terminal. The default is 50 Ω .

Initial values

The initial values subsection is used to provide initial values for the internal state variables in the Terminal. These settings can be useful to provide an initial guess to the nonlinear solver or the initial value in a time-dependent study.

The **Initial value for voltage** is available for all Terminal types except Voltage. Enter an **Initial value for voltage** V_{init} .

The **Initial value for charge** is available for the Circuit excitation. It specifies the initial charge on the terminal when the circuit is connected. Enter an **Initial value for charge** Q_{init} .

CONSTRAINT SETTINGS

To display this section, click the **Show** button () and select **Advanced Physics Options**.



[Studies and Solvers](#) in the *COMSOL Multiphysics Reference Manual*

Floating Potential

The **Floating Potential** node is used when modeling a metallic electrode at floating potential. The electrode might have a charge Q_0 deposited on it, or can be connected to an electrical circuit.

FLOATING POTENTIAL

The **Floating potential identifier** text area shows the unique identifier for the floating potential feature. It is used to identify the global variables created by the node, such as the voltage. The **Floating potential group** check box on the **Settings** window for **Floating Potential** controls how potentials are assigned to boundary segments in the feature's boundary selection. If this check box is not selected, a single potential is applied to all boundaries in the selection. If the check box is selected, each group of contiguous boundaries in the selection is given a unique potential. This simplifies the setup of models with many floating electrodes. The values of the potential at each group of boundaries are then made available in postprocessing, collected in a vector variable.

The following options are not available if the **Floating potential group** check box is selected. Select a specification for the **Electric charge**—choose **User defined** to specify a

total **Charge** Q_0 deposited on the surface. Select **Circuit** to connect the floating potential to an Electrical Circuit.



The **Electrical Circuit Interface** requires a current from the Floating Potential, so the **Circuit** terminal type can only be used in Time Dependent or Frequency Domain studies, in which the current can be computed as the time derivative of the charge.

Initial Values

The initial values subsection is used to provide initial values for the internal state variables in the Floating potential. These settings can be useful to provide an initial guess to the nonlinear solver or the initial value in a time-dependent study.

Enter an **Initial value for voltage** V_{init} .

The **Initial value for charge** is available for the Circuit excitation. It specifies the initial charge on the floating potential when the circuit is connected. Enter an **Initial value for charge** Q_{init} .

CONSTRAINT SETTINGS

To display this section, click the **Show** button () and select **Advanced Physics Options**.



Settings and Properties Windows for Features Nodes in the *COMSOL Multiphysics Reference Manual*

Distributed Capacitance

The **Distributed Capacitance** node adds a distributed capacitance boundary condition according to the following equations for exterior boundaries (left) and interior boundaries (right):

$$-\mathbf{n} \cdot \mathbf{D} = \epsilon_0 \epsilon_r \frac{V_{\text{ref}} - V}{d_s} \quad \mathbf{n} \cdot (\mathbf{D}_1 - \mathbf{D}_2) = \epsilon_0 \epsilon_r \frac{V_{\text{ref}} - V}{d_s}$$

Use this boundary condition to model a thin sheet or film of a dielectric material. The sheet has the relative permittivity ϵ_r and the surface thickness d_s , and it is connected to the reference potential V_{ref} .

DISTRIBUTED CAPACITANCE

Enter the values or expressions for **Relative permittivity** ϵ_r (dimensionless), **Surface thickness** d_s (SI unit: m), and **Reference potential** V_{ref} . The default value for the surface thickness is 10^{-3} m (1 mm) and 0 V for the reference potential.

Line Charge

For 3D components, use the **Line Charge** node to specify line charges along the edges of a geometry. Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

EDGE SELECTION



Beware that constraining the potential on edges usually yields a current outflow that is mesh dependent.

LINE CHARGE

Enter a value or expression to apply a **Line charge** Q_L (SI unit: C/m). This source represents electric charge per unit length and the default is 0 C/m.



[Line Charge \(on Axis\)](#) and [Line Charge \(Out-of-Plane\)](#)

Line Charge (on Axis)

For 2D axisymmetric components, use the **Line Charge (on Axis)** node to specify line charges along the symmetry axis.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or click **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

LINE CHARGE (ON AXIS)

Enter a value or expression to apply a **Line charge** Q_L (SI unit: C/m). This source represents electric charge per unit length and the default is 0 C/m.



Line Charge and Line Charge (Out-of-Plane)

Line Charge (Out-of-Plane)

For 2D and 2D axisymmetric components, points are selected and this is the same as a line out-of-plane.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

Use the **Line Charge (Out-of-Plane)** node to specify line charges along the points of a geometry for 2D and 2D axisymmetric components.

POINT SELECTION



Beware that constraining the potential on points usually yields a current outflow that is mesh dependent.

LINE CHARGE (OUT-OF-PLANE)

Enter a value or expression to apply a **Line charge** Q_L (SI unit: C/m). This source represents electric charge per unit length and the default is 0 C/m.



Line Charge and Line Charge (on Axis)

Point Charge

The **Point Charge** node adds a point source to 3D components. The point charge represents an electric displacement field flowing out of the point.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

POINT SELECTION



Beware that constraining the potential on points usually yields a current outflow that is mesh dependent.

POINT CHARGE

Enter a value or expression to apply a **Point charge** Q_P (SI unit: C) to points. This source represents an electric displacement field flowing out of the point. The default is 0 C.



[Point Charge \(on Axis\)](#) and [Line Charge \(Out-of-Plane\)](#)

Point Charge (on Axis)

The **Point Charge (on Axis)** node adds a point source to 2D axisymmetric components. The point charge represents an electric displacement field flowing out of the point.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

POINT SELECTION



Beware that constraining the potential on points usually yields a current outflow that is mesh dependent.

POINT CHARGE (ON AXIS)

Enter a value or expression to apply a **Point charge** Q_P (SI unit: C) to points on an axis. This source represents an electric displacement field flowing out of the point. The default is 0 C.



Point Charge and Line Charge (Out-of-Plane)

Change Cross Section

This node is available with 1D components. This setting overrides the global **Cross-Section Area** setting made in any physics interface that uses this feature. For 2D components, see [Change Thickness \(Out-of-Plane\)](#).

Use the **Change Cross-Section** node to set the cross-section area for specific geometric entities.

CHANGE CROSS SECTION

Enter a value or expression for the **Cross-section area** A . The default value of 1 unit length is typically not representative for a thin domain. Instead it describes a unit thickness that makes the 1D equation identical to the equation used for 3D components.

Change Thickness (Out-of-Plane)

This node is available for 2D components. This setting overrides the global **Thickness** setting made in any physics interface that uses this node. For 1D components, see [Change Cross Section](#).

Use the **Change Thickness (Out-of-Plane)** node to set the out-of-plane thickness for specific geometric entities.

CHANGE THICKNESS (OUT-OF-PLANE)

Enter a value or expression for the **Out-of-plane thickness** d (SI unit: m). The default value is, in most cases, 1 unit length, which is typically not representative for a thin domain. Instead it describes a unit thickness that makes the 2D equation identical to the equation used for 3D components.

Charge Conservation, Piezoelectric



This feature is available with the Piezoelectric Devices interface, which requires the Acoustics Module, MEMS Module, or the Structural Mechanics Module. See the individual documentation for information.

The **Charge Conservation, Piezoelectric** is normally used together with a **Piezoelectric Effect** multiphysics coupling node and a corresponding **Piezoelectric Material** node in the **Solid Mechanics** interface. The node is added by default to the **Electrostatics** interface when adding a Piezoelectric Devices interface. It is also available from the context menu (right-click the **Electrostatics** interface parent node) or from the **Physics** toolbar.



When the **Charge Conservation, Piezoelectric** node is added to the electrostatics interface in the absence of an active **Piezoelectric Effect** multiphysics coupling node, the material behaves similarly to a **Charge Conservation** node, with electric properties corresponding to the relative permittivity entered (see below). The piezoelectric effect is *not* included in the corresponding equation system.

ELECTRIC DISPLACEMENT

If the node is used together with an active **Piezoelectric Effect** multiphysics coupling node, then these settings are locked. Note that if they are unlocked, then the material behaves like a dielectric and *not* a piezoelectric. In this case, the default is to take the **Relative permittivity** ϵ_{TS} (dimensionless) values **From material**. For **User defined**, select **Isotropic**, **Diagonal**, **Symmetric**, or **Anisotropic** and enter values or expressions in the field or matrix.

Metal Contact

The **Metal Contact** node is used for modeling what happens when a plasma is in contact with a metal surface. The **Metal Contact** can only be applied on boundaries adjacent to the **Plasma Model** feature. This feature is available when using [The Plasma Interface](#), [The Inductively Coupled Plasma Interface](#), and [The Microwave Plasma Interface](#).

TERMINAL

Specify the terminal properties. To indicate which boundaries belong to the same terminal, enter the same name in the **Terminal name** field.

Select a **Terminal type**—**Voltage** (the default), **Current** or **Circuit**. Select:

- **Voltage** to enter an electric potential V_0 .
- **Current** to enter the Terminal current I_0 . This ensures that the total current arriving on the boundary is equal to the specified value. The current arriving on the boundary is the sum of the displacement current due to the time varying electric potential and conduction current due to the fact that ions and electrons may be incident on the surface.
- **Circuit** to specify a terminal connected to an external circuit. The Terminal node provides a current-voltage characteristic to the circuit element.

If the finite element discretization is employed, then the **Use weak constraints** check box should be activated in the **Constraint Settings** section when using the **Current** or **Circuit** options (see below).

QUICK CIRCUIT SETTINGS

This section is available when **Voltage** is selected as the **Terminal type**. Select a **Quick circuit type**—**None** (the default), **Series RC circuit**, **Ballast resistor**, or **Blocking capacitor**. These settings are the same as described under [Circuit Settings](#) for [Electric Potential](#).

CONSTRAINT SETTINGS

To display this section, click the **Show** button () and select **Advanced Physics Options**.

Dielectric Contact

Use the **Dielectric Contact** node is used for modeling what happens when a plasma is in contact with a dielectric (insulating) surface. The **Dielectric Contact** can only be applied on boundaries adjacent to the **Plasma Model** feature.

SURFACE CHARGE TYPE

Select a Surface charge type—**Surface charge accumulation** (the default) or **Surface charge density**. For **Surface charge density** enter the value or expression for σ_s .

DISTRIBUTED CAPACITANCE

Click to select the **Enable distributed capacitance** check box. Then enter the values or expressions for **Relative permittivity** ϵ_r (dimensionless), **Surface thickness** d_s

(SI unit: m), and **Reference potential** V_{ref} . The default value for the surface thickness is 10^{-3} m (1 mm) and 0 V for the reference potential.

Electrostatic Point Dipole

Add an **Electrostatic Point Dipole** node to 3D and 2D components. Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or click **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.



The **Electrostatic Point Dipole** represents the limiting case of zero separation distance between two equally strong point sources of opposing signs while maintaining the product between separation distance and source strength at a fixed value (p). The dipole moment is a vector entity with positive direction from the negative charge to the positive one.

ELECTROSTATIC POINT DIPOLE

Select a **Dipole specification**—**Magnitude and direction** (the default) or **Electric dipole moment**.

- For **Magnitude and direction** enter coordinates for the **Electric dipole moment direction** \mathbf{n}_p (dimensionless) and the **Electric dipole moment, magnitude** p .
- For **Electric dipole moment** enter coordinates for the **Electric dipole moment** \mathbf{p} .

The Electric Currents Interface

The **Electric Currents (ec)** interface () , found under the **AC/DC** branch () when adding a physics interface, is used to compute electric field, current, and potential distributions in conducting media under conditions where inductive effects are negligible; that is, when the skin depth is much larger than the studied device.

Depending on the licensed products, stationary, frequency-domain, small-signal analysis, and time-domain modeling are supported in all space dimensions. In the time and frequency domains, capacitive effects are also accounted for.

The physics interface solves a current conservation equation based on Ohm's law using the scalar electric potential as the dependent variable.

Current Conservation is the main node, which adds the equation for the electric potential and provides a Settings window for defining the electrical conductivity as well as the constitutive relation for the electric displacement field and its associated material properties, such as the relative permittivity.

When this physics interface is added, these default nodes are also added to the **Model Builder**—**Current Conservation**, **Electric Insulation** (the default boundary condition), and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and current sources. You can also right-click **Electric Currents** to select physics features from the context menu.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the name string must be unique. Only letters, numbers, and underscores (`_`) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `ec`.

PHYSICS-CONTROLLED MESH

Select the **Enable** check box to allow the physics interface to control the meshing process. Information from the physics, such as the presence of an infinite elements

domain or periodic condition, will be used to automatically set up an appropriate mesh sequence.

CROSS-SECTION AREA (1D)

Enter a default value for the **Cross-section area** A (SI unit: m^2). The default value of 1 is typically not representative for a thin domain. Instead it describes a unit thickness that makes the 1D equation identical to the equation used for 3D components. See also [Change Cross Section](#) (described for the Electrostatics interface).

THICKNESS (2D)

Enter a default value for the **Out-of-plane thickness** d (SI unit: m) (see [Equation 3-1](#)). The default value of 1 m is typically not representative for a thin dielectric medium, for example. Instead it describes a unit thickness that makes the 2D equation identical to the equation used for 3D components. See also [Change Thickness \(Out-of-Plane\)](#) (described for the Electrostatics interface).

TERMINAL SWEEP SETTINGS

Enter a **Reference impedance** Z_{ref} (SI unit: Ω). The default is 50 Ω .


Select the **Activate terminal sweep** check box to switch on the sweep and invoke a parametric sweep over the terminals. Enter a **Sweep parameter name** to assign a specific name to the variable that controls the terminal number solved for during the sweep. The default is PortName. The generated lumped parameters are in the form of capacitance matrix elements. The terminal settings must consistently be of either fixed voltage or fixed charge type.



The lumped parameters are subject to **Touchstone file export**. Enter a file path or **Browse** for a file. Select a **Parameter format (value pairs)** for the Touchstone export—**Magnitude and angle (MA)** (the default), **Magnitude in dB and angle (DB)**, or **Real and imaginary parts (RI)**. Select an option from the **If file exists** list—**Overwrite** or **Create new**. Create new is useful when the model is solved multiple times with different settings. Select a **Parameter to export**—**Z** (the default), **Y**, or **S**.

DEPENDENT VARIABLES

The dependent variable is the **Electric potential** V . You can change its name, which changes both the field name and the variable name. If the new name coincides with the name of another electric potential field in the model, the physics interfaces share degrees of freedom. The new name must not coincide with the name of a field of another type or with a component name belonging to some other field.


DISCRETIZATION

To display this section, click the **Show** button () and select **Discretization**.

	<ul style="list-style-type: none">• Domain, Boundary, Edge, Point, and Pair Nodes for the Electric Currents Interface• Theory for the Electric Currents Interface
	<i>Pacemaker Electrode</i> : Application Library path COMSOL_Multiphysics/Electromagnetics/pacemaker_electrode

Domain, Boundary, Edge, Point, and Pair Nodes for the Electric Currents Interface

The [Electric Currents Interface](#) has these domain, boundary, edge, point, and pair nodes available from the **Physics** ribbon toolbar (Windows users) or **Physics** context menu (Mac or Linux users). You can also right-click to access the context menu (all users).

	In general, to add a node, go to the Physics toolbar, no matter what operating system you are using. Subnodes are available by clicking the parent node and selecting it from the Attributes menu.
---	--

ABOUT THE BOUNDARY CONDITIONS

The exterior and interior boundary conditions listed in [Table 3-1](#) are available. The relevant physics interface condition at interfaces between different media and interior boundaries is continuity; that is,

$$\mathbf{n}_2 \cdot (\mathbf{J}_1 - \mathbf{J}_2) = 0$$

which is the natural boundary condition.

AVAILABLE NODES

These nodes are available for this physics interface, listed in alphabetical order. Also see [Table 3-1](#) for a list of interior and exterior boundary conditions, including edge, point, and pair availability.

- [Boundary Current Source](#)
- [Contact Impedance](#)
- [Current Conservation](#)
- [Current Source](#)
- [Distributed Impedance](#)
- [Electric Insulation](#)
- [External Current Density](#)
- [Floating Potential](#)
- [Initial Values](#)
- [Line Current Source](#)
- [Line Current Source \(on Axis\)](#)
- [Normal Current Density](#)
- [Piezoresistive Material](#)¹
- [Point Current Source](#)
- [Sector Symmetry](#)
- [Terminal](#)

¹This feature is available with the Piezoresistivity, Domain Currents interface, which requires the MEMS Module.

These nodes are described for the Electrostatics interface:

- [Change Cross Section](#)
- [Change Thickness \(Out-of-Plane\)](#)
- [Electric Potential](#)
- [Ground](#)
- [Periodic Condition](#)



In the *COMSOL Multiphysics Reference Manual*, see [Table 2-3](#) for links to common sections and [Table 2-4](#) for common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.



For axisymmetric components, COMSOL Multiphysics takes the axial symmetry boundaries (at $r = 0$) into account and automatically adds an **Axial Symmetry** node to the model that is valid on the axial symmetry boundaries only.

Table 3-1 lists the interior and exterior boundaries available with this physics interface. It also includes edge, point, and pair availability.

TABLE 3-2: INTERIOR AND EXTERIOR BOUNDARY CONDITIONS (INCLUDING EDGE, POINT, AND PAIR AVAILABILITY) FOR THE ELECTRIC CURRENTS INTERFACE

NODE	INTERIOR	EXTERIOR	ALSO AVAILABLE FOR
Boundary Current Source	x		pairs
Contact Impedance	x		pairs
Distributed Impedance	x	x	not applicable
Electric Insulation	x	x	pairs
Electric Potential	x	x	edges, points, and pairs
Floating Potential	x	x	pairs
Ground	x	x	edges, points, and pairs
Normal Current Density		x	not applicable
Periodic Condition		x	not applicable
Terminal	x	x	pairs, domains

Current Conservation

The **Current Conservation** node adds the continuity equation for the electrical potential and provides an interface for defining the electric conductivity as well as the constitutive relation and the relative permittivity for the displacement current.

MATERIAL TYPE

The **Material type** setting decides how materials behave and how material properties are interpreted when the mesh is deformed. Select **Solid** for materials whose properties change as functions of material strain, material orientation, and other variables evaluated in a material reference configuration (material frame). Select **Non-solid** for materials whose properties are defined only as functions of the current local state at each point in the spatial frame, and for which no unique material reference configuration can be defined. Select **From material** to pick up the corresponding setting from the domain material on each domain.

CONDUCTION CURRENT

By default, the **Electrical conductivity** σ (SI unit: S/m) for the media is defined **From material**. Or select **User defined** or **Linearized resistivity**.

User Defined

For **User defined** select **Isotropic**, **Diagonal**, **Symmetric**, or **Anisotropic** depending on the characteristics of the electrical conductivity, and then enter values or expressions for the electrical conductivity σ in the field or matrix. The default is 0 S/m. If type of temperature dependence is used other than a linear temperature relation, enter any expression for the conductivity as a function of temperature.

Linearized Resistivity

Select **Linearized resistivity** for a temperature-dependent conductivity (this occurs in, for example, Joule heating, and is also called resistive heating). The equation describing the conductivity:

$$\sigma = \frac{1}{\rho_0(1 + \alpha(T - T_0))}$$

where ρ_0 is the resistivity at the reference temperature T_0 , and α is the temperature coefficient of resistance, which describes how the resistivity varies with temperature.

The default **Reference resistivity** ρ_0 (SI unit: $\Omega \cdot m$), **Reference temperature** T_{ref} (SI unit: K), and **Resistivity temperature coefficient** α (SI unit: 1/K) are taken **From material**, which means that the values are taken from the domain (or boundary) material. T is the current temperature, which can be a value that is specified as a model input or the temperature from a heat transfer interface. The definition of the temperature field is in the **Model Inputs** section.

To specify other values for any of these properties, select **User defined** from the list and then enter a value or expression for each. The default values are:

- 1 $\Omega \cdot m$ for the Reference resistivity
- 273.15 K for the Reference temperature, and
- 0 1/K for the Resistivity temperature coefficient

ELECTRIC FIELD

Select a **Constitutive relation** to describe the macroscopic properties of the medium (relating the electric displacement **D** with the electric field **E**) and the applicable material properties, such as the relative permittivity. For a description of the constitutive relations **Relative permittivity**, **Polarization**, and **Remanent electric**

displacement, see **Electric Field** as described for the [Charge Conservation](#) node for the Electrostatics interface. The constitutive relations specific to Electric Currents are:

- **Dielectric losses**: uses the constitutive relation $\mathbf{D} = \epsilon_0(\epsilon' + j\epsilon'')\mathbf{E}$. Specify that the **Relative permittivity (real part)** ϵ' (dimensionless) and the **Relative permittivity (imaginary part)** ϵ'' (dimensionless) must be taken **From material** or be **User defined**. For **User defined**, select **Isotropic**, **Diagonal**, **Symmetric**, or **Anisotropic** and enter values or expressions in the field or matrix. The default is 1.
- **Loss tangent, loss angle**: uses the constitutive relation $\mathbf{D} = \epsilon_0\epsilon'(1 + j\tan\delta)\mathbf{E}$. Specify the **Relative permittivity (real part)** ϵ' (dimensionless) and **Loss angle** δ (SI unit: rad).
- **Loss tangent, dissipation factor**: uses the constitutive relation $\mathbf{D} = \epsilon_0\epsilon'(1 + j\tan\delta)\mathbf{E}$. Specify the **Relative permittivity (real part)** ϵ' (dimensionless) and the **Dissipation factor** $\tan\delta$ (dimensionless).

Initial Values

The **Initial Values** node adds an initial value for the electric potential that can serve as an initial condition for a transient simulation or as an initial guess for a nonlinear solver. If more than one set of initial values is required, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and current sources. Add more **Initial Values** nodes from the **Physics** toolbar.

INITIAL VALUES

Enter a value or expression for the initial value of the **Electric potential** V (SI unit: V). The default value is 0 V.

External Current Density

The **External Current Density** node adds an externally generated current density \mathbf{J}_e , which appears in Ohm's law

$$\mathbf{J} = \sigma\mathbf{E} + \mathbf{J}_e$$

and in the equation that the physics interface defines.

The external current density does not contribute to the losses (due to Joule heating), since there is no electric field associated with it. To include the contribution to the losses from the external current density, select the **Add contribution of the external current density to the losses** check box. Then select an option from the **External losses** list—**From domain conductivity** (the default) or **User defined**. If **From domain**

conductivity is selected, the heat source is computed using the conductivity specified in the material model feature (such as **Current Conservation**) that is applied in the domain. For **User defined**, enter a value for Q_e (SI unit: W/m^3) to specify a user-defined heat source.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

EXTERNAL CURRENT DENSITY

Based on space dimension, enter the coordinates (**x**, **y**, and **z** for 3D components, for example) of the **External current density \mathbf{J}_e** (SI unit: A/m^2). The defaults are $0 \text{ A}/\text{m}^2$.

Current Source

The **Current Source** node adds a distributed current source Q_j in the equation that the physics interface defines. Use this node with caution as it can violate the current conservation law that is inherent in Maxwell-Ampère's law.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

CURRENT SOURCE

Enter a value or expression for the **Current source Q_j** (SI unit: A/m^3). The default is $0 \text{ A}/\text{m}^3$.

Electric Insulation

The **Electric Insulation** node, which is the default boundary condition, adds electric insulation as the boundary condition:

$$\mathbf{n} \cdot \mathbf{J} = 0$$

This boundary condition means that no electric current flows into the boundary. At interior boundaries, it means that no current can flow through the boundary and that the electric potential is discontinuous across the boundary. It is also applicable at symmetric boundaries where the potential is known to be symmetric with respect to the boundary.

Electric insulation as the default boundary condition is not applicable to interior boundaries. To add electric insulation to an interior boundary, add an **Electric Insulation** node in addition to the one that represents the default boundary condition.

Floating Potential

The **Floating Potential** node is useful when modeling a metallic electrode at floating potential. This is a good approximation when the conductivity of the electrode is many orders of magnitude larger than that of the surrounding medium. The electrode might inject a current I_0 in the model, that can be specified by the user or obtained from a circuit connection.

The **Harmonic Perturbation** subnode (it is of the exclusive type) is available from the context menu (right-click the parent node) or on the **Physics** toolbar, click the **Attributes** menu and select **Harmonic Perturbation**. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

FLOATING POTENTIAL

The Floating potential identifier text area shows the unique identifier for the floating potential feature. It is used to identify the global variables created by the node, such as the voltage. The **Floating potential group** check box on the **Settings** window for **Floating Potential** controls how potentials are assigned to boundary segments in the feature's boundary selection. If this check box is not selected, a single potential is applied to all boundaries in the selection. If the check box is selected, each group of contiguous boundaries in the selection is given a unique potential. This simplifies the setup of models with many floating electrodes. The values of the potential at each group of boundaries are then made available in postprocessing, collected in a vector variable.

The following options are not available if the **Floating potential group** check box is selected. Select a specification for the **Floating potential current**—choose **User defined** (the default) to specify a total **Current** I_0 (SI unit: A) that flows from the electrode. The default is 0 A, corresponding to an unconnected electrode. Select **Circuit** to connect the floating potential to an Electrical Circuit interface.

Initial values

The initial values subsection is used to provide initial values for the internal state variables in the Floating Potential. These settings can be useful to provide an initial guess to the nonlinear solver or the initial value in a time-dependent study when the initial values of the other dependent variables (such as the electric potential) are

computed from a separate physics or study. In most cases, the default values are appropriate.

Enter an **Initial value for voltage** V_{init} (SI unit: V). The default is 0 V.

CONSTRAINT SETTINGS

To display this section, click the **Show** button () and select **Advanced Physics Options**.



[Settings and Properties Windows for Features Nodes](#) in the *COMSOL Multiphysics Reference Manual*

Boundary Current Source

The **Boundary Current Source** node adds a current source Q_j on the boundary.

$$\mathbf{n} \cdot (\mathbf{J}_1 - \mathbf{J}_2) = Q_j$$

It is applicable to interior boundaries that represent either a source or a sink of current. Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or click **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

BOUNDARY CURRENT SOURCE

Enter a value or expression for the **Boundary current source** Q_j (SI unit: A/m²). The default is 0 A/m².

Normal Current Density

The **Normal Current Density** node is applicable to exterior boundaries that represent either a source or a sink of current. It provides a condition for specifying the normal current density as an inward or outward current flow:

$$-\mathbf{n} \cdot \mathbf{J} = J_n$$

Or alternatively, as a current density \mathbf{J}_0 :

$$\mathbf{n} \cdot \mathbf{J} = \mathbf{n} \cdot \mathbf{J}_0$$

The normal current density is positive when the current flows inward in the domain. Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

NORMAL CURRENT DENSITY

Select a **Type**—**Inward current density** (the default) or **Current density**.

- For **Inward current density** enter a value or expression for the **Normal current density** J_n (SI unit: A/m²). Use a positive value for an inward current flow or a negative value for an outward current flow. The default is 0 A/m².
- For **Current density** enter values or expressions for the components of the **Current density** \mathbf{J}_0 (SI unit: A/m²). The defaults are 0 A/m².

Distributed Impedance

The **Distributed Impedance** node adds a distributed impedance boundary condition to a model.

The **Harmonic Perturbation** subnode (it is of the exclusive type) is available from the context menu (right-click the parent node) or on the **Physics** toolbar, click the **Attributes** menu and select **Harmonic Perturbation**. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

Use this boundary condition to model a thin sheet of a resistive material connected to a reference potential V_{ref} .

The layer impedance can be specified either with the bulk material conductivity σ_s , the relative permittivity ϵ_r and layer thickness d_s , or directly with the surface resistance ρ_s and capacitance C_s . Assuming DC currents, the equation is:

$$\mathbf{n} \cdot (\mathbf{J}_1 - \mathbf{J}_2) = \frac{\sigma_s}{d_s} (V - V_{\text{ref}})$$

$$\mathbf{n} \cdot (\mathbf{J}_1 - \mathbf{J}_2) = \frac{1}{\rho_s} (V - V_{\text{ref}})$$

For the Frequency Domain and Time Dependent study types, this boundary condition is slightly more sophisticated and accounts also for capacitive coupling. The equations are:

$$\mathbf{n} \cdot (\mathbf{J}_1 - \mathbf{J}_2) = \frac{(\sigma + j\omega\epsilon_0\epsilon_r)}{d_s}(V - V_{\text{ref}})$$

$$\mathbf{n} \cdot (\mathbf{J}_1 - \mathbf{J}_2) = \left(\frac{1}{r_s} + j\omega C_s\right)(V - V_{\text{ref}})$$

$$\mathbf{n} \cdot (\mathbf{J}_1 - \mathbf{J}_2) = \frac{1}{d_L} \left(\sigma(V - V_{\text{ref}}) + \epsilon_0\epsilon_r \frac{\partial}{\partial t}(V - V_{\text{ref}}) \right)$$

$$\mathbf{n} \cdot (\mathbf{J}_1 - \mathbf{J}_2) = \left(\frac{1}{\rho_s}(V - V_{\text{ref}}) + C_s \frac{\partial}{\partial t}(V - V_{\text{ref}}) \right)$$

DISTRIBUTED IMPEDANCE

Enter the reference potential V_{ref} (SI unit: V). The default is 0 V.

Select a potentially complex-valued **Layer specification**—**Thin layer** (the default) or **Surface impedance**.

- For **Thin layer**, enter values or expressions for the:
 - **Surface thickness** d_s (SI unit: m). The default is $5 \cdot 10^{-3}$ m (5 mm).
 - **Electrical conductivity** σ (SI unit: S/m) and **Relative permittivity** ϵ_r (dimensionless). The defaults take values **From material**. For **User defined**, enter different values or expressions. The default electrical conductivity is 1×10^{-2} S/m and the default relative permittivity is 1.
- For **Surface impedance**, enter values or expressions for the **Surface resistance** ρ_s (SI unit: $\Omega \cdot \text{m}^2$) and the **Surface capacitance** C_s (SI unit: F/m²). The default surface impedance is 1×10^{-8} $\Omega \cdot \text{m}^2$ and the default surface capacitance is 0 F/m².

Terminal

The **Terminal** node provides a boundary or domain condition for connection to external circuits, to transmission lines, or with a specified voltage or current. By specifying zero current, a floating potential condition is obtained. When the **Terminal** node is applied on the domain level, the electric potential is set to constant on the entire domain selection.

The **Terminal** node also enables the computation of the lumped parameter of the system, such as the impedance.

The **Harmonic Perturbation** subnode (it is of the exclusive type) is available from the context menu (right-click the parent node) or on the **Physics** toolbar, click the **Attributes** menu and select **Harmonic Perturbation**. The **Harmonic Perturbation** subnode

only has effect if **Current** or **Voltage** are selected as terminal type. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

TERMINAL

Specify the terminal's properties. The **Terminal name** identifies the terminal and the variables it creates. The name should be numeric for sweeps to work properly.

Select a **Terminal type**—**Current** (the default), **Voltage**, **Circuit**, **Terminated**, or **Power**.



The **Circuit** and **Power** types cannot be used together with a terminal sweep.

Select:

- **Current** to specify a total current flowing from the terminal. Enter the current I_0 (SI unit: A). The default is 0 A for an open circuit.
- **Voltage** to specify a fixed electric potential at the terminal. Enter the value of the electric potential V_0 (SI unit: V). The default is 1 V.
- **Circuit** to specify a terminal connected to an Electrical Circuit interface that models an external circuit.
- **Terminated** to connect the terminal to an impedance that might represent a load or a transmission line. When **Terminated** is selected, the scattering parameters (S-parameters) are computed.
 - Enter a **Terminal power** P_0 (SI unit: W) to specify the input power at this terminal from the transmission line. This excitation can be toggled using a terminal sweep. The default is 1 W.
 - Select an option from the **Characteristic impedance from** list to define the value of the impedance—**Physics interface** (the default) or **User defined**. For **Physics interface** the **Reference impedance** Z_{ref} defined on the **Settings** window for the

physics interface under Sweep Settings is used. For **User defined** enter a specific impedance Z_{ref} (SI unit: Ω) for this terminal. The default is 50 Ω .

- **Power** to specify a fixed electrical power for the terminal. The power is computed as the product of the terminal current and voltage and, in frequency domain, is the cycle-averaged power. When **Power** is selected:
 - Enter a **Terminal power** P_0 (SI unit: W) to specify the input power at this terminal. The default is 1 W.



Using Power excitation causes the equation system to become nonlinear. COMSOL Multiphysics automatically adjusts the solver settings when Power is selected.

Be aware that, in general, the values of the voltage and the current might not uniquely be determined by this constraint and may depend on the initial value provided for the current and voltage variables. Particularly in the frequency domain, the absolute phase of the quantities can be arbitrary. This indeterminacy can have an impact on the solution process. See [Compile Equations](#) in the *COMSOL Multiphysics Reference Manual* for a possible solution to this problem.

Initial Values

The initial values subsection is used to provide initial values for the internal state variables in the Terminal. These settings can be useful to provide an initial guess to the nonlinear solver or the initial value in a time-dependent study when the initial values of the other dependent variables (such as the electric potential) are computed from a separate physics or study. In most cases, the default values are appropriate.

The **Initial value for voltage** is available for all Terminal types except Voltage. Enter an **Initial value for voltage** V_{init} (SI unit: V). The default is 0 V.

The **Initial value for current** is available for Power only. In this case, it provides an initial guess for the nonlinear solver, and should not be set to zero. The initial value provided may affect the solution. Enter an **Initial value for current** I_{init} (SI unit: A). The default is 1 mA.

CONSTRAINT SETTINGS

To display this section, click the **Show** button () and select **Advanced Physics Options**.

Contact Impedance

Use the **Contact Impedance** node on interior boundaries to model a thin layer of resistive material. It can also be added as a pair using a **Pair Contact Impedance** node. The feature allows specifying the contact impedance either by entering the properties of the material together with the layer thickness, or by entering the impedance properties of the thin layer directly.

The feature, in stationary studies, applies the following conditions that relate the normal electric current density with the jump in the electric potential:

$$\mathbf{n} \cdot \mathbf{J}_1 = \frac{\sigma}{d_s}(V_1 - V_2)$$

$$\mathbf{n} \cdot \mathbf{J}_2 = \frac{\sigma}{d_s}(V_2 - V_1)$$

$$\mathbf{n} \cdot \mathbf{J}_1 = \frac{1}{\rho_s}(V_1 - V_2)$$

$$\mathbf{n} \cdot \mathbf{J}_2 = \frac{1}{\rho_s}(V_2 - V_1)$$

The first two equations refer to a layer impedance specified using the bulk material conductivity σ_s and the layer thickness d_s , while the last two equations refer to the case in which the surface resistance ρ_s is specified. The indices 1 and 2 refer to the two sides of the boundary. These parameters work the same as with [Distributed Impedance](#).

For the frequency domain and Time Dependent study types, this boundary condition is slightly more sophisticated and accounts also for capacitive coupling. The bulk material relative permittivity ϵ_r or the layer capacitance C_s can be specified.

The equations used in Frequency Domain studies are:

$$\mathbf{n} \cdot \mathbf{J}_1 = \frac{(\sigma + j\omega\epsilon_0\epsilon_r)}{d_s}(V_1 - V_2)$$

$$\mathbf{n} \cdot \mathbf{J}_2 = \frac{(\sigma + j\omega\epsilon_0\epsilon_r)}{d_s}(V_2 - V_1)$$

$$\mathbf{n} \cdot \mathbf{J}_1 = \left(\frac{1}{\rho_s} + j\omega C_s\right)(V_1 - V_2)$$

$$\mathbf{n} \cdot \mathbf{J}_2 = \left(\frac{1}{\rho_s} + j\omega C_s\right)(V_2 - V_1)$$

In Time Dependent studies:

$$\mathbf{n} \cdot \mathbf{J}_1 = \frac{1}{d_s} \left(\sigma(V_1 - V_2) + \epsilon_0 \epsilon_r \frac{\partial}{\partial t} (V_1 - V_2) \right)$$

$$\mathbf{n} \cdot \mathbf{J}_2 = \frac{1}{d_s} \left(\sigma(V_2 - V_1) + \epsilon_0 \epsilon_r \frac{\partial}{\partial t} (V_2 - V_1) \right)$$

$$\mathbf{n} \cdot \mathbf{J}_1 = \left(\frac{1}{\rho_s} (V_1 - V_2) + C_s \frac{\partial}{\partial t} (V_1 - V_2) \right)$$

$$\mathbf{n} \cdot \mathbf{J}_2 = \left(\frac{1}{\rho_s} (V_2 - V_1) + C_s \frac{\partial}{\partial t} (V_2 - V_1) \right)$$

CONTACT IMPEDANCE

Select a potentially complex-valued **Layer specification—Thin layer** (the default) or **Surface impedance**.

- For **Thin layer**, enter values or expressions for the:
 - **Surface thickness** d_s (SI unit: m). The default is $5 \cdot 10^{-3}$ m (5 mm).
 - **Electrical conductivity** σ (SI unit: S/m) and **Relative permittivity** ϵ_r (dimensionless). The defaults take values **From material**. For **User defined**, enter different values or expressions. The default electrical conductivity is 1×10^{-2} S/m and the default relative permittivity is 1.
- For **Surface impedance**, enter values or expressions for the **Surface resistance** ρ_s (SI unit: $\Omega \cdot \text{m}^2$) and the **Surface capacitance** C_s (SI unit: F/m²). The default surface impedance is 1×10^{-8} $\Omega \cdot \text{m}^2$ and the default surface capacitance is 0 F/m².



Thin-Film Resistance: Application Library path **COMSOL_Multiphysics/Electromagnetics/thin_film_resistance**

Sector Symmetry

Select **Sector Symmetry** at interfaces between rotating objects where sector symmetry is used. It is only available for pairs. A default subnode is added. Right-click to select

additional features from the **Fallback Features** submenu. In 2D, this feature assumes rotation around the origin.



This feature is always used in conjunction with a [Periodic Condition](#) on adjacent radial sector boundaries. Note that the same periodic condition feature cannot be used on both sides of where the sector symmetry boundaries connect with the periodic boundaries. At least two periodic condition features are required for the model to compute correctly.

PAIR SELECTION

When using nonconforming meshes on the source and destination of a pair, for numerical stability, a finer mesh should be applied on the destination side for any pair with a condition that imposes a coupling or a constraint across the pair. The sector symmetry feature falls into this category.

SECTOR SETTINGS

Enter the **Number of sectors (<50)** n_{sect} . The default is 2.

Select a **Type of periodicity**—**Continuity** (the default) or **Antiperiodicity**.

Based on space dimension, enter values or expressions in the table for the **Axis of rotation** \mathbf{a}_{rot} .

CONSTRAINT SETTINGS

To display this section, click the **Show** button () and select **Advanced Physics Options**.

Line Current Source

The **Line Current Source** node adds a line source to edges in 3D components and to points in 2D and 2D axisymmetric components. The line source represents electric current per unit length.

EDGE OR POINT SELECTION



Beware that constraining the potential on edges or points usually yields a current outflow that is mesh dependent.

LINE CURRENT SOURCE

Enter a value or expression to apply a **Line current source** Q_j (SI unit: A/m). This source represents electric current per unit length. The default is 0 A/m.



Line Current Source (on Axis) for 2D axisymmetric components.

Line Current Source (on Axis)

The **Line Current Source (on Axis)** node adds a line source to boundaries in 2D axisymmetric components. The line source represents electric current per unit length.

LINE CURRENT SOURCE (ON AXIS)

Enter a value or expression to apply a **Line current source** Q_j (SI unit: A/m) to boundaries. This source represents electric current per unit length.



Line Current Source

Point Current Source

The **Point Current Source** node adds a point source and represents an electric current flowing out of the point. Add point sources to 3D components from the **Points** menu. Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** on the **Physics** toolbar. For more information see [Harmonic Perturbation — Exclusive and Contributing Nodes](#) in the *COMSOL Multiphysics Reference Manual*.

POINT SELECTION



Beware that constraining the potential on points usually yields a current outflow that is mesh dependent.

POINT CURRENT SOURCE

Enter a value or expression to apply a **Point current source** Q_j (SI unit: A) to points.
This source represents an electric current flowing out of the point.



- [Line Current Source](#) for 2D components
- [Point Current Source \(on Axis\)](#) for 2D axisymmetric components

Point Current Source (on Axis)

The **Point Current Source (on Axis)** node adds a point source and represents an electric current flowing out of the point in 2D axisymmetric components.

POINT SELECTION



Beware that constraining the potential on points usually yields a current outflow that is mesh dependent.

POINT CURRENT SOURCE

Enter a value or expression to apply a **Point current source** Q_j (SI unit: A) to points.
This source represents an electric current flowing out of the point.



- [Point Current Source](#) for 3D components
- [Line Current Source](#) for 2D components

Piezoresistive Material

The **Piezoresistive Material** is normally used together with a **Piezoresistive Effect, Domain Currents** multiphysics coupling node. The node is added by default to the **Electric Currents** interface when adding a **Piezoresistivity, Domain Currents** predefined

multiphysics coupling interface. It is also available from the context menu (right-click the **Electric Currents** interface parent node) or from the **Physics** toolbar.





When the **Piezoresistive Material** node is added to the Electric Currents interface in the absence of an active **Piezoelectric Effect, Domain Currents** multiphysics coupling node, the material behaves similarly to a **Current Conservation** node, with electric properties corresponding to the relative permittivity and electrical conductivity entered. The piezoresistive effect is *not* included in the corresponding equation system.

PIEZORESISTIVE MATERIAL PROPERTY

This node should be used together with an active **Piezoresistive Effect, Domain Currents** multiphysics coupling node. Select a **Constitutive model**—**Piezoresistance form** or **Elastoresistance form**. For each of the following, the default uses values **From material**. For **User defined** enter other values in the matrix or field.

- Specify a **Electrical conductivity, zero stress** (SI unit: S/m). This typically comes from the material added under the Materials node.
- For **Piezoresistance form**, select a **Piezoresistance coupling matrix** Π_l (SI unit: m⁴/(s·A²), note that this is equivalent to Ω·m/Pa).
- For a **Elastoresistance form**, select an **Elastoresistance coupling matrix** M_l (SI unit: Ω·m).

The Electrical Circuit Interface

The **Electrical Circuit (cir)** interface () , found under the **AC/DC** branch () when adding a physics interface, is used to model currents and voltages in circuits including voltage and current sources, resistors, capacitors, inductors, and semiconductor devices. Models created with the Electrical Circuit interface can include connections to distributed field models. The physics interface supports stationary, frequency-domain and time-domain modeling and solves Kirchhoff's conservation laws for the voltages, currents and charges associated with the circuit elements.

When this physics interface is added, it adds a default **Ground Node** feature and associates that with node zero in the electrical circuit.



Circuit nodes are nodes in the electrical circuit (electrical nodes) and should not be confused with nodes in the Model Builder tree of the COMSOL Multiphysics software. Circuit node names are not restricted to numerical values but can contain alphanumeric characters.

DEVICE NAMES

Each circuit component has an associated **Device name**, which is constructed from a prefix identifying the type of the device and a string. The string can be specified in the feature's **Settings** window. The **Device name** is used to identify variables defined by the component, and for the SPICE Import and Export functionality.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the **name** string must be unique. Only letters, numbers and underscores (`_`) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `cir`.

RESISTANCE IN PARALLEL TO PN JUNCTIONS

For numerical stability, a large resistance is added automatically in parallel to the pn junctions in diodes and BJT devices. Enter a default value for the **Resistance in parallel to pn junctions** R_j (SI unit: Ω). The default value is $1 \cdot 10^{12} \Omega$.

CREATE UNIQUE NODES FOR NEW DEVICES

When this setting is selected (the default), newly added devices will be assigned unused node names. The devices will be disconnected from the rest of the circuit and the nodes should be updated to reflect the actual circuit connections. When the this setting is deselected, new devices will be connected to the lowest-numbered nodes starting from 0.



- [Theory for the Electrical Circuit Interface](#)
 - [Connecting to Electrical Circuits](#)
-

ELECTRICAL CIRCUIT TOOLBAR

The following nodes are available from the **Electrical Circuit** ribbon toolbar (Windows users), **Electrical Circuit** context menu (Mac or Linux users), or right-click to access the context menu (all users).:



For step-by-step instructions and general documentation descriptions, this is the **Electrical Circuit** toolbar.


- Ground Node
- Resistor
- Capacitor
- Inductor
- Voltage Source
- Current Source
- Diode
- Voltage-Controlled Voltage Source¹
- Voltage-Controlled Current Source¹
- Current-Controlled Voltage Source¹
- Current-Controlled Current Source¹
- Subcircuit Definition
- Subcircuit Instance
- Mutual Inductance
- Transformer
- NPN BJT²
- PNP BJT²
- n-Channel MOSFET²
- p-Channel MOSFET²
- External I vs. U³
- External U vs. I³
- External I-Terminal³
- SPICE Circuit Import
- SPICE Circuit Export

¹ Selected from the **Dependent Sources** submenu when you right-click main node.

² Selected from the **Transistors** submenu when you right-click main node.

³ Selected from the **External Couplings** submenu when you right-click main node.


Ground Node

The **Ground Node** () feature adds a ground node with the default node number zero to the electrical circuit. This is the default node in the Electrical Circuit interface. More ground nodes can be added but those must have unique node numbers and are by default given higher node numbers.

GROUND CONNECTION

Set the **Node name** for the ground node in the circuit. The convention is to use 0 (zero) for the ground node. If adding more ground nodes, each must have a unique node name (number).

Resistor

The **Resistor** () feature connects a resistor between two nodes in the electrical circuit.

DEVICE NAME

Enter a **Device name** for the resistor. The prefix is R.


NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the resistor.

DEVICE PARAMETERS

Enter the **Resistance** of the resistor.

Capacitor

The **Capacitor** () feature connects a capacitor between two nodes in the electrical circuit.

DEVICE NAME

Enter a **Device name** for the capacitor. The prefix is C.


NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the capacitor.

DEVICE PARAMETERS

Enter the **Capacitance** of the capacitor.

Inductor

The **Inductor** () feature connects an inductor between two nodes in the electrical circuit.

DEVICE NAME

Enter a **Device name** for the inductor. The prefix is L.


NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the inductor.

DEVICE PARAMETERS

Enter the **Inductance** of the inductor.

Voltage Source

The **Voltage Source** () feature connects a voltage source between two nodes in the electrical circuit.

DEVICE NAME

Enter a **Device name** for the voltage source. The prefix is V.

NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the voltage source. The first node represents the positive reference terminal.

DEVICE PARAMETERS

Enter the **Source type** that should be adapted to the selected study type. It can be **DC-source**, **AC-source**, or a time-dependent **Sine source** or **Pulse source**. Depending on the choice of source, also specify the following parameters:


- For a DC-source, the **Voltage V_{src}** (default value: 1 V). DC-sources are active in Stationary and Time-Dependent studies.
- For an AC-source: the **Voltage V_{src}** (default value: 1 V) and the **Phase Θ** (default value: 0 rad). AC-sources are active in Frequency Domain studies only.
- For a sine source: the **Voltage V_{src}** (default value: 1 V), the **Offset V_{off}** (default value: 0 V), the **Frequency** (default value: 1 kHz), and the **Phase Θ** (default value: 0 rad). The sine sources are active in Time-Dependent studies and also in Stationary studies, providing that a value for t has been provided as a model parameter or global variable.
- For a pulse source: the **Voltage V_{src}** (default value: 1 V), the **Offset V_{off}** (default value: 0 V), the **Delay t_d** (default value: 0s), the **Rise time t_r** and **Fall time t_f** (default values: 0 s), the **Pulse width p_w** (default value: 1 μ s), and the **Period T_{per}** (default value: 2 μ s). The pulse sources are active in Time-Dependent studies and also in Stationary studies, providing that a value for t has been provided as a model parameter or global variable.

All values are peak values rather than RMS.



For the AC source, the frequency is a global input set by the solver. AC sources should be used in Frequency-domain studies only. Do not use the **Sine source** unless the model is time-dependent.

Current Source

The **Current Source** () feature connects a current source between two nodes in the electrical circuit.

DEVICE NAME

Enter a **Device name** for the current source. The prefix is I.

NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the current source. The first node represents the positive reference terminal from where the current flows through the source to the second node.

DEVICE PARAMETERS

Enter the **Source type** that should be adapted to the selected study type. It can be **DC-source**, **AC-source**, or a time-dependent **Sine source** or **Pulse source**. Depending on the choice of source, also specify the following parameters:


- For a DC-source, the **Current i_{src}** (default value: 1 A). DC-sources are active in Stationary and Time-Dependent studies.
- For an AC-source: the **Current i_{src}** (default value: 1 A) and the **Phase Θ** (default value: 0 rad). AC-sources are active in Frequency Domain studies only.
- For a sine source: the **Current i_{src}** (default value: 1 A), the **Offset i_{off}** (default value: 0 A), the **Frequency** (default value: 1 kHz), and the **Phase Θ** (default value: 0 rad). The sine sources are active in Time-Dependent studies and also in Stationary studies, providing that a value for t has been provided as a model parameter or global variable.
- For a pulse source: the **Current i_{src}** (default value: 1 A), the **Offset i_{off}** (default value: 0 A), the **Delay t_d** (default value: 0 s), the **Rise time t_r** and **Fall time t_f** (default values: 0 s), the **Pulse width p_w** (default value: 1 μ s), and the **Period T_{per}** (default value: 2 μ s). The pulse sources are active in Time-Dependent studies and also in Stationary studies, providing that a value for t has been provided as a model parameter or global variable.

All values are peak values rather than RMS.



For the AC source, the frequency is a global input set by the solver. AC sources should be used in Frequency-domain studies only. Do not use the **Sine source** unless the model is time-dependent.

Voltage-Controlled Voltage Source

The **Voltage-Controlled Voltage Source** () feature connects a voltage-controlled voltage source between two nodes in the electrical circuit. A second pair of nodes define the input control voltage.

DEVICE NAME

Enter a **Device name** for the voltage-controlled voltage source. The prefix is E.


NODE CONNECTIONS

Specify four **Node names**: the first pair for the connection nodes for the voltage source and the second pair defining the input control voltage. The first node in a pair represents the positive reference terminal.

DEVICE PARAMETERS

Enter the voltage **Gain**. The resulting voltage is this number multiplied by the control voltage.

Voltage-Controlled Current Source

The **Voltage-Controlled Current Source** () feature connects a voltage-controlled current source between two nodes in the electrical circuit. A second pair of nodes define the input control voltage.

DEVICE NAME

Enter a **Device name** for the voltage-controlled current source. The prefix is G.


NODE CONNECTIONS

Specify four **Node names**: the first pair for the connection nodes for the current source and the second pair defining the input control voltage. The first node in a pair represents the positive voltage reference terminal or the one from where the current flows through the source to the second node.

DEVICE PARAMETERS

Enter the source **Gain** (SI units: S). The resulting current is this number multiplied by the control voltage. It represents the transconductance of the source.

Current-Controlled Voltage Source

The **Current-Controlled Voltage Source** () feature connects a current-controlled voltage source between two nodes in the electrical circuit. The input control current is the one flowing through a two-pin device.

DEVICE NAME

Enter a **Device name** for the current-controlled voltage source. The prefix is H.


NODE CONNECTIONS

Set two **Node names** for the connection nodes for the voltage source. The first node in a pair represents the positive reference terminal.

DEVICE PARAMETERS

Enter the voltage **Gain** and select the **Device** whose current is taken as the control current. The resulting voltage is this number multiplied by the control current through the named **Device** (any two-pin device). Thus it formally has the unit of resistance.

Current-Controlled Current Source

The **Current-Controlled Current Source** () feature connects a current-controlled current source between two nodes in the electrical circuit. The input control current is the one flowing through a named device that must be a two-pin device.

DEVICE NAME

Enter a **Device name** for the current-controlled current source. The prefix is F.


NODE CONNECTIONS

Specify two **Node names** for the connection nodes for the current source. The first node in a pair represents the positive reference terminal from where the current flows through the source to the second node.

DEVICE PARAMETERS

Enter the current **Gain** and select the **Device** whose current is taken as the control current. The resulting current is this number multiplied by the control current through the **Device**.


Subcircuit Definition

The **Subcircuit Definition** () feature is used to define subcircuits, which can be inserted as devices into the main circuit using [Subcircuit Instance](#) nodes. Create the subcircuit by adding subnodes to the **Subcircuit Definition** node, either by using the **Physics** toolbar, or by right-clicking the **Subcircuit Definition**.

SUBCIRCUIT PINS

Define the **Pin names** at which the subcircuit connects to the main circuit or to other subcircuits when referenced by a [Subcircuit Instance](#) node. The **Pin names** refer to circuit nodes in the subcircuit. The order in which the **Pin names** are defined is the order in which they are referenced by a Subcircuit Instance node. The devices constituting the subcircuit should be connected only to the subcircuit's pins and to themselves.

Subcircuit Instance

The **Subcircuit Instance** () feature represents an instance of a subcircuits defined by a [Subcircuit Definition](#) feature.


DEVICE NAME

Enter a **Device name** for the subcircuit instance. The prefix is X.

NODE CONNECTIONS

Select the **Name of subcircuit link** from the list of defined subcircuits in the circuit model and the circuit **Node names** at which the subcircuit instance connects to the main circuit or to another subcircuit if used therein.

NPN BJT

The **NPN BJT** device model () is a large-signal model for an NPN bipolar junction transistor (BJT). It is an advanced device model and no thorough description and motivation of the many input parameters are attempted here. Many device manufacturers provide model input parameters for this BJT model. For any particular make of BJT, the device manufacturer should be the primary source of information.

DEVICE NAME

Enter a **Device name** for the BJT. The prefix is Q.

NODE CONNECTIONS

Specify three **Node names** for the connection nodes for the **NPN BJT** device. These represent the *collector*, *base*, and *emitter* nodes, respectively.

MODEL PARAMETERS


Specify the **Model Parameters**. Reasonable defaults are provided but for any particular BJT, the device manufacturer should be the primary source of information.



The interested reader is referred to [Ref. 1](#) for more details on semiconductor modeling within circuits.

For an explanation of the **Model Parameters** see [Bipolar Transistors](#).

PNP BJT

The **PNP BJT** device model () is a large-signal model for a PNP bipolar junction transistor (BJT). It is an advanced device model and no thorough description and motivation of the many input parameters are attempted here. Many device manufacturers provide model input parameters for this BJT model. For any particular make of BJT, the device manufacturer should be the primary source of information.

DEVICE NAME

Enter a **Device name** for the BJT. The prefix is Q.


NODE CONNECTIONS

Specify three **Node names** for the connection nodes for the **PNP BJT** device. These represent the *emitter*, *base*, and *collector* nodes, respectively.

MODEL PARAMETERS

Specify the **Model Parameters**. Reasonable defaults are provided but for any particular BJT, the device manufacturer should be the primary source of information.

n-Channel MOSFET

The **n-Channel MOSFET** device model () is a large-signal model for an n-Channel MOS transistor (MOSFET). It is an advanced device model and no thorough description and motivation of the many input parameters are attempted here. Many device manufacturers provide model parameters for this MOSFET model. For any

particular make of MOSFET, the device manufacturer should be the primary source of information.

DEVICE NAME

Enter a **Device name** for the MOSFET. The prefix is M.

NODE CONNECTIONS

Specify four **Node names** for the connection nodes for the **n-Channel MOSFET** device. These represent the *drain*, *gate*, *source*, and *bulk* nodes, respectively.

MODEL PARAMETERS


Specify the **Model Parameters**. Reasonable defaults are provided but for any particular MOSFET, the device manufacturer should be the primary source of information.



The interested reader is referred to [Ref. 1](#) for more details on semiconductor modeling within circuits.

For an explanation of the **Model Parameters** see [MOSFET Transistors](#).

p-Channel MOSFET

The **p-Channel MOSFET** device model () is a large-signal model for a p-Channel MOS transistor (MOSFET). It is an advanced device model and no thorough description and motivation of the many input parameters are attempted here. Many device manufacturers provide model parameters for this MOSFET model. For any particular make of MOSFET, the device manufacturer should be the primary source of information.

DEVICE NAME

Enter a **Device name** for the MOSFET. The prefix is M.

NODE CONNECTIONS

Specify four **Node names** for the connection nodes for the **p-Channel MOSFET** device. These represent the *drain*, *gate*, *source*, and *bulk* nodes, respectively.

MODEL PARAMETERS

Specify the **Model Parameters**. Reasonable defaults are provided but for any particular MOSFET, the device manufacturer should be the primary source of information.

Mutual Inductance

The **Mutual Inductance** allows specifying a coupling between two existing **Inductor** features in the circuit. The mutual inductance of the coupling is

$$M = k\sqrt{L_1 L_2}$$

where k is the coupling factor and L_1 and L_2 are the inductances of the inductors.

DEVICE PARAMETERS

Enter values or expressions for the:

- **Coupling factor** k (dimensionless). The value must be between 0 and 1, and the default is 0.98.
- **First inductance** L_1 (SI unit: H) and **Second inductance** L_2 (SI unit: H). These must be set to two different **Inductor** features in the circuit.

Transformer

The **Transformer** feature represents either a combination of two **Inductor** and a **Mutual Inductance** features, or an ideal transformer.

NODE CONNECTIONS

Enter or edit the table in the **Node names** column for the **primary** and **secondary** node connections.

DEVICE PARAMETERS


Chose a **Transformer model**—**Specify inductors** (the default) or **Ideal transformer**.

For **Specify inductors** enter values or expressions for the:

- **Coupling factor** k (dimensionless). The default is 0.98.
- **First inductance** L_1 (SI unit: H). The default is 1 mH.
- **Second inductance** L_2 (SI unit: H). The default is 1 mH.

For **Ideal transformer** enter values or expressions for the **Winding ratio** N_1/N_2 (dimensionless). The default is 10.

Diode

The **Diode** device model () is a large-signal model for a diode. It is an advanced device model and no thorough description and motivation of the many input parameters are attempted here. The interested reader is referred to [Ref. 1](#) for more details on semiconductor modeling within circuits. Many device manufacturers provide model parameters for this diode model. For any particular make of diode, the device manufacturer should be the primary source of information.

DEVICE NAME

Enter a **Device name** for the diode. The prefix is D.

NODE CONNECTIONS

Specify two **Node names** for the positive and negative nodes for the **Diode** device.


MODEL PARAMETERS

Specify the **Model Parameters**. Reasonable defaults are provided but for any particular diode, the device manufacturer should be the primary source of information.



For an explanation of the **Model Parameters** see [Diode](#).

External I vs. U

The **External I vs. U** () feature connects an arbitrary voltage measurement (for example, a circuit terminal or circuit port boundary or a coil domain from another physics interface) as a voltage source between two nodes in the electrical circuit. The resulting circuit current from the first node to the second node is typically coupled back as a prescribed current source in the context of the voltage measurement.

DEVICE NAME

Enter a **Device name** for the External I vs. U node.

NODE CONNECTIONS

Specify the two **Node names** for the connecting nodes for the voltage source. The first node represents the positive reference terminal.

EXTERNAL DEVICE

Enter the source of the **Voltage**. If circuit or current excited terminals or circuit ports are defined on boundaries or domains or a multiturn coil domains is defined in other physics interfaces, these display as options in the **Voltage** list. Also select the **User defined** option and enter your own voltage variable, for example, using a suitable coupling operator. For inductive or electromagnetic wave propagation models, the voltage measurement must be performed as an integral of the electric field because the electric potential only does not capture induced EMF. Also the integration must be performed over a distance that is short compared to the local wavelength.




Except when coupling to a circuit terminal, circuit port, or coil, the current flow variable must be manually coupled back in the electrical circuit to the context of the voltage measurement. This applies also when coupling to a current excited terminal. The name of this current variable follows the convention `cirn.IvsUm_i`, where `cirn` is the tag of the Electrical Circuit interface node and `IvsUm` is the tag of the **External I vs. U** node. The tags are typically displayed within curly brackets `{}` in the Model Builder.



[Component Couplings and Coupling Operators](#) in the *COMSOL Multiphysics Reference Manual*

External U vs. I

The **External U vs. I** () feature connects an arbitrary current measurement (for example, a coil domain from another physics interface) as a current source between two nodes in the electrical circuit. The resulting circuit voltage between the first node and the second node is typically coupled back as a prescribed voltage source in the context of the current measurement.

DEVICE NAME



Enter a **Device name** for the External U vs. I node.

NODE CONNECTIONS


Specify the two **Node names** for the connecting nodes for the current source. The current flows from the first node to the second node.

EXTERNAL DEVICE

Enter the source of the **Current**. Voltage excited terminals or lumped ports defined on boundaries in other physics interfaces are natural candidates but do not appear as options in the **Voltage** list because those do not have an accurate built-in current measurement variable. A **User defined** option must be selected and a current variable entered, for example, using a suitable coupling operator.

	The voltage variable must be manually coupled back in the electrical circuit to the context of the current measurement. This applies also when coupling to a voltage excited terminal or lumped port. The name of this voltage variable follows the convention <code>cirn.UvsIm_v</code> , where <code>cirn</code> is the tag of the Electrical Circuit interface node and <code>UvsIm</code> is the tag of the External U vs. I node. The tags are typically displayed within curly brackets <code>{ }</code> in the Model Builder.
	Component Couplings and Coupling Operators in the COMSOL Multiphysics Reference Manual

External I-Terminal

The **External I-Terminal** () feature connects an arbitrary voltage-to-ground measurement (for example, a circuit terminal from another physics interface) as a voltage-to-ground assignment to a node in the electrical circuit. The resulting circuit current from the node is typically coupled back as a prescribed current source in the context of the voltage measurement. This node does not apply when coupling to inductive or electromagnetic wave propagation models because then voltage must be defined as a line integral between two points rather than a single point measurement of electric potential. For such couplings, use the **External I vs. U** node instead.

DEVICE NAME



Enter a **Device name** for the External I-terminal.

NODE CONNECTIONS


Set the **Node name** for the connecting node for the voltage assignment.


EXTERNAL TERMINAL

Enter the source of the **Voltage**. If circuit- or current-excited terminals are defined on boundaries in other physics interfaces, these display as options in the **Voltage** list. Also select the **User defined** option and enter a voltage variable, for example, using a suitable coupling operator.



	Except when coupling to a circuit terminal, the current flow variable must be manually coupled back in the electrical circuit to the context of the voltage measurement. This applies also when coupling to a current excited terminal. The name of this current variable follows the convention <code>cirn.termIm_i</code> , where <code>cirn</code> is the tag of the Electrical Circuit interface node and <code>termIm</code> is the tag of the External I-Terminal node. The tags are typically displayed within curly brackets {} in the Model Builder.
	Component Couplings and Coupling Operators in the <i>COMSOL Multiphysics Reference Manual</i>

SPICE Circuit Import

Right-click the **Electrical Circuit** () feature node to import an existing SPICE netlist (select **Import Spice Netlist**). A window opens—enter a file location or browse your directories to find one. The default file extension for a SPICE netlist is `.cir`. The SPICE circuit import translates the imported netlist into Electrical Circuit interface nodes so these define the subset of SPICE features that can be imported.

	See Spice Import about the supported SPICE commands.
---	--

SPICE Circuit Export

Right-click the **Electrical Circuit** () feature node to export the current circuit to the SPICE netlist file format (select **Export Spice Netlist** ). A window opens—enter a file location or browse your directories to find one. The default file extension for a SPICE netlist is `.cir`. The compatible circuit nodes are exported as SPICE devices



See [SPICE Export](#) for more details on the supported SPICE commands.

Theory for the Electrostatics Interface

The [Electrostatics Interface](#) is available for 3D, 2D in-plane, and 2D axisymmetric components. Applications with electrostatic equations include high-voltage apparatus, electronic devices, and capacitors. The term “statics” is not to be interpreted literally—it is the observation time, or time scale at which the applied excitation changes, that is short compared to the charge relaxation time; also, the electromagnetic wavelength and skin depth are very large compared to the size of the domain of interest.

If you do not know whether to use the Electric Currents or the Electrostatics interface, which both solve for the scalar electric potential V , consider using an explicit charge transport model. See [Charge Relaxation Theory](#).

Charge Relaxation Theory

COMSOL Multiphysics includes physics interfaces for the modeling of static electric fields and currents. Deciding what specific physics interface and study type to select for a particular modeling situation requires a basic understanding of the charge dynamics in conductors.

The different physics interfaces involving only the scalar electric potential can be interpreted in terms of the charge relaxation process. The fundamental equations involved are *Ohm's law*

$$\mathbf{J} = \sigma \mathbf{E}$$

the *equation of continuity*

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{J} = 0$$

and *Gauss' law*

$$\nabla \cdot (\epsilon \mathbf{E}) = \rho$$

By combining these, one can deduce the following differential equation for the space charge density in a homogeneous medium

$$\frac{\partial \rho}{\partial t} + \frac{\sigma}{\epsilon} \rho = 0$$

This equation has the solution

$$\rho(t) = \rho_0 e^{-t/\tau}$$

where

$$\tau = \frac{\epsilon}{\sigma}$$

is called the charge relaxation time. For a good conductor like copper, τ is of the order of 10^{-19} s, whereas for a good insulator like silica glass, it is of the order of 10^3 s. For a pure insulator, it becomes infinite.

When modeling real-world devices, there is not only the *intrinsic time scale* of the charge relaxation time but also an *external time scale* t at which a device is energized or the observation time. It is the relation between the external time scale and the charge relaxation time that determines what physics interface and study type to use. The results are summarized in [Table 3-3](#) below,

TABLE 3-3: SUITABLE PHYSICS INTERFACE AND STUDY TYPE FOR DIFFERENT TIME-SCALE REGIMES.

CASE	PHYSICS INTERFACE	STUDY TYPE
$\tau \gg t$	Electrostatics	Stationary
$\tau \ll t$	Electric Currents	Stationary
$\tau \sim t$	Electric Currents	Time Dependent or Frequency Domain

FIRST CASE: $\tau \gg t$

If the external time scale is short compared to the charge relaxation time, the charges do not have time to redistribute to any significant degree. Thus the charge distribution can be considered as a given model input. The best approach is to solve the Electrostatics formulation using the electric potential V .

By combining the definition of the potential with Gauss' law, you can derive the classical Poisson's equation. Under static conditions, the electric potential V is defined by the equivalence $\mathbf{E} = -\nabla V$. Using this together with the constitutive relation $\mathbf{D} = \epsilon_0 \mathbf{E} + \mathbf{P}$ between \mathbf{D} and \mathbf{E} , you can rewrite Gauss' law as a variant of Poisson's equation

$$-\nabla \cdot (\epsilon_0 \nabla V - \mathbf{P}) = \rho$$

This equation is used in the Electrostatics interface. It is worth noting that Gauss' law does not require the charge distribution to be static. Thus, provided dynamics are slow enough that induced electric fields can be neglected and hence a scalar electric potential is justified, the formulation can be used also in the Time Dependent study type. That typically involves either prescribing the charge dynamics or coupling a separate formulation for this.



Such separate charge transport formulations can be found in the Plasma Module, the Semiconductor Module, and the Chemical Reaction Engineering Module.

SECOND CASE: $\tau < T$

If the external time scale is long compared to the charge relaxation time, the stationary solution to the equation of continuity has been reached. In a stationary coordinate system, a slightly more general form of Ohm's law than above states that

$$\mathbf{J} = \sigma \mathbf{E} + \mathbf{J}_e$$

where \mathbf{J}_e is an externally generated current density. The static form of the equation of continuity then reads

$$\nabla \cdot \mathbf{J} = -\nabla \cdot (\sigma \nabla V - \mathbf{J}_e) = 0$$

To handle current sources, the equation can be generalized to

$$-\nabla \cdot (\sigma \nabla V - \mathbf{J}_e) = Q_j$$

This equation is used in the static study type for the Electric Currents interface.

Electrostatics Equations

Under static conditions, the electric potential, V , is defined by the relationship:

$$\mathbf{E} = -\nabla V$$

Combining this equation with the constitutive relationship $\mathbf{D} = \epsilon_0 \mathbf{E} + \mathbf{P}$ between the electric displacement \mathbf{D} and the electric field \mathbf{E} , it is possible to represent Gauss' law as the following equation:

$$-\nabla \cdot (\epsilon_0 \nabla V - \mathbf{P}) = \rho$$

In this equation, the physical constant, ϵ_0 (SI unit: F/m) is the permittivity of vacuum, \mathbf{P} (SI unit: C/m²) is the electric polarization vector, and ρ (SI unit: C/m³) is a space charge density. This equation describes the electrostatic field in dielectric materials.

For in-plane 2D modeling, the Electrostatics interface assumes a symmetry where the electric potential varies only in the x and y directions and is constant in the z direction. This implies that the electric field, \mathbf{E} , is tangential to the xy -plane. With this symmetry, the same equation is solved as in the 3D case. The physics interface solves the following equation where d is the thickness in the z direction:

$$-\nabla \cdot d(\epsilon_0 \nabla V - \mathbf{P}) = \rho$$

The axisymmetric version of the physics interface considers the situation where the fields and geometry are axially symmetric. In this case, the electric potential is constant in the ϕ direction, which implies that the electric field is tangential to the rz -plane.

The Electrostatics Interface in Time Dependent or Frequency Domain Studies

[The Electrostatics Interface](#) can also be solved in a dynamic study (Time Dependent or Frequency Domain). The equation system solved, however, is always the one presented in the previous section for the stationary case, in which no transient electromagnetic effects are taken into account. The difference is that the sources of the problem (charge densities, electric potential) are assumed to be time-varying (in a Time Dependent study) or time-harmonic (in a Frequency Domain study). The support for dynamic studies simplifies the coupling of the Electrostatics interface with other physics interfaces. Using the physics interface in a dynamic study is a valid approximation only if the time-scale (or the frequency) of the study is so slow that transient electromagnetic effects can be neglected; for example, in acoustic or structural problems.

Most of the features in the Electrostatics interface have the same behavior in all study types, with the important exception of the Terminal node.

The Electrostatics interface also supports the small-signal analysis study sequence, which can be used when a time-harmonic perturbation is superposed on a static bias charge or voltage.

Theory for the Electric Currents Interface

The [Electric Currents Interface](#) solves a current conservation problem for the scalar electric potential V and is available for 3D, 2D in-plane, and 2D axisymmetric components. Electrolysis and the computation of resistances of grounding plates are examples that involve conductive media with electrical conductivities and electric currents. If you are uncertain of whether to use the Electric Currents or the Electrostatics interface, which both solve for the scalar electric potential V , refer to the section on .

In this section:

- [Electric Currents Equations in Steady State](#)
- [Dynamic Electric Currents Equations](#)

Electric Currents Equations in Steady State

When handling stationary electric currents in conductive media you must consider the stationary equation of continuity. In a stationary coordinate system, the point form of *Ohm's law* states that:

$$\mathbf{J} = \sigma \mathbf{E} + \mathbf{J}_e$$

where σ is the electrical conductivity (SI unit: S/m), and \mathbf{J}_e is an externally generated current density (SI unit: A/m²). The static form of the equation of continuity then states:

$$\nabla \cdot \mathbf{J} = -\nabla \cdot (\sigma \nabla V - \mathbf{J}_e) = 0$$

To handle current sources, you can generalize the equation to:

$$-\nabla \cdot (\sigma \nabla V - \mathbf{J}_e) = Q_j$$

In planar 2D the Electric Currents interface assumes that the model has a symmetry where the electric potential varies only in the x and y directions and is constant in the z direction. This implies that the electric field, \mathbf{E} , is tangential to the xy -plane. The Electric Currents interface then solves the following equation, where d is the thickness in the z direction:

$$-\nabla \cdot d(\sigma \nabla V - \mathbf{J}_e) = dQ_j \quad (3-1)$$

In 2D axisymmetry, the Electric Currents interface considers the situation where the fields and geometry are axially symmetric. In this case, the electric potential is constant in the ϕ direction, which implies that the electric field is tangential to the rz -plane.

Dynamic Electric Currents Equations

In the Frequency Domain and Time Dependent study types dynamic formulations accounting for both conduction currents and displacement currents are used.

Combining the time-harmonic equation of continuity

$$\nabla \cdot \mathbf{J} = \nabla \cdot (\sigma \mathbf{E} + \mathbf{J}_e) = -j\omega \rho$$

with the equation

$$\nabla \cdot \mathbf{D} = \rho$$

and generalized to handle current sources yields the following equation:

$$-\nabla \cdot ((\sigma + j\omega\epsilon_0)\nabla V - (\mathbf{J}_e + j\omega\mathbf{P})) = Q_j$$

For the transient case, using the transient equation of continuity

$$\nabla \cdot \mathbf{J} = \nabla \cdot (\sigma \mathbf{E} + \mathbf{J}_e) = \frac{\partial \rho}{\partial t}$$

and generalized to handle current sources the resulting equation becomes:

$$-\nabla \cdot \frac{\partial}{\partial t}(\epsilon_0 \nabla V - \mathbf{P}) - \nabla \cdot (\sigma \nabla V - \mathbf{J}_e) = Q_j \quad (3-2)$$

In planar 2D the dynamic formulations also involves the thickness d in the z direction:

$$-\nabla \cdot d((\sigma + j\omega\epsilon_0)\nabla V - (\mathbf{J}_e + j\omega\mathbf{P})) = dQ_j$$

$$-\nabla \cdot d\frac{\partial}{\partial t}(\epsilon_0 \nabla V - \mathbf{P}) - \nabla \cdot d(\sigma \nabla V - \mathbf{J}_e) = dQ_j$$

Theory for the Electrical Circuit Interface

The [Electrical Circuit Interface](#) theory is discussed in this section:

- [Electric Circuit Modeling and the Semiconductor Device Models](#)
- [Bipolar Transistors](#)
- [MOSFET Transistors](#)
- [Diode](#)
- [Reference for the Electrical Circuit Interface](#)



Connecting to Electrical Circuits

Electric Circuit Modeling and the Semiconductor Device Models

Electrical circuit modeling capabilities are useful when simulating all sorts of electrical and electromechanical devices ranging from heaters and motors to advanced plasma reactors in the semiconductor industry. There are two fundamental ways that an electrical circuit model relates to a physical field model.

- The field model is used to get a better, more accurate description of a single device in the electrical circuit model.
- The electrical circuit is used to drive or terminate the device in the field model in such a way that it makes more sense to simulate both as a tightly coupled system.

The Electrical Circuit interface makes it possible to add nodes representing circuit elements directly to the Model Builder tree in a COMSOL Multiphysics model. The circuit variables can then be connected to a physical device model to perform co-simulations of circuits and multiphysics. The model acts as a device connected to the circuit so that its behavior is analyzed in larger systems.

The fundamental equations solved by the Electrical Circuit interface are Kirchhoff's circuit laws, which in turn can be deduced from Maxwell's equations. The supported study types are Stationary, Frequency Domain, and Time Dependent.

There are three more advanced large-signal semiconductor device features available in the Electrical Circuit interface. The equivalent circuits and the equations defining their non-ideal circuit elements are described in this section. For a more detailed account on semiconductor device modeling, see [Ref. 1](#).

Bipolar Transistors

[Figure 3-1](#) illustrates the equivalent circuit for the npn bipolar junction transistor.

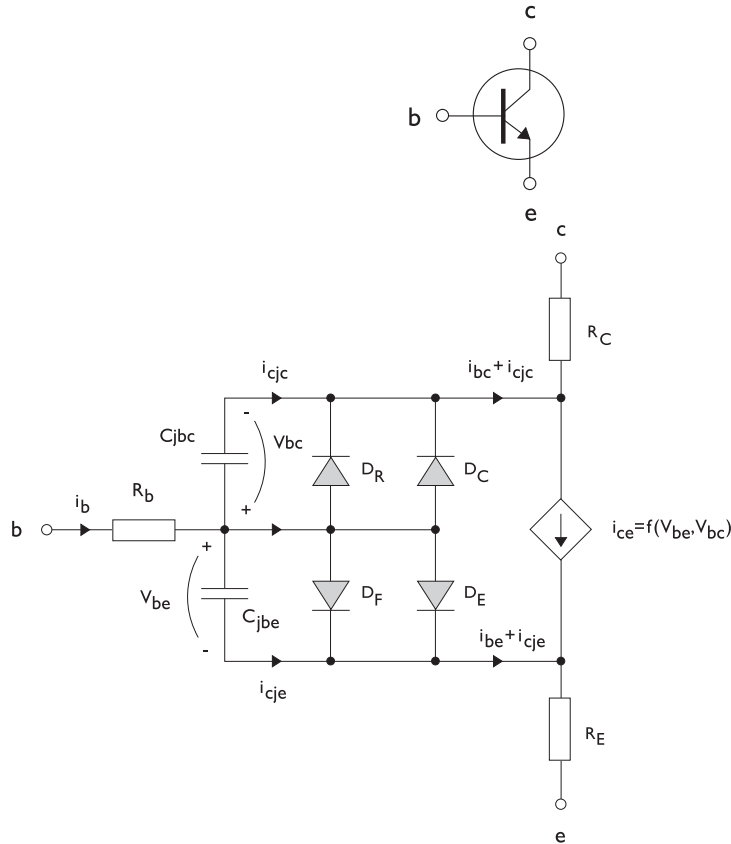


Figure 3-1: A circuit for the bipolar transistor.

The pnp transistor model is similar in all regards to the npn transistor, with the difference that the polarities of the currents and voltages involved are reversed. The following equations are used to compute the relations between currents and voltages in the circuit.

$$v_{rb} = \frac{1}{A} \left(R_{BM} - \frac{R_B - R_{BM}}{f_{bq}} \right) i_b$$

$$f_{bq} = \frac{1}{2 \left(1 - \frac{v_{bc}}{V_{AF}} - \frac{v_{be}}{V_{AR}} \right)} \left(1 + \sqrt{1 + 4 I_S \left(e^{\frac{v_{be}}{N_F V_T} - 1} \frac{1}{I_{KF} A} + e^{\frac{v_{bc}}{N_R V_T} - 1} \frac{1}{I_{KR} A} \right)} \right)$$

$$i_{be} = A \left(\frac{I_S}{B_F} \left(e^{\frac{v_{be}}{N_F V_T} - 1} \right) + I_{SE} \left(e^{\frac{v_{be}}{N_E V_T} - 1} \right) \right)$$

$$i_{bc} = A \left(\frac{I_S}{B_R} \left(e^{\frac{v_{bc}}{N_R V_T} - 1} \right) + I_{SC} \left(e^{\frac{v_{bc}}{N_C V_T} - 1} \right) \right)$$

$$i_{ce} = A \left(\frac{I_S}{f_{bq}} \left(e^{\frac{v_{be}}{N_F V_T}} + e^{\frac{v_{bc}}{N_C V_T}} \right) \right)$$

$$V_T = \frac{k_B T_{NOM}}{q}$$

There are also two capacitances that use the same formula as the junction capacitance of the diode model. In the parameter names below, replace x with C for the base-collector capacitance and E for the base-emitter capacitance.

$$C_{jbx} = AC_{Jx} \times \begin{cases} \left(1 - \frac{v_{bx}}{V_{Jx}} \right)^{-M_{Jx}} & v_{bx} < F_C V_{Jx} \\ (1 - F_C)^{-1 - M_{Jx}} \left(1 - F_C (1 + M_{Jx}) + M_{Jx} \frac{v_{bx}}{V_{Jx}} \right) & v_{bx} \geq F_C V_{Jx} \end{cases}$$

The model parameters are listed in the table below.

TABLE 3-1: BIPOLAR TRANSISTOR MODEL PARAMETERS

PARAMETER	DEFAULT	DESCRIPTION
B_F	100	Ideal forward current gain
B_R	1	Ideal reverse current gain
C_{JC}	0 F/m ²	Base-collector zero-bias depletion capacitance
C_{JE}	0 F/m ²	Base-emitter zero-bias depletion capacitance
F_C	0.5	Breakdown current
I_{KF}	Inf (A/m ²)	Corner for forward high-current roll-off
I_{KR}	Inf (A/m ²)	Corner for reverse high-current roll-off

TABLE 3-1: BIPOLAR TRANSISTOR MODEL PARAMETERS

PARAMETER	DEFAULT	DESCRIPTION
I_S	$1\text{e-}15\text{ A/m}^2$	Saturation current
I_{SC}	0 A/m^2	Base-collector leakage saturation current
I_{SE}	0 A/m^2	Base-emitter leakage saturation current
M_{JC}	1/3	Base-collector grading coefficient
M_{JE}	1/3	Base-emitter grading coefficient
N_C	2	Base-collector ideality factor
N_E	1.4	Base-emitter ideality factor
N_F	1	Forward ideality factor
N_R	1	Reverse ideality factor
R_B	$0\text{ }\Omega\text{m}^2$	Base resistance
R_{BM}	$0\text{ }\Omega\text{m}^2$	Minimum base resistance
R_C	$0\text{ }\Omega\text{m}^2$	Collector resistance
R_E	$0\text{ }\Omega\text{m}^2$	Emitter resistance
T_{NOM}	298.15 K	Device temperature
V_{AF}	Inf (V)	Forward Early voltage
V_{AR}	Inf (V)	Reverse Early voltage
V_{JC}	0.71 V	Base-collector built-in potential
V_{JE}	0.71 V	Base-emitter built-in potential

Figure 3-2 illustrates an equivalent circuit for the n-channel MOSFET transistor. The p-channel MOSFET transistor is treated similarly, but the polarities of the involved voltages are reversed.

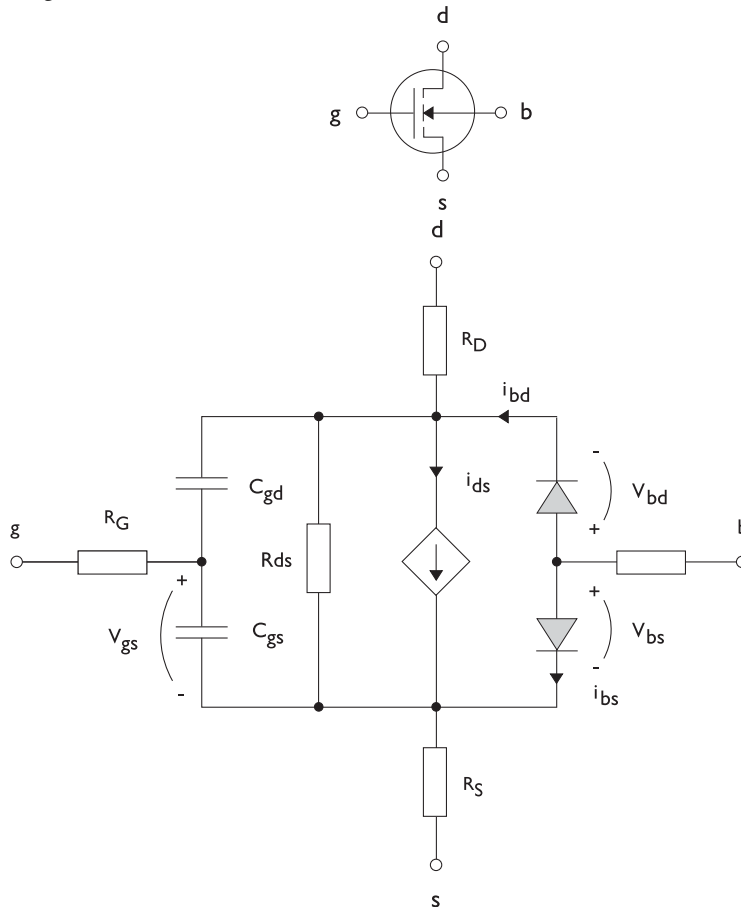


Figure 3-2: A circuit for the MOSFET transistor.

The following equations are used to compute the relations between currents and voltages in the circuit.

$$i_{ds} = \begin{cases} \frac{WK_P}{L} \frac{1}{2} (1 + \Lambda v_{ds}) v_{ds} (2v_{th} - v_{ds}) & v_{ds} < v_{th} \\ \frac{WK_P}{L} \frac{1}{2} (1 + \Lambda v_{ds}) v_{th}^2 & v_{ds} \geq v_{th} \\ 0 & v_{ds} < v_{th} \leq 0 \end{cases}$$

$$v_{th} = v_{gs} - (V_{TO} + \Gamma(\sqrt{\Phi} - v_{bs} - \sqrt{\Phi}))$$

$$i_{bd} = I_S \left(e^{\frac{v_{bd}}{NV_T}} - 1 \right)$$

$$i_{bs} = I_S \left(e^{\frac{v_{bs}}{NV_T}} - 1 \right)$$

$$V_T = \frac{k_B T_{NOM}}{q}$$

There are also several capacitances between the terminals

$$\begin{aligned} C_{gd} &= C_{gd0} W \\ C_{gs} &= C_{gs0} W \\ C_{jbd} &= C_{BD} \times \begin{cases} \left(1 - \frac{v_{bd}}{P_B}\right)^{-M_J} & v_{bx} < F_C P_B \\ (1 - F_C)^{-1-M_J} \left(1 - F_C(1 + M_J) + M_J \frac{v_{bx}}{P_B}\right) & v_{bx} \geq F_C P_B \end{cases} \end{aligned}$$

The model parameters are as follows:

TABLE 3-2: MOSFET TRANSISTOR MODEL PARAMETERS

PARAMETER	DEFAULT	DESCRIPTION
C_{BD}	0 F/m	Bulk-drain zero-bias capacitance
C_{GDO}	0 F/m	Gate-drain overlap capacitance
C_{GSO}	0 F/m	Gate-source overlap capacitance
F_C	0.5	Capacitance factor
I_S	1e-13 A	Bulk junction saturation current
K_P	2e-5 A/V ²	Transconductance parameter
L	50e-6 m	Gate length
M_J	0.5	Bulk junction grading coefficient
N	1	Bulk junction ideality factor

TABLE 3-2: MOSFET TRANSISTOR MODEL PARAMETERS

PARAMETER	DEFAULT	DESCRIPTION
P_B	0.75 V	Bulk junction potential
R_B	0 Ω	Bulk resistance
R_D	0 Ω	Drain resistance
R_{DS}	Inf (Ω)	Drain-source resistance
R_G	0 Ω	Gate resistance
R_S	0 Ω	Source resistance
T_{NOM}	298.15 K	Device temperature
V_{TO}	0 V	Zero-bias threshold voltage
W	50e-6 m	Gate width
Γ (GAMMA)	0 $\sqrt{0.5}$	Bulk threshold parameter
Φ (PHI)	0.5 V	Surface potential
Λ (LAMBDA)	0 1/V	Channel-length modulation

Diode

Figure 3-3 illustrates equivalent circuit for the diode.

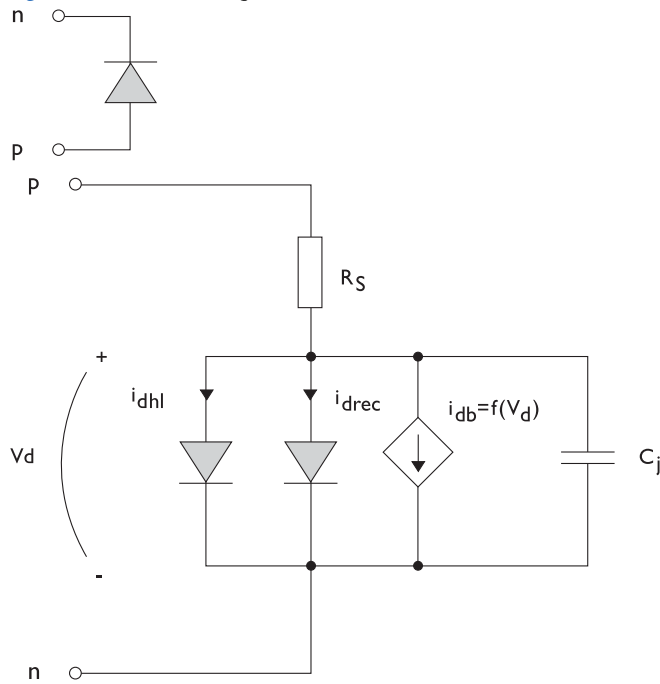


Figure 3-3: A circuit for the diode.

The following equations are used to compute the relations between currents and voltages in the circuit.

$$\begin{aligned}
i_d &= i_{dhl} + i_{drec} + i_{db} + i_c \\
i_{dhl} &= I_S \left(e^{\frac{v_d}{N V_T}} - 1 \right) \frac{1}{\sqrt{1 + \frac{I_S}{I_{KF}} \left(e^{\frac{v_d}{N V_T}} - 1 \right)}} \\
i_{drec} &= I_{SR} \left(e^{\frac{v_d}{N_R V_T}} - 1 \right) \\
i_{db} &= I_{BV} e^{\frac{v_d + B_V}{N_{BV} V_T}} \\
C_j &= C_{J0} \times \begin{cases} \left(1 - \frac{v_d}{V_J} \right)^{-M} & v_d < F_C V_J \\ (1 - F_C)^{-1-M} \left(1 - F_C (1 + M) + M \frac{v_d}{V_J} \right) & v_d \geq F_C V_J \end{cases} \\
V_T &= \frac{k_B T_{NOM}}{q}
\end{aligned}$$

where the following model parameters are required

TABLE 3-3: DIODE TRANSISTOR MODEL PARAMETERS

PARAMETER	DEFAULT	DESCRIPTION
B_V	Inf (V)	Reverse breakdown voltage
C_{J0}	0 F	Zero-bias junction capacitance
F_C	0.5	Forward-bias capacitance coefficient
I_{BV}	1e-09 A	Current at breakdown voltage
I_{KF}	Inf (A)	Corner for high-current roll-off
I_S	1e-13 A	Saturation current
M	0.5	Grading coefficient
N	1	Ideality factor
N_{BV}	1	Breakdown ideality factor
N_R	2	Recombination ideality factor
R_S	0 Ω	Series resistance
T_{NOM}	298.15 K	Device temperature
V_J	1.0 V	Junction potential

Reference for the Electrical Circuit Interface

1. P. Antognetti and G. Massobrio, *Semiconductor Device Modeling with Spice*, 2nd ed., McGraw-Hill, 1993.

Connecting to Electrical Circuits

In this section:

- [About Connecting Electrical Circuits to Physics Interfaces](#)
- [Connecting Electrical Circuits Using Predefined Couplings](#)
- [Connecting Electrical Circuits by User-Defined Couplings](#)
- [Solving](#)
- [Postprocessing](#)

About Connecting Electrical Circuits to Physics Interfaces

This section describes the various ways electrical circuits can be connected to other physics interfaces in COMSOL Multiphysics. If you are not familiar with circuit modeling, it is recommended that you review the [Theory for the Electrical Circuit Interface](#).

In general electrical circuits connect to other physics interfaces via one or more of three special circuit features:

- [External I vs. U](#)
- [External U vs. I](#)
- [External I-Terminal](#)

These features either accept a voltage measurement from the connecting non-circuit physics interface and return a current from the Electrical Circuit interface or the other way around.



The “External” features are considered “ideal” current or voltage sources by the Electrical Circuit interface. Hence, you cannot connect them directly in parallel (voltage sources) or in series (current sources) with other ideal sources. This results in the error message *The DAE is structurally inconsistent*. A workaround is to provide a suitable parallel or series resistor, which can be tuned to minimize its influence on the results.

Connecting Electrical Circuits Using Predefined Couplings

In addition to these circuit features, interfaces in the AC/DC Module, RF Module, MEMS Module, Plasma Module, and Semiconductor Module (the modules that include the Electrical Circuit interface) also contain features that provide couplings to the Electrical Circuit interface by accepting a voltage or a current from one of the specific circuit features ([External I vs. U](#), [External U vs. I](#), and [External I-Terminal](#)).

This coupling is typically activated when:

- A choice is made in the **Settings** window for the non-circuit physics interface feature, which then announces (that is, includes) the coupling to the Electrical Circuit interface. Its voltage or current is then included to make it visible to the connecting circuit feature.
- A voltage or current that has been announced (that is, included) is selected in a feature node's **Settings** window.

These circuit connections are supported in Terminals.

Connecting Electrical Circuits by User-Defined Couplings

A more general way to connect a physics interface to the Electrical Circuit interface is to:

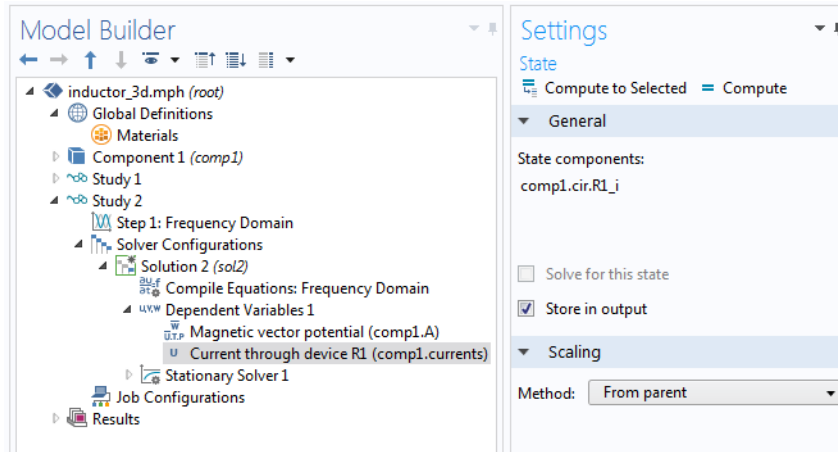
- Apply the voltage or current from the connecting “External” circuit feature as an excitation in the non-circuit physics interface.
- Define your own voltage or current measurement in the non-circuit physics interface using variables, coupling operators and so forth.
- In the **Settings** window for the Electrical Circuit interface feature, selecting the User-defined option and entering the name of the variable or expression using coupling operators defined in the previous step.

DETERMINING A CURRENT OR VOLTAGE VARIABLE NAME

To determine a current or voltage variable name, look at the **Dependent Variables** node under the **Study** node. To do this:

- I In the **Model Builder**, right-click the **Study** node and select **Show Default Solver**.

- 2 Expand the **Solver>Dependent Variables** node and click the state node, in this example, **Current through device R1 (comp1.currents)**. The variable name is shown on the **Settings** window for **State**



Typically, voltage variables are named `cir.Xn_v` and current variables `cir.Xn_i`, where n is the “External” device number—1, 2, and so on.

Solving



Some modeling errors lead to the error message *The DAE is structurally inconsistent*, being displayed when solving. This typically occurs from having an open current loop, from connecting voltage sources in parallel, or connecting current sources in series.

In this respect, the predefined coupling features are also treated as (ideal) voltage or current sources. The remedy is to close current loops and to connect resistors in series with voltage sources or in parallel with current sources.

Postprocessing

The Electrical Circuits interface, unlike most of the other physics interfaces, solves for a relatively large number of Global dependent variables (such as voltages and currents), instead of solving for a few space-varying fields (such as temperature or displacement). For this reason, the Electrical Circuit interface does not provide default plots when computing a Study.

The physics interface defines a number of variables that can be used in postprocessing. All variables defined by the Electrical Circuit interface are of a global scope, and can be evaluated in a **Global Evaluation** node (under **Derived Values**). In addition, the time evolution or dependency on a parameter can be plotted in a **Global** plot (under a **ID Plot Group** node).

The physics interface defines a Node voltage variable for each electrical node in the circuit, with name `cir.v_name`, where `cir` is the physics interface Label and `<name>` is the node Name. For each two pin component, the physics interface also defines variables containing the voltage across it and the current flowing through it.



In the *COMSOL Multiphysics Reference Manual*:

- [Derived Values and Tables](#) and [Global Evaluation](#)
 - [Plot Groups and Plots](#) and [Global](#)
-

Spice Import

SPICE Import

The circuit definition in COMSOL Multiphysics adheres to the SPICE format developed at the University of California, Berkeley ([Ref. 1](#)). SPICE netlists can be imported and the corresponding circuit nodes are generated in the COMSOL Multiphysics model. Most circuit simulators can export to this format or some version of it.

The Electrical Circuit interface supports the following device models:

TABLE 3-4: SUPPORTED SPICE DEVICE MODELS

STATEMENT	DEVICE MODEL
R	Resistor
C	Capacitor
L	Inductor
V	Voltage Source
I	Current Source
E	Voltage-Controlled Voltage Source
F	Current-Controlled Current Source
G	Voltage-Controlled Current Source
H	Current-Controlled Voltage Source
D	Diode
Q	NPN BJT
M	n-Channel MOSFET
X	Subcircuit Instance

Statements corresponding to multiple devices are resolved by parsing the associated `.model` statement. The physics interface also supports the `.subckt` statement, which is represented in COMSOL by a [Subcircuit Definition](#) node, and the `.include` statement. SPICE commands are interpreted case-insensitively. The statement defining each device is also interpreted as the **Device name**.

According to SPICE specification, the first line in the netlist file is assumed to be the title of the netlist and it is ignored by the parser.

SPICE Export

The **SPICE Export** functionality creates a SPICE netlist file containing a description of the circuit represented by the physics interface. This functionality can be accessed from the physics interface context menu (right-click the physics interface node and select **Export SPICE Netlist**). After specifying a file name, the circuit is exported and messages from the export process display in the Messages window. During the export process, a series of operations are performed:


- In order to avoid conflicts, each component must be identified by a unique **Device name**. If one or more components have the same device name, the export operation fails and an error message is displayed. All characters in a **Device name** that are not letters, digits or underscores are replaced by underscores.
- According to the SPICE specification, each circuit must have a node with name 0, which is assumed to be the only ground node. When exporting a circuit, any node with name 0 that is not connected to a **Ground** component is exported with a different node name. All nodes that are connected to a Ground components are exported as a merged node with name 0. The Messages window shows a log message if these operations are performed, showing the name of the renamed or merged nodes.
- All characters in node names that are not letters, digits or underscores are replaced by underscores.
- Some components (most notably, the **External** components used to couple to other physics interfaces) cannot be exported to a SPICE netlist. These components are ignored during the export process, and a message is shown in the Messages window. Note that this can change the exported circuit, since some components are then missing.
- Subcircuit definitions are added as `.subckt` statements in the netlist. Semiconductor devices (such as MOSFETs, BJTs, and diodes) are exported as a SPICE device with a corresponding `.model` statement.

The title of the exported netlist file is the model's file name, and the time, date and the version of COMSOL Multiphysics is added as a comment in the netlist file.

Reference for SPICE Import and Export

1. <http://bwrc.eecs.berkeley.edu/Classes/IcBook/SPICE/>

Fluid Flow Interface

This chapter has information about the enhanced Laminar Flow interface found under the **Fluid Flow>Single-Phase Flow** branch () when adding a physics interface.

In this chapter:

- [The Single-Phase Flow, Laminar Flow Interface](#)
- [Theory for the Single-Phase Flow Laminar Interface](#)


The Single-Phase Flow, Laminar Flow Interface

In this section:


- [The Laminar Flow Interface](#)



- [Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow](#)
 - [Theory for the Single-Phase Flow Laminar Interface](#)
-

In the *COMSOL Multiphysics Reference Manual*, see [Table 2-3](#) for links to common sections such as **Discretization**, **Consistent Stabilization**, **Inconsistent Stabilization**, and **Advanced Settings** sections, all accessed by clicking the **Show** button () and choosing the applicable option. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

The Laminar Flow Interface

The **Laminar Flow (spf)** interface () is used to compute the velocity and pressure fields for the flow of a single-phase fluid in the laminar flow regime. A flow remains laminar as long as the Reynolds number is below a certain critical value. At higher Reynolds numbers, disturbances have a tendency to grow and cause transition to turbulence. This critical Reynolds number depends on the model, but a classical example is pipe flow where the critical Reynolds number is known to be approximately 2000.

The physics interface supports incompressible flow, weakly compressible flow (the density depends on temperature but not pressure) and compressible flow at low Mach numbers (typically less than 0.3). It also supports flow of non-Newtonian fluids.

The equations solved by the Laminar Flow interface are the Navier-Stokes equations for conservation of momentum and the continuity equation for conservation of mass.

The Laminar Flow interface can be used for stationary and time-dependent analyses. Time-dependent studies should be used in the high-Reynolds number regime as these flows tend to become inherently unsteady.

When the Laminar Flow interface is added, the following default nodes are also added in the **Model Builder: Fluid Properties**, **Wall** (the default boundary condition is **No slip**), and **Initial Values**. Other nodes, that implement, for example, boundary conditions and volume forces, can be added from the **Physics** toolbar or from the context menu displayed when right-clicking **Laminar Flow**.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Physics interface variables can be referred to using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the name string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `spf`.

PHYSICAL MODEL

Compressibility

Depending of the fluid properties and the flow regime, three options are available for the **Compressibility** option. In general the computational complexity increases from **Incompressible flow** to **Weakly compressible flow** to **Compressible flow (Ma<0.3)** but the underlying hypotheses are increasingly more restrictive in the opposite direction.

When the **Incompressible flow** option (default) is selected, the incompressible form of the Navier-Stokes and continuity equations is applied. In addition, the fluid density is evaluated at the **Reference pressure level** and at the **Reference temperature** defined in **Reference values**. The fluid dynamic viscosity is evaluated at the **Reference temperature**.

The **Weakly compressible flow** option models compressible flow when the pressure dependency of the density can be neglected. When selected, the compressible form of the Navier-Stokes and continuity equations is applied. In addition, the fluid density is evaluated at the **Reference pressure level** defined in **Reference values**.

When the **Compressible flow (Ma<0.3)** option is selected, the compressible form of the Navier-Stokes and continuity equations is applied. $Ma < 0.3$ indicates that the inlet and outlet conditions, as well as the stabilization, may not be suitable for transonic and supersonic flow. For more information, see [The Mach Number Limit](#).

Porous Media Domains

With the addition of various modules, the **Enable porous media domains** check box is available. Selecting this option, a **Fluid and Matrix Properties** node, a **Mass Source** node, and a **Forchheimer Drag** subnode are added to the physics interface. These are described for the **Brinkman Equations** interface in the respective module's documentation. The **Fluid and Matrix Properties** can be applied on all domains or on a subset of the domains.

Reference values

Reference values are global quantities used to evaluate the density and viscosity of the fluid when the **Incompressible flow** or the **Weakly compressible flow** option is selected.

Reference pressure level There are generally two ways to include the pressure in fluid flow computations: either to use the absolute pressure $p_A = p + p_{\text{ref}}$, or the gauge pressure p . When p_{ref} is nonzero, the physics interface solves for the gauge pressure whereas material properties are evaluated using the absolute pressure. The reference pressure level is also .

Reference temperature The reference temperature is .


Reference position When **Include gravity** is selected, the reference position can be defined. It corresponds to the location where the total pressure (that includes the hydrostatic pressure) is equal to the **Reference pressure level**.

DEPENDENT VARIABLES



The following dependent variables (fields) are defined for this physics interface—the **Velocity field \mathbf{u}** and its components, and the **Pressure p** .

If required, the names of the field, component, and dependent variable may be edited. Editing the name of a scalar dependent variable changes both its field name and the dependent variable name. If a new field name coincides with the name of another field of the same type, the fields share degrees of freedom and dependent variable names. A new field name must not coincide with the name of a field of another type or with a component name belonging to some other field. Component names must be unique within a model except when two fields share a common field name.

ADVANCED SETTINGS

To display this section, click the **Show** button () and select **Advanced Physics Options**. Normally these settings do not need to be changed.

The **Use pseudo time stepping for stationary equation form** option adds pseudo time derivatives to the equation when the **Stationary equation** form is used in order to speed up convergence. When selected, a **CFL number expression** should also be defined. For the default **Automatic** option, the local CFL number (from the Courant–Friedrichs–Lewy condition) is determined by a PID regulator.


	<ul style="list-style-type: none"> • Pseudo Time Stepping for Laminar Flow Models • Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow • Theory for the Single-Phase Flow Laminar Interface
	<i>Flow Past a Cylinder:</i> Application Library path COMSOL_Multiphysics/Fluid_Dynamics/cylinder_flow

Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow

The following nodes, listed in alphabetical order, are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or by right-clicking to access the context menu (all users).

- | | |
|--|---|
| <ul style="list-style-type: none"> • Boundary Stress • Flow Continuity • Fluid Properties • Initial Values • Inlet • Line Mass Source¹ • Open Boundary • Outlet | <ul style="list-style-type: none"> • Periodic Flow Condition • Pipe Connection¹ • Point Mass Source¹ • Pressure Point Constraint • Symmetry • Vacuum Pump • Volume Force • Wall |
|--|---|

¹ A feature that may require an additional license

	For 2D axisymmetric components, COMSOL Multiphysics takes the axial symmetry boundaries (at $r = 0$) into account and adds an Axial Symmetry node that is valid on the axial symmetry boundaries only.
---	--

In the *COMSOL Multiphysics Reference Manual*, see [Table 2-3](#) for links to common sections and [Table 2-4](#) to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

Fluid Properties

The **Fluid Properties** node adds the momentum and continuity equations solved by the physics interface, except for volume forces which are added by the [Volume Force](#) feature. The node also provides an interface for defining the material properties of the fluid.

MODEL INPUTS

Fluid properties, such as density and viscosity, can be defined through user inputs, variables, or by selecting a material. For the latter option, additional inputs, for example temperature and/or pressure, may be required to define these properties.

Temperature

By default, the single-phase flow interfaces are set to model isothermal flow. If a Heat Transfer interface is included in the component, the temperature field may alternatively be selected from this physics interface. All physics interfaces have their own tags (**Name**). For example, if a Heat Transfer in Fluids interface is included in the component, the **Temperature (ht)** option is available for T .


Absolute Pressure

This input appears when a material requires the absolute pressure as a model input. The absolute pressure is used to evaluate material properties, but it also relates to the value of the calculated pressure field. There are generally two ways to calculate the pressure when describing fluid flow: either to solve for the absolute pressure or for a pressure (often denoted gauge pressure) that relates to the absolute pressure through a reference pressure.

The choice of pressure variable depends on the system of equations being solved. For example, in a unidirectional incompressible flow problem, the pressure drop over the modeled domain is probably many orders of magnitude smaller than the atmospheric pressure, which, when included, may reduce the stability and convergence properties of the solver. In other cases, such as when the pressure is part of an expression for the gas volume or the diffusion coefficients, it may be more convenient to solve for the absolute pressure.

The default **Absolute pressure** p_A is $p + p_{\text{ref}}$, where p is the dependent pressure variable from the Navier-Stokes or RANS equations, and p_{ref} is from the user input defined at

the physics interface level. When p_{ref} is nonzero, the physics interface solves for a gauge pressure. If the pressure field instead is an absolute pressure field, p_{ref} should be set to 0.

The **Absolute pressure** field can be edited by clicking **Make All Model Inputs Editable** () and entering the desired value in the input field.

FLUID PROPERTIES

Density

If density variations with respect to pressure are to be included in the computations, the flow must be set to compressible (at the physics interface level).

Dynamic Viscosity

The **Dynamic viscosity** μ describes the relationship between the shear rate and the shear stresses in a fluid. Intuitively, water and air have low viscosities, and substances often described as thick (such as oil) have higher viscosities.

Using the built-in variable for the shear rate magnitude, $\text{spf} \cdot \text{sr}$, makes it possible to define arbitrary expressions of the dynamic viscosity as a function of the shear rate.

The non-Newtonian fluids models have a shear-rate dependent viscosity. Examples of non-Newtonian fluids include yogurt, paper pulp, and polymer suspensions. See [Non-Newtonian Flow: The Power Law and the Carreau Model](#).

Volume Force

The **Volume Force** node specifies the volume force \mathbf{F} on the right-hand side of the momentum equation.

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot \left[-p \mathbf{I} + \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu(\nabla \cdot \mathbf{u}) \mathbf{I} \right] + \mathbf{F}$$

If several volume-force nodes are added to the same domain, then the sum of all contributions are added to the momentum equation.

Initial Values

The initial values serve as initial conditions for a transient simulation or as an initial guess for a nonlinear solver in a stationary simulation. Note that for a transient compressible-flow simulation employing a material for which the density depends on the pressure (such as air), discontinuities in the initial values trigger pressure waves

even when the Mach number is small. The pressure waves must be resolved and this puts a restriction on the time step.

INITIAL VALUES

Initial values or expressions should be specified for the **Velocity field \mathbf{u}** and the **Pressure p** .

Wall

The **Wall** node includes a set of boundary conditions describing fluid-flow conditions at stationary, moving, and leaking walls.

BOUNDARY CONDITION

Select a **Boundary condition** for the wall.

- No Slip¹
- Slip
- Sliding Wall
- Moving Wall
- Leaking Wall
- Slip Velocity

¹ The default

No Slip

No slip is the default boundary condition for a stationary solid wall for laminar flow. The condition prescribes $\mathbf{u} = 0$; that is, the fluid at the wall is not moving.

Slip

The **Slip** option prescribes a no-penetration condition, $\mathbf{u} \cdot \mathbf{n} = 0$. It is implicitly assumed that there are no viscous effects at the slip wall and hence, no boundary layer develops. From a modeling point of view, this can be a reasonable approximation if the main effect of the wall is to prevent fluid from leaving the domain.

Sliding Wall

The **Sliding wall** boundary condition is appropriate if the wall behaves like a conveyor belt; that is, the surface is sliding in its tangential direction. A velocity is prescribed at the wall and the boundary itself does not have to actually move relative to the reference frame.

- For 3D components, values or expressions for the **Velocity of sliding wall \mathbf{u}_w** should be specified. If the velocity vector entered is not in the plane of the wall, COMSOL

Multiphysics projects it onto the tangential direction. Its magnitude is adjusted to be the same as the magnitude of the vector entered.

- For 2D components, the tangential direction is unambiguously defined by the direction of the boundary. For this reason, the sliding wall boundary condition has different definitions in different space dimensions. A single entry for the **Velocity of the tangentially moving wall** U_w should be specified in 2D.

Moving Wall

For an arbitrary wall movement, the condition $\mathbf{u} = \mathbf{u}_w$ may be prescribed. In this case, the components of the **Velocity of moving wall** \mathbf{u}_w should be specified.

Specifying this boundary condition does not automatically cause the associated wall to move. An additional Moving Mesh interface needs to be added to physically track the wall movement in the spatial reference frame.

Leaking Wall

This boundary condition may be used to simulate a wall where fluid is leaking into or leaving the domain with the velocity $\mathbf{u} = \mathbf{u}_l$ through a perforated wall. The components of the **Fluid velocity** \mathbf{u}_l on the leaking wall should be specified.


Slip Velocity

In the microscale range, the flow condition at a boundary is seldom strictly no slip or slip. Instead, the boundary condition is something in between, and there is a **Slip velocity** at the boundary. Two phenomena account for this velocity: noncontinuum effects and the flow induced by a thermal gradient along the boundary. The components of **Velocity of moving wall: \mathbf{u}_w** should be specified. Zero values are used for a stationary wall.

When the **Use viscous slip** check box is selected, the default **Slip length** L_s is **User defined**. Another value or expression may be entered if the default value is not applicable. For **Maxwell's model** values or expressions for the **Tangential momentum accommodation coefficient** α_v and the **Mean free path** λ should be specified. Tangential accommodation coefficients are typically in the range of 0.85 to 1.0 and can be found in G. Kariadakis, A. Beskok, and N. Aluru, *Microflows and Nanoflows*, Springer Science and Business Media, 2005.

When the **Use thermal creep** check box is selected, a thermal creep contribution with **Thermal slip coefficient** σ_T is activated. Thermal slip coefficients are typically between 0.3 and 1.0 and can be found in G. Kariadakis, A. Beskok, and N. Aluru, *Microflows and Nanoflows*, Springer Science and Business Media, 2005.

CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button () and selecting **Advanced Physics Options**.



- [Theory for the Wall Boundary Condition](#)
- [Moving Mesh Interface](#) in the *COMSOL Multiphysics Reference Manual*

Inlet

This condition should be used on boundaries for which there is a net flow into the domain. To obtain a numerically well-posed problem, it is advisable to also consider the Outlet conditions when specifying an Inlet condition. For example, if the pressure is specified at the outlet, the velocity may be specified at the inlet, and vice versa. Specifying the velocity vector at both the inlet and the outlet may cause convergence difficulties.

BOUNDARY CONDITION

The available **Boundary condition** options for an inlet are **Velocity**, **Laminar inflow**, **Mass flow**, and **Pressure**. After selecting a **Boundary Condition** from the list, a section with the same or a similar name displays underneath. For example, if **Velocity** is selected, a **Velocity** section, where further settings are defined, is displayed.

VELOCITY

The **Normal inflow velocity** is specified as $\mathbf{u} = -\mathbf{n}U_0$, where \mathbf{n} is the boundary normal pointing out of the domain and U_0 is the normal inflow speed.

The **Velocity field** option sets the velocity vector to $\mathbf{u} = \mathbf{u}_0$. The components of the inlet velocity vector \mathbf{u}_0 should be defined for this choice.

PRESSURE CONDITIONS

This option specifies the normal stress, which in most cases is approximately equal to the pressure. If the reference pressure p_{ref} defined at the physics interface level, is equal to 0, the value of the **Pressure** p_0 , at the boundary, is the absolute pressure. Otherwise, p_0 is the relative pressure at the boundary.

- The **Suppress backflow** option adjusts the inlet pressure locally in order to prevent fluid from exiting the domain through the boundary. If suppress backflow is

deselected, the inlet boundary can become an outlet depending on the pressure field in the rest of the domain.

- **Flow direction** controls in which direction the fluid enters the domain.
 - For **Normal flow**, it prescribes zero tangential velocity component.
 - For **User defined**, an **Inflow velocity direction \mathbf{d}_u** (dimensionless) should be specified. The magnitude of \mathbf{d}_u does not matter, only the direction. \mathbf{d}_u must point into the domain.

LAMINAR INFLOW

This boundary condition is applicable when the fluid enters the domain from a long pipe or channel, in which the laminar flow profile is fully developed. The normal stress at the inlet is determined from the flow conditions at the entrance to a fictitious channel of length L_{entr} appended to the boundary. The inflow can be specified by the **Average velocity U_{av}** , the **Flow rate V_0** , or the **Entrance pressure p_{entr}** .

The **Entrance pressure p_{entr}** The **Entrance length L_{entr}** should be significantly greater than $0.06\text{Re}D$, where Re is the Reynolds number and D is the inlet length scale (hydraulic diameter), in order that the flow can adjust to a fully developed laminar profile.

The **Constrain outer edges to zero** option forces the laminar profile to go to zero at the bounding points or edges of the inlet channel. Otherwise the velocity is defined by the boundary condition of the adjacent boundary in the computational domain. For example, if one end of a boundary with a **Laminar inflow** condition connects to a slip boundary, the laminar profile will have a maximum at that end.

MASS FLOW

The mass flow at an inlet can be specified by the **Mass flow rate**, the **Pointwise mass flux**, the **Standard flow rate**, or the **Standard flow rate (SCCM)**.

Mass Flow Rate

The **Mass flow rate** option sets the integrated mass flow across the entire boundary, the **Normal mass flow rate** to a specific value, m . The mass flow is assumed to be parallel to the boundary normal, and the tangential flow velocity is set to zero.

For 2D components, the **Channel thickness d_{bc}** is used to define the area across which the mass flow occurs. This setting is not applied to the whole model. Line or surface integrals of the mass flow over the boundary evaluated during postprocessing or used in integration coupling operators do not include this scaling automatically. Such results should be appropriately scaled when comparing them with the specified mass flow.

Pointwise Mass Flux

The **Pointwise mass flux** sets the mass flow parallel to the boundary normal. The tangential flow velocity is set to zero. The mass flux is a model input, which means that COMSOL Multiphysics can take its value from another physics interface when available. When **User defined** is selected a value or function M_f should be specified for the **Mass flux**.

Standard Flow Rate

The **Standard flow rate** Q_{sv} sets a standard volumetric flow rate, according to the SEMI standard E12-0303. The mass flow rate is specified as the volumetric flow rate of a gas at standard density—the **Mean molar mass** M_n divided by a **Standard molar volume** V_m (that is, the volume of one mole of a perfect gas at standard pressure and standard temperature). The flow occurs across the whole boundary in the direction of the boundary normal and is computed by a surface (3D) or line (2D) integral. The tangential flow velocity is set to zero.

The standard density can be defined directly, or by specifying a standard pressure and temperature, in which case the ideal gas law is assumed. The options in the **Standard flow rate defined by** list are:


- **Standard density**, for which the **Standard molar volume** V_m should be specified.
- **Standard pressure and temperature**, for which the **Standard pressure** P_{st} and the **Standard temperature** T_{st} should be defined.

For 2D components, the **Channel thickness** d_{bc} is used to define the area across which the mass flow occurs. This setting is not applied to the whole model. Line or surface integrals of the mass flow over the boundary evaluated during post-processing or used in integration coupling operators do not include this scaling automatically. Such results should be appropriately scaled when comparing them with the specified mass flow.

Standard Flow Rate (SCCM)

The **Standard flow rate (SCCM)** boundary condition is equivalent to the **Standard flow rate** boundary condition, except that the flow rate is entered directly in SCCMs (standard cubic centimeters per minute) without the requirement to specify units. Here, the dimensionless **Number of SCCM units** Q_{sccm} should be specified.

CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button () and selecting **Advanced Physics Options**.



- [Prescribing Inlet and Outlet Conditions](#)
- [Normal Stress Boundary Condition](#)

Outlet

This condition should be used on boundaries for which there is a net outflow from the domain. To obtain a numerically well-posed problem, it is advisable to also consider the Inlet conditions when specifying an Outlet condition. For example, if the velocity is specified at the inlet, the pressure may be specified at the outlet, and vice versa. Specifying the velocity vector at both the inlet and the outlet may cause convergence difficulties. Selecting appropriate outlet conditions for the Navier-Stokes equations is a nontrivial task. Generally, if there is something interesting happening at an outflow boundary, the computational domain should be extended to include this phenomenon.

BOUNDARY CONDITION

The available **Boundary condition** options for an outlet are **Pressure**, **Laminar outflow**, and **Velocity**.

PRESSURE CONDITIONS

This option specifies the normal stress, which in most cases is approximately equal to the pressure. The tangential stress component is set to zero. If the reference pressure p_{ref} , defined at the physics interface level, is equal to 0, the value of the **Pressure** p_0 , at the boundary, is the absolute pressure. Otherwise, p_0 is the relative pressure at the boundary.

- The **Normal flow** option changes the no tangential stress condition to a no tangential velocity condition. This forces the flow to exit (or enter) the domain perpendicularly to the outlet boundary.
- The **Suppress backflow** check box is selected by default. This option adjusts the outlet pressure in order to prevent fluid from entering the domain through the boundary.

VELOCITY

See the **Inlet** node [Velocity](#) section for the settings.

LAMINAR OUTFLOW


This boundary condition is applicable when the flow exits the domain into a long pipe or channel, at the end of which a laminar flow profile is fully developed. The normal stress at the outlet is determined from the flow conditions at the end of a fictitious channel appended to the outlet boundary. The outflow can be specified by the **Average velocity** U_{av} , the **Flow rate** V_0 , or the **Exit pressure** p_{exit} .

the **Entrance pressure** p_{exit}

The **Exit length** L_{exit} should be significantly greater than $0.06ReD$, where Re is the Reynolds number and D is the outlet length scale (hydraulic diameter), in order that the flow can adjust to a fully developed laminar profile.

The **Constrain outer edges to zero** option forces the laminar profile to go to zero at the bounding points or edges of the outlet channel. Otherwise the velocity is defined by the boundary condition of the adjacent boundary in the computational domain. For example, if one end of a boundary with a **Laminar outflow** condition connects to a slip boundary, the laminar profile will have a maximum at that end.

CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button () and selecting **Advanced Physics Options**.



Prescribing Inlet and Outlet Conditions

Symmetry

The **Symmetry** boundary condition prescribes no penetration and vanishing shear stresses. The boundary condition is a combination of a Dirichlet condition and a Neumann condition:

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \left(-p\mathbf{I} + \left(\mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \right) \right) \mathbf{n} = \mathbf{0}$$

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad (-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = \mathbf{0}$$

for the compressible and incompressible formulations. The Dirichlet condition takes precedence over the Neumann condition, and the above equations are equivalent to the following equation for both the compressible and incompressible formulations:


$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \mathbf{K} - (\mathbf{K} \cdot \mathbf{n})\mathbf{n} = \mathbf{0}$$

$$\mathbf{K} = \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)\mathbf{n}$$

BOUNDARY SELECTION

For 2D axial symmetry, a boundary condition does not need to be defined for the symmetry axis at $r = 0$. The software automatically provides a condition that prescribes $u_r = 0$ and vanishing stresses in the z direction and adds an **Axial Symmetry** node that implements these conditions on the axial symmetry boundaries only.

CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button () and selecting **Advanced Physics Options**.

Open Boundary

The **Open Boundary** condition describes boundaries in contact with a large volume of fluid. Fluid can both enter and leave the domain on boundaries with this type of condition.

BOUNDARY CONDITIONS

The **Boundary condition** options for open boundaries are **Normal stress** and **No viscous stress**.

Normal Stress

The **Normal stress** f_0 condition implicitly imposes $p \approx f_0$.

No Viscous Stress

The **No Viscous Stress** condition specifies vanishing viscous stress on the boundary. This condition does not provide sufficient information to fully specify the flow at the open boundary and must at least be combined with pressure constraints at adjacent points.

The **No viscous stress** condition prescribes:

$$\left(\mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \right) \mathbf{n} = \mathbf{0}$$

$$\mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)\mathbf{n} = \mathbf{0}$$

for the compressible and the incompressible formulations. This condition can be useful in some situations because it does not impose any constraint on the pressure. A typical example is a model with volume forces that give rise to pressure gradients that are hard

to prescribe in advance. To make the model numerically stable, this boundary condition should be combined with a point constraint on the pressure.

Boundary Stress

The **Boundary Stress** node adds a boundary condition that represents a general class of conditions also known as traction boundary conditions.

BOUNDARY CONDITION

The **Boundary condition** options for the boundary stress are **General stress**, **Normal stress**, and **Normal stress, normal flow**.

General Stress

When **General stress** is selected, the components for the **Stress** \mathbf{F} should be specified. The total stress on the boundary is set equal to the given stress \mathbf{F} :

$$\left(-p\mathbf{I} + \left(\mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I}\right)\right)\mathbf{n} = \mathbf{F}$$

$$(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = \mathbf{F}$$

for the compressible and the incompressible formulations.

This boundary condition implicitly sets a constraint on the pressure that for 2D flows is

$$p = 2\mu \frac{\partial u_n}{\partial n} - \mathbf{n} \cdot \mathbf{F} \quad (4-1)$$

If $\partial u_n / \partial n$ is small, [Equation 4-1](#) states that $p \approx -\mathbf{n} \cdot \mathbf{F}$.

Normal Stress

[Normal Stress](#) is described for the Open Boundary node.

Normal Stress, Normal Flow

For **Normal stress, normal flow**, the magnitude of the **Normal stress** f_0 should be specified. The tangential velocity is set to zero on the boundary:

$$\mathbf{n}^T \left(-p\mathbf{I} + \left(\mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I}\right)\right)\mathbf{n} = -f_0, \quad \mathbf{t} \cdot \mathbf{u} = 0$$

$$\mathbf{n}^T (-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = -f_0, \quad \mathbf{t} \cdot \mathbf{u} = 0$$


for the compressible and the incompressible formulations.

This boundary condition implicitly sets a constraint on the pressure that for 2D flows is

$$p = 2\mu \frac{\partial u_n}{\partial n} + f_0 \quad (4-2)$$

If $\partial u_n / \partial n$ is small, Equation 4-2 states that $p \approx f_0$.

CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button () and selecting **Advanced Physics Options**.

If **Normal Stress**, **Normal Flow** is selected as the **Boundary condition**, then to **Apply reaction terms on** all dependent variables, the **All physics (symmetric)** option should be selected. Alternatively, the **Individual dependent variables** could be selected to restrict the reaction terms as needed.

Vacuum Pump

The **Vacuum Pump** node models the effective outlet pressure created by a vacuum pump device that is attached to the outlet.

PARAMETERS

The **Static pressure curve** option defines a lumped curve—**Linear**, **Static pressure curve data**, or **User defined**.

Linear

For the **Linear** option, values or expressions should be specified for the **Static pressure at no flow** p_{nf} and the **Free delivery flow rate** $V_{0,fd}$.

The static pressure curve value is equal to the static pressure at no flow rate when $V_0 < 0$ and equal to 0 when the flow rate is larger than the free delivery flow rate.

Static Pressure Curve Data


The **Static pressure curve data** option can be used to enter or load data under the **Static Pressure Curve Data** section below. The interpolation between points given in the **Static Pressure Curve Data** table is defined using the **Interpolation function type** list in the **Static Pressure Curve Interpolation** section.

User Defined

The **User defined** option can be used to enter a different value or expression. The flow rate across the selection where this boundary condition is applied is defined by $phys_id.V0$ where $phys_id$ is the name (for example, $phys_id$ is **spf** by default for

this physics interface). In order to avoid unexpected behavior, the function used for the fan curve is the maximum of the user-defined function and 0.

STATIC PRESSURE CURVE DATA

This section displays when **Static pressure curve data** is selected as the **Static pressure curve**. In the table, values or expressions for the **Flow rate** and **Static pressure curve** should be specified. Alternatively, a text file can be imported by clicking the **Load from file** button () under the table.

UNITS

This section is available when **Static pressure curve data** is selected as the **Static pressure curve**. Units can be specified here for the **Flow rate** and **Static pressure curve**.

STATIC PRESSURE CURVE INTERPOLATION

This section is available when **Static pressure curve data** is selected as the **Static pressure curve**. The available options for the **Interpolation function type** are **Linear**, **Piecewise cubic**, and **Cubic spline**.

The extrapolation method always returns a constant value. In order to avoid problems with undefined data, the function used for the boundary condition is the maximum of the interpolated function and 0.



Vacuum Pump Boundary Condition

Periodic Flow Condition

The **Periodic Flow Condition** splits its selection into a source group and a destination group. Fluid that leaves the domain through one of the destination boundaries enters the domain through the corresponding source boundary. This corresponds to a situation where the geometry is a periodic part of a larger geometry. If the boundaries are not parallel to each other, the velocity vector is automatically transformed.

If the boundaries are curved, it is recommended to only include two boundaries.

No input is required when **Compressible flow (Ma<0.3)** is selected for **Compressibility** under the **Physical Model** section for the physics interface. Typically when a periodic boundary condition is used with a compressible flow, the pressure is the same at both boundaries and the flow is driven by a volume force.


PRESSURE DIFFERENCE

This section is available when **Incompressible flow** is selected for **Compressibility** under the **Physical Model** section for the physics interface.

A value or expression should be specified for the **Pressure difference**, $p_{\text{src}} - p_{\text{dst}}$. This pressure difference can, for example, drive the fully developed flow in a channel.

To set up a periodic boundary condition, both boundaries must be selected in the **Periodic Flow Condition** node. COMSOL Multiphysics automatically assigns one boundary as the source and the other as the destination. To manually set the destination selection, a **Destination Selection** subnode is available from the context menu (by right-clicking the parent node) or from the **Physics** toolbar, **Attributes** menu. All destination sides must be connected.

CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button () and selecting **Advanced Physics Options**.

ORIENTATION OF SOURCE

For information about the **Orientation of Source** section, see [Orientation of Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

Pipe Connection

This feature is available with a license for the Pipe Flow Module. For details, see [Pipe Connection](#) in the *Pipe Flow Module User's Guide*.

Flow Continuity

The **Flow Continuity** condition is suitable for pairs where the boundaries match; it prescribes that the flow field is continuous across the pair.

A **Wall** subnode is added by default and it applies to the parts of the pair boundaries where a source boundary lacks a corresponding destination boundary and vice versa. The **Wall** feature can be overridden by any other boundary condition that applies to exterior boundaries. By right-clicking the **Flow Continuity** node, additional Fallback feature subnodes can be added.


Pressure Point Constraint

The **Pressure Point Constraint** condition can be used to specify the pressure level. If it is not possible to specify the pressure level using a boundary condition, the pressure level must be set in some other way, for example, by specifying a fixed pressure at a point.

PRESSURE CONSTRAINT

The relative pressure value is set by specifying the **Pressure** p_0 . Or, if the reference pressure p_{ref} defined at the physics interface level is equal to zero, p_0 represents the absolute pressure.

CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button () and selecting **Advanced Physics Options**.

Point Mass Source

This feature requires at least one of the following licenses: Batteries & Fuel Cells Module, CFD Module, Chemical Reaction Engineering Module, Corrosion Module, Electrochemistry Module, Electrodeposition Module, Microfluidics Module, Pipe Flow Module, or Subsurface Flow Module.

The **Point Mass Source** feature models mass flow originating from an infinitely small domain centered around a point

For the Reacting Flow in Porous Media, Diluted Species interface, which is available with the CFD Module, Chemical Reaction Engineering Module, or Batteries & Fuel Cells Module, there are two nodes: one for the fluid flow (**Fluid Point Source**) and one for the species (**Species Point Source**).

SOURCE STRENGTH

The source **Mass flux**, \dot{q}_p should be specified. A positive value results in mass being ejected from the point into the computational domain. A negative value results in mass being removed from the computational domain.

Point sources located on a boundary or on an edge affect the adjacent computational domains. This has the effect, for example, that a point source located on a symmetry plane has twice the given strength.



Line Mass Source

This feature requires at least one of the following licenses: Batteries & Fuel Cells Module, CFD Module, Chemical Reaction Engineering Module, Corrosion Module, Electrochemistry Module, Electrodeposition Module, Microfluidics Module, Pipe Flow Module, or Subsurface Flow Module.

The **Line Mass Source** feature models mass flow originating from a tube region with infinitely small radius.

For the Reacting Flow in Porous Media, Diluted Species interface, which is available with the CFD Module, Chemical Reaction Engineering Module, or Batteries & Fuel Cells Module, there are two nodes, one for the fluid flow (**Fluid Line Source**) and one for the species (**Species Line Source**).

SELECTION

The **Line Mass Source** feature is available for all dimensions, but the applicable selection differs between the dimensions.

MODEL DIMENSION	APPLICABLE GEOMETRICAL ENTITY
2D	Points
2D Axisymmetry	Points not on the axis of symmetry
3D	Edges

SOURCE STRENGTH

The source **Mass flux**, \dot{q}_1 , should be specified. A positive value results in mass being ejected from the line into the computational domain and a negative value means that mass is removed from the computational domain.

Line sources located on a boundary affect the adjacent computational domains. This, for example, has the effect that a line source located on a symmetry plane has twice the given strength.



Numerical Stability — Stabilization Techniques for Fluid Flow in the
COMSOL Multiphysics Reference Manual

Gravity

This feature requires at least one of the following licenses: CFD Module, Heat Transfer Module.

The **Gravity** global feature is automatically added when **Include gravity** is selected at the interface level in the **Physical Model** settings. It defines the gravity forces from the **Acceleration of gravity** value.

ACCELERATION OF GRAVITY

The **Acceleration of gravity** (SI unit m/s, default value $-g_{\text{const}}\mathbf{e}_z$ in 2Daxi and 3D and $-g_{\text{const}}\mathbf{e}_y$ in 2D) is used to define the gravity forces. It should be a global quantity.

Theory for the Single-Phase Flow Laminar Interface

The theory for the Single-Phase Flow, Laminar Flow interface is described in this section:

- [General Single-Phase Flow Theory](#)
- [Compressible Flow](#)
- [Weakly Compressible Flow](#)
- [The Mach Number Limit](#)
- [Incompressible Flow](#)
- [The Reynolds Number](#)
- [The Boussinesq Approximation](#)
- [Theory for the Wall Boundary Condition](#)
- [Prescribing Inlet and Outlet Conditions](#)
- [Laminar Inflow](#)
- [Laminar Outflow](#)
- [Mass Flow](#)
- [No Viscous Stress](#)
- [Normal Stress Boundary Condition](#)
- [Vacuum Pump Boundary Condition](#)
- [Mass Sources for Fluid Flow](#)
- [Numerical Stability—Stabilization Techniques for Fluid Flow](#)
- [Solvers for Laminar Flow](#)
- [Pseudo Time Stepping for Laminar Flow Models](#)
- [Discontinuous Galerkin Formulation](#)
- [Particle Tracing in Fluid Flow](#)
- [References for the Single-Phase Flow, Laminar Flow Interfaces](#)



The theory about most boundary conditions is found in [Ref. 2](#).

General Single-Phase Flow Theory

The Single-Phase Fluid Flow interfaces are based on the Navier-Stokes equations, which in their most general form read

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (4-3)$$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho (\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot [-p \mathbf{I} + \boldsymbol{\tau}] + \mathbf{F} \quad (4-4)$$

$$\rho C_p \left(\frac{\partial T}{\partial t} + (\mathbf{u} \cdot \nabla) T \right) = -(\nabla \cdot \mathbf{q}) + \boldsymbol{\tau} : \mathbf{S} - \frac{T}{\rho} \frac{\partial \rho}{\partial T} \bigg|_p \left(\frac{\partial p}{\partial t} + (\mathbf{u} \cdot \nabla) p \right) + Q \quad (4-5)$$

where

- ρ is the density (SI unit: kg/m³)
- \mathbf{u} is the velocity vector (SI unit: m/s)
- p is pressure (SI unit: Pa)
- $\boldsymbol{\tau}$ is the viscous stress tensor (SI unit: Pa)
- \mathbf{F} is the volume force vector (SI unit: N/m³)
- C_p is the specific heat capacity at constant pressure (SI unit: J/(kg·K))
- T is the absolute temperature (SI unit: K)
- \mathbf{q} is the heat flux vector (SI unit: W/m²)
- Q contains the heat sources (SI unit: W/m³)
- \mathbf{S} is the strain-rate tensor:

$$\mathbf{S} = \frac{1}{2}(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)$$

The operation “:” denotes a contraction between tensors defined by

$$\mathbf{a} : \mathbf{b} = \sum_n \sum_m a_{nm} b_{nm} \quad (4-6)$$

This is sometimes referred to as the double dot product.

Equation 4-3 is the continuity equation and represents conservation of mass.

Equation 4-4 is a vector equation which represents conservation of momentum.

Equation 4-5 describes the conservation of energy, formulated in terms of

temperature. This is an intuitive formulation that facilitates boundary condition specifications.

To close the equation system, [Equation 4-3](#) through [Equation 4-5](#), constitutive relations are needed. For a Newtonian fluid, which has a linear relationship between stress and strain, Stokes ([Ref. 1](#)) deduced the following expression:

$$\boldsymbol{\tau} = 2\mu\mathbf{S} - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \quad (4-7)$$

The dynamic viscosity, μ (SI unit: Pa·s), for a Newtonian fluid is allowed to depend on the thermodynamic state but not on the velocity field. All gases and many liquids can be considered Newtonian. Examples of non-Newtonian fluids are honey, mud, blood, liquid metals, and most polymer solutions. Other commonly used constitutive relations are Fourier's law of heat conduction and the ideal gas law.

In theory, the same equations describe both laminar and turbulent flows. In practice, however, the mesh resolution required to simulate turbulence with the Laminar Flow interface makes such an approach impractical.



There are several books where derivations of the Navier-Stokes equations and detailed explanations of concepts such as Newtonian fluids can be found. See, for example, the classical text by Batchelor ([Ref. 3](#)) and the more recent work by Panton ([Ref. 4](#)).

Many applications describe isothermal flows for which [Equation 4-5](#) is decoupled from [Equation 4-3](#) and [Equation 4-4](#).

2D AXISYMMETRIC FORMULATIONS

A 2D axisymmetric formulation of [Equation 4-3](#) and [Equation 4-4](#) requires $\partial/\partial\phi$ to be zero. That is, there must be no gradients in the azimuthal direction. A common additional assumption is however that $u_\phi=0$. In such cases, the ϕ -equation can be removed from [Equation 4-4](#). The resulting system of equations is both easier to converge and computationally less expensive compared to retaining the ϕ -equation. The default 2D axisymmetric formulation of [Equation 4-3](#) and [Equation 4-4](#) therefore assumes that

$$\begin{aligned} \partial/\partial\phi &= 0 \\ u_\phi &= 0 \end{aligned}$$

Compressible Flow

The equations of motion for a single-phase fluid are the continuity equation:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (4-8)$$

and the momentum equation:

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nabla \cdot \left(\mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} \right) + \mathbf{F} \quad (4-9)$$

These equations are applicable for incompressible as well as for compressible flow with density and viscosity variations.

Weakly Compressible Flow

The same equations as for [Compressible Flow](#) are applied for weakly compressible flow. The only difference is that the density is evaluated at the reference pressure. The density may be a function of other quantities, in particular it may be temperature dependent.

The weakly compressible flow equations are valid for incompressible as well as compressible flow with density variations independent of the pressure.

The Mach Number Limit

An important dimensionless number in fluid dynamics is the Mach number, Ma , defined by

$$\text{Ma} = \frac{|\mathbf{u}|}{a}$$

where a is the speed of sound. A flow is formally incompressible when $\text{Ma} = 0$. This is theoretically achieved by letting the speed of sound tend to infinity. The Navier-Stokes equations then have the mathematical property that pressure disturbances are instantaneously propagated throughout the entire domain. This results in a parabolic equation system.

The momentum equation, [Equation 4-9](#), is parabolic for unsteady flow and elliptic for steady flow, whereas the continuity equation, [Equation 4-8](#), is hyperbolic for both steady and unsteady flow. The combined system of equations is thus hybrid parabolic-hyperbolic for unsteady flow and hybrid elliptic-hyperbolic for steady flow.

An exception occurs when the viscous term in Equation 4-9 becomes vanishingly small, such as at an outflow boundary, in which case the momentum equation becomes locally hyperbolic. The number of boundary conditions to apply on the boundary then depends on the number of characteristics propagating into the computational domain. For the purely hyperbolic system, the number of characteristics propagating from the boundary into the domain changes as the Mach number passes through unity. Hence, the number of boundary conditions required to obtain a numerically well-posed system must also change. The compressible formulation of the laminar and turbulent interfaces uses the same boundary conditions as the incompressible formulation, which implies that the compressible interfaces are not suitable for flows with a Mach number larger than or equal to one.

The practical Mach number limit is lower than one, however. The main reason is that the numerical scheme (stabilization and boundary conditions) of the Laminar Flow interface does not recognize the direction and speed of pressure waves. The fully compressible Navier-Stokes equations do, for example, start to display very sharp gradients already at moderate Mach numbers. But the stabilization for the single-phase flow interface does not necessarily capture these gradients. It is impossible to give an exact limit where the low Mach number regime ends and the moderate Mach number regime begins, but a rule of thumb is that the Mach number effects start to appear at $Ma = 0.3$. For this reason, the compressible formulation is referred to as *Compressible flow* ($Ma < 0.3$) in COMSOL Multiphysics.

Incompressible Flow

When the temperature variations in the flow are small, a single-phase fluid can often be assumed incompressible; that is, ρ is constant or nearly constant. This is the case for all liquids under normal conditions and also for gases at low velocities. For constant ρ , Equation 4-8 reduces to

$$\rho \nabla \cdot \mathbf{u} = 0 \quad (4-10)$$

and Equation 4-9 becomes

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho (\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot [-p \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)] + \mathbf{F} \quad (4-11)$$

In order to apply the incompressible flow formulation, the density is evaluated at the reference pressure level, and both the density and viscosity are evaluated at the reference temperature. However, if the density is a function of other quantities such as

a concentration field, the user has to make sure that the density is defined as constant when the incompressible flow formulation is used.

The Reynolds Number

A fundamental characteristic in analyses of fluid flow is the Reynolds number:

$$\text{Re} = \frac{\rho UL}{\mu}$$

where U denotes a velocity scale, and L denotes a representative length. The Reynolds number represents the ratio between inertial and viscous forces. At low Reynolds numbers, viscous forces dominate and tend to damp out all disturbances, which leads to laminar flow. At high Reynolds numbers, the damping in the system is very low, giving small disturbances the possibility to grow by nonlinear interactions. If the Reynolds number is high enough, the flow field eventually ends up in a chaotic state called turbulence.

Observe that the Reynolds number can have different meanings depending on the length scale and velocity scale. To be able to compare two Reynolds numbers, they must be based on equivalent length and velocity scales.

The Fluid Flow interfaces automatically calculate the local cell Reynolds number $\text{Re}^c = \rho |\mathbf{u}| h / (2\mu)$ using the element length h for L and the magnitude of the velocity vector \mathbf{u} for the velocity scale U . This Reynolds number is not related to the character of the flow field, but to the stability of the numerical discretization. The risk for numerical oscillations in the solution increases as Re^c grows. The cell Reynolds number is a predefined quantity available for visualization and evaluation (typically it is available as: `spf.cellRe`).

The Boussinesq Approximation

The Boussinesq approximation is a way to treat certain simple cases of buoyant flow without having to use the compressible formulation of the Navier-Stokes equations.

The Boussinesq approximation assumes that variations in density have no effect on the flow field except that they give rise to a buoyancy force. The density is assigned a reference value, ρ_0 , everywhere except in the volume force term, which is set to

$$\mathbf{F} = (\rho_0 + \Delta\rho)\mathbf{g} \quad (4-12)$$

where \mathbf{g} is the gravity vector. A further simplification is often possible. Because \mathbf{g} can be written in terms of a potential, Φ , Equation 4-12 can be written as:

$$\mathbf{F} = -\nabla(\rho_0\Phi) + \Delta\rho\mathbf{g}$$

The first part can be canceled out by splitting the true pressure, p , into a hydrodynamic component, P , and a hydrostatic component, $-\rho_0\Phi$. Equation 4-10 and Equation 4-11 are expressed in terms of the hydrodynamic pressure $P = p + \rho_0\Phi$:

$$\rho\nabla \cdot \mathbf{u} = 0 \quad (4-13)$$

$$\rho_0 \frac{\partial \mathbf{u}}{\partial t} + (\rho_0 \mathbf{u} \cdot \nabla) \mathbf{u} = -\nabla P + \nabla \cdot (\mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)) + \mathbf{g}\Delta\rho \quad (4-14)$$

To obtain the Boussinesq approximation on this form, the flow must be defined as **Incompressible** with the **Include gravity** and **Use reduced pressure** options selected in the flow interface, and the **Non-Isothermal Flow** multiphysics feature should be used to coupled the heat transfer and fluid flow interfaces.

In practice, the shift from p to P can be ignored except where the pressure appears in boundary conditions. The pressure that is specified at boundaries is the reduced pressure in this case. For example, at a vertical outflow or inflow boundary, the reduced pressure is typically a constant, whereas the true pressure is a function of the vertical coordinate.

The system formed by Equation 4-13 and Equation 4-14 has its limitations. The main assumption is that the density fluctuations must be small; that is, $\Delta\rho/\rho_0 \ll 1$. There are also some more subtle constraints that, for example, make the Boussinesq approximation unsuitable for systems of very large dimensions. An excellent discussion of the Boussinesq approximation and its limitations appears in Chapter 14 of Ref. 10.

Theory for the Wall Boundary Condition

See [Wall](#) for the node settings.

SLIP

The [Slip](#) condition assumes that there are no viscous effects at the slip wall and hence, no boundary layer develops. From a modeling point of view, this is a reasonable approximation if the important effect of the wall is to prevent fluid from leaving the domain. Mathematically, the constraint can be formulated as:

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad (-p\mathbf{I} + \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T))\mathbf{n} = \mathbf{0}$$

The no penetration term takes precedence over the Neumann part of the condition and the above expression is therefore equivalent to

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \mathbf{K} - (\mathbf{K} \cdot \mathbf{n})\mathbf{n} = \mathbf{0}$$

$$\mathbf{K} = \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)\mathbf{n}$$

expressing that there is no flow across the boundary and no viscous stress in the tangential direction.

SLIDING WALL

The [Sliding Wall](#) boundary condition is appropriate if the wall behaves like a conveyor belt; that is, the surface is sliding in its tangential direction. The wall does not have to actually move in the coordinate system.

- In 2D, the tangential direction is unambiguously defined by the direction of the boundary, but the situation becomes more complicated in 3D. For this reason, this boundary condition has slightly different definitions in the different space dimensions.
- For 2D and 2D axisymmetric components, the velocity is given as a scalar U_w and the condition prescribes

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \mathbf{u} \cdot \mathbf{t} = U_w$$

where $\mathbf{t} = (n_y, -n_x)$ for 2D and $\mathbf{t} = (n_z, -n_r)$ for axial symmetry.

- For 3D components, the velocity is set equal to a given vector \mathbf{u}_w projected onto the boundary plane:

$$\mathbf{u} = \frac{\mathbf{u}_w - (\mathbf{n} \cdot \mathbf{u}_w)\mathbf{n}}{\|\mathbf{u}_w - (\mathbf{n} \cdot \mathbf{u}_w)\mathbf{n}\|} \|\mathbf{u}_w\|$$

The normalization makes \mathbf{u} have the same magnitude as \mathbf{u}_w even if \mathbf{u}_w is not exactly parallel to the wall.

SLIP VELOCITY

In the microscale range, the flow at a boundary is seldom strictly no slip or slip. Instead, the boundary condition is something in between, and there is a slip velocity at the boundary. Two phenomena account for this velocity: violation of the continuum hypothesis for the viscosity and flow induced by a thermal gradient along the boundary.

The following equation relates the viscosity-induced jump in tangential velocity to the tangential shear stress along the boundary:

$$\Delta u = \frac{1}{\beta} \tau_{n,t}$$

For gaseous fluids, the coefficient β is given by

$$\beta = \frac{\mu}{\left(\frac{2 - \alpha_v}{\alpha_v}\right) \lambda}$$

where μ is the fluid's dynamic viscosity (SI unit: Pa·s), α_v represents the tangential momentum accommodation coefficient (TMAC) (dimensionless), and λ is the molecules' mean free path (SI unit: m). The tangential accommodation coefficients are typically in the range of 0.85 to 1.0 and can be found in [Ref. 15](#).

A simpler expression for β is

$$\beta = \frac{\mu}{L_s}$$

where L_s , the slip length (SI unit: m), is a straight channel measure of the distance from the boundary to the virtual point outside the flow domain where the flow profile extrapolates to zero. This equation holds for both liquids and gases.

Thermal creep results from a temperature gradient along the boundary. The following equation relates the thermally-induced jump in tangential velocity to the tangential gradient of the natural logarithm of the temperature along the boundary:

$$\Delta u = \sigma_T \frac{\mu}{\rho} \nabla_t \log T$$

where σ_T is the thermal slip coefficient (dimensionless) and ρ is the density of the fluid. The thermal slip coefficients range between 0.3 and 1.0 and can be found in [Ref. 15](#).

Combining the previous relationships results in the following equation:

$$\mathbf{u} - \mathbf{u}_{w,t} = \frac{L_s}{\mu} \tau_{n,t} + \sigma_T \frac{\mu}{\rho T} \nabla_t T$$

Relate the tangential shear stress to the viscous boundary force by

$$\tau_{n,t} = \mathbf{K} - (\mathbf{n} \cdot \mathbf{K}) \mathbf{n}$$

where the components of \mathbf{K} are the Lagrange multipliers that are used to implement the boundary condition. Similarly, the tangential temperature gradient results from the difference of the gradient and its normal projection:

$$\nabla_t T = \nabla T - (\mathbf{n} \cdot \nabla T) \mathbf{n}$$

Use Viscous Slip

When viscous slip is used, select Maxwell's model to calculate L_s using:

$$L_s = \left(\frac{2 - \alpha_v}{\alpha_v} \right) \lambda$$

Also see [Wall](#) for the node settings.

Prescribing Inlet and Outlet Conditions

The Navier-Stokes equations can show large variations in mathematical behavior, ranging from almost completely elliptic to almost completely hyperbolic. This has implications when it comes to prescribing admissible boundary conditions. There is also a discrepancy between mathematically valid boundary conditions and practically useful boundary conditions. See [Inlet](#) and [Outlet](#) for the node settings.

INLET CONDITIONS

An inlet requires specification of the velocity components. The most robust way to do this is to prescribe a velocity field using a Velocity condition.

A common alternative to prescribing the complete velocity field is to prescribe a pressure and all but one velocity component. The pressure cannot be specified pointwise because this is mathematically over-constraining. Instead the pressure can be specified via a stress condition:

$$-p + 2\mu \frac{\partial u_n}{\partial n} = F_n \quad (4-15)$$

where $\partial u_n / \partial n$ is the normal derivative of the normal velocity component.

[Equation 4-15](#) is prescribed by the Pressure condition in the Inlet and Outlet features and the Normal stress condition in the Open Boundary and Boundary Stress features. [Equation 4-15](#) is mathematically more stringent compared to specifying the pressure pointwise and at the same time cannot guarantee that p obtains the desired value. In practice, p is close to F_n , except for low Reynolds number flows where viscous effects are the only effects that balance the pressure. In addition to [Equation 4-15](#), all but one

velocity component must be specified. For low Reynolds numbers, this can be specified by a vanishing tangential stress condition:

$$\mu \frac{\partial u_t}{\partial n} = 0$$

which is what the Normal stress condition does. Vanishing tangential stress becomes a less well-posed inlet condition as the Reynolds number increases. The Pressure condition in the Inlet feature therefore requires a flow direction to be prescribed, which provides a well-posed condition independent of Reynolds number.

OUTLET CONDITIONS

The most common approach is to prescribe a pressure via a normal stress condition on the outlet. This is often accompanied by a vanishing tangential stress condition:

$$\mu \frac{\partial u_t}{\partial n} = 0$$

where $\partial u_t / \partial n$ is the normal derivative of the tangential velocity field. It is also possible to prescribe u_t to be zero. The latter option should be used with care since it can have a significant effect on the upstream solution.

The elliptic character of the Navier-Stokes equations mathematically permit specifying a complete velocity field at an outlet. This can, however, be difficult to apply in practice. The reason being that it is hard to prescribe the outlet velocity so that it is consistent with the interior solution at each point. The adjustment to the specified velocity then occurs across an outlet boundary layer. The thickness of this boundary layer depends on the Reynolds number; the higher the Reynolds number, the thinner the boundary layer.

ALTERNATIVE FORMULATIONS

COMSOL provides several specialized boundary conditions that either provide detailed control over the flow at the boundary or that simulate specific devices. In practice they often prescribe a velocity or a pressure, but calculate the prescribed values using for example ODEs.



Normal Stress Boundary Condition

Laminar Inflow

In order to prescribe a fully developed inlet velocity profile, this boundary condition adds a weak form contribution and constraints corresponding to unidirectional flow perpendicular to the boundary. The applied condition corresponds to the situation shown in [Figure 4-1](#): a fictitious domain of length L_{entr} is assumed to be attached to the inlet of the computational domain. The domain is an extrusion of the inlet boundary, which means that laminar inflow requires the inlet to be flat. The boundary condition uses the assumption that the flow in this fictitious domain is *fully developed laminar flow*. The “wall” boundary conditions for the fictitious domain is inherited from the real domain, Ω , unless the option to constrain outer edges or endpoints to zero is selected in which case the fictitious “walls” are no-slip walls.

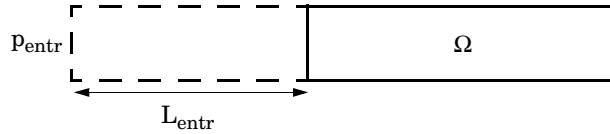


Figure 4-1: An example of the physical situation simulated when using the Laminar inflow boundary condition. Ω is the actual computational domain while the dashed domain is a fictitious domain.

If an average inlet velocity or inlet volume flow is specified instead of the pressure, COMSOL Multiphysics adds an ODE that calculates a pressure, p_{entr} , such that the desired inlet velocity or volume flow is obtained.

Laminar Outflow

In order to prescribe an outlet velocity profile, this boundary condition adds a weak form contribution and constraints corresponding to unidirectional flow perpendicular to the boundary. The applied condition corresponds to the situation shown in [Figure 4-2](#): assume that a fictitious domain of length L_{exit} is attached to the outlet of the computational domain. The domain is an extrusion of the outlet boundary, which means that laminar outflow requires the outlet to be flat. The boundary condition uses the assumption that the flow in this fictitious domain is *fully developed laminar flow*. The “wall” boundary conditions for the fictitious domain is inherited from the real

domain, Ω , unless the option to constrain outer edges or endpoints to zero is selected in which case the fictitious “walls” are no-slip walls.

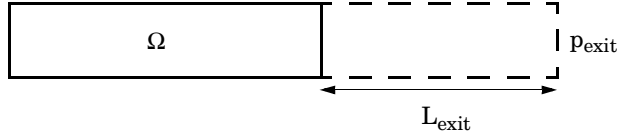


Figure 4-2: An example of the physical situation simulated when using the Laminar outflow boundary condition. Ω is the actual computational domain while the dashed domain is a fictitious domain.

If the average outlet velocity or outlet volume flow is specified instead of the pressure, the software adds an ODE that calculates p_{exit} such that the desired outlet velocity or volume flow is obtained.

Mass Flow

The **Mass Flow** boundary condition constrains the mass flowing into the domain across an inlet boundary. The mass flow can be specified in a number of ways.

POINTWISE MASS FLUX

The pointwise mass flux sets the velocity at the boundary to:

$$\mathbf{u} = -\frac{m_f}{\rho} \mathbf{n}$$

where m_f is the normal mass flux and ρ is the density.

MASS FLOW RATE

The mass flow rate boundary condition sets the total mass flow through the boundary according to:

$$-\int_{\partial\Omega} d_{bc} \rho (\mathbf{u} \cdot \mathbf{n}) dS = m$$

where d_{bc} (only present in the 2D Cartesian axis system) is the boundary thickness normal to the fluid-flow domain and m is the total mass flow rate.

In addition to the constraint on the total flow across the boundary, the tangential velocity components are set to zero on the boundary

$$\mathbf{u} \times \mathbf{n} = \mathbf{0} \quad (4-16)$$

STANDARD FLOW RATE

The standard flow rate boundary condition specifies the mass flow as a standard volumetric flow rate. The mass flow through the boundary is set by the equation:

$$-\int_{\partial\Omega} d_{bc} \frac{\rho}{\rho_{st}} (\mathbf{u} \cdot \mathbf{n}) dS = Q_{sv}$$

where d_{bc} (only present in the 2D component Cartesian axis system) is the boundary thickness normal to the fluid-flow domain, ρ_{st} is the standard density, and Q_{sv} is the standard flow rate. The standard density is defined by one of the following equations:

$$\rho_{st} = \frac{M_n}{V_n}$$

$$\rho_{st} = \frac{p_{st} M_n}{R T_{st}}$$

where M_n is the mean molar mass of the fluid, V_n is the standard molar volume, p_{st} is the standard pressure, R is the universal molar gas constant, and T_{st} is the standard temperature.

Equation 4-16 or Equation 4-17 is also enforced for compressible and incompressible flow, respectively, ensuring that the normal component of the viscous stress and the tangential component of the velocity are zero at the boundary.

No Viscous Stress

For this module, and in addition to the Pressure, No Viscous Stress boundary condition, the viscous stress condition sets the viscous stress to zero:

$$\left(\mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu(\nabla \cdot \mathbf{u}) \mathbf{I} \right) \mathbf{n} = \mathbf{0}$$

$$(\mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)) \mathbf{n} = \mathbf{0}$$

using the compressible/weakly compressible and the incompressible formulation, respectively.

The condition is not a sufficient outlet condition since it lacks information about the outlet pressure. It must hence be combined with pressure point constraints on one or several points or lines surrounding the outlet.

This boundary condition is numerically the least stable outlet condition, but can still be beneficial if the outlet pressure is nonconstant due to, for example, a nonlinear volume force.

Normal Stress Boundary Condition

The total stress on the boundary is set equal to a stress vector of magnitude f_0 , oriented in the negative normal direction:

$$\left(-p\mathbf{I} + \left(\mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I}\right)\right)\mathbf{n} = -f_0\mathbf{n}$$

$$(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = -f_0\mathbf{n}$$

using the compressible/weakly compressible and the incompressible formulation, respectively.

This implies that the total stress in the tangential direction is zero. This boundary condition implicitly sets a constraint on the pressure which for 2D flows is

$$p = 2\mu \frac{\partial u_n}{\partial n} + f_0 \quad (4-17)$$

If $\partial u_n / \partial n$ is small, [Equation 4-17](#) states that $p \approx f_0$.

The Normal Stress condition is the mathematically correct version of the [Pressure Conditions](#) condition ([Ref. 4](#)), but it is numerically less stable.

Pressure Boundary Condition

For single-phase flow, a mathematically correct natural boundary condition for outlets is

$$\left(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I}\right)\mathbf{n} = -p_0\mathbf{n} \quad (4-18)$$

$$(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = -p_0\mathbf{n} \quad (4-19)$$

using the compressible/weakly compressible and the incompressible formulation, respectively.

This is a normal stress condition together with a no-tangential-stress condition. When $\mu > 0$, Equation 4-18 or Equation 4-19 can be supplemented with a tangential velocity condition

$$\mathbf{u} \cdot \mathbf{t} = 0 \quad (4-20)$$

If so, the no-tangential-stress condition is overridden. An issue with Equation 4-18 or Equation 4-19 is that it does not strongly enforce unidirectional flow on the boundary. If the prescribed pressure on an outlet is too high, parts of the outlet can actually have inflow. This is not as much of an issue for the Navier-Stokes equations as it is an issue for scalar transport equations solved along with the Navier-Stokes equations. Hence, when applying the **Pressure** boundary condition at an outlet or inlet you can further constrain the flow. With the Suppress backflow option

$$\begin{aligned} \left(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \right) \mathbf{n} &= -\hat{p}_0 \mathbf{n} \\ (-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)) \mathbf{n} &= -\hat{p}_0 \mathbf{n} \\ \hat{p}_0 &\leq p_0 \end{aligned}, \quad (4-21)$$

the normal stress is adjusted to keep

$$\mathbf{u} \cdot \mathbf{n} \geq 0 \quad (4-22)$$

Equation 4-21 effectively means that the prescribed pressure is p_0 if $\mathbf{u} \cdot \mathbf{n} \geq 0$, but smaller at locations where $\mathbf{u} \cdot \mathbf{n} < 0$. This means that Equation 4-21 does not completely prevent backflow, but the backflow is substantially reduced.

A pressure condition can also be applied at an inlet. In this case, either the normal stress is prescribed

$$\begin{aligned} \mathbf{n}^T \left(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \right) \mathbf{n} &= -\hat{p}_0 \\ \mathbf{n}^T (-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)) \mathbf{n} &= -\hat{p}_0 \\ \hat{p}_0 &\geq p_0 \end{aligned} \quad (4-23)$$

together with the tangential condition in Equation 4-20, or, a general flow direction is prescribed.

$$\begin{aligned}
\mathbf{r}_{\mathbf{u}}^T \left(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \right) \mathbf{n} &= -\hat{p}_0(\mathbf{r}_{\mathbf{u}} \cdot \mathbf{n}) \\
\mathbf{r}_{\mathbf{u}}^T (-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)) \mathbf{n} &= -\hat{p}_0(\mathbf{r}_{\mathbf{u}} \cdot \mathbf{n}) \\
\hat{p}_0 &\geq p_0 \\
\mathbf{u} - (\mathbf{u} \cdot \mathbf{r}_{\mathbf{u}})\mathbf{r}_{\mathbf{u}} &= \mathbf{0}, \quad \mathbf{r}_{\mathbf{u}} = \frac{\mathbf{d}_{\mathbf{u}}}{\|\mathbf{d}_{\mathbf{u}}\|}
\end{aligned} \tag{4-24}$$

The “>” option is used with suppress backflow to have $\mathbf{u} \cdot \mathbf{n} \leq 0$ or $\mathbf{u} \cdot \mathbf{r}_{\mathbf{u}} \geq 0$.

See [Inlet](#), [Outlet](#), [Open Boundary](#), and [No Viscous Stress](#) for the individual node settings. Note that some modules have additional theory sections describing options available with that module.

Vacuum Pump Boundary Condition

Vacuum pumps (devices) can be represented using lumped curves implemented as boundary conditions. These simplifications also imply some assumptions. In particular, it is assumed that a given boundary can only be either an inlet or an outlet. Such a boundary should not be a mix of inlets/outlets nor switch between them during a simulation.

Manufacturers usually provide curves that describe the static pressure as a function of flow rate for a vacuum pump. Also see [Vacuum Pump](#) for the node settings.

DEFINING A DEVICE AT AN OUTLET

In this case (see [Figure 4-3](#)), the device’s inlet is the interior face situated between the blue (cube) and green (circle) domains while its outlet is an external boundary, here the circular boundary of the green domain. The lumped curve gives the flow rate as a function of the pressure difference between the interior face and the external boundary. This boundary condition implementation follows the [Pressure Boundary Condition](#) for outlets with the Suppress backflow option:

$$\begin{aligned}
\left(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \right) \mathbf{n} &= -\hat{p}_0\mathbf{n} \\
\hat{p}_0 &\leq p_{\text{vacuum pump}}(V_0) - p_{\text{ref}}
\end{aligned} \tag{4-25}$$

Here, V_0 is the flow rate across the boundary and $p_{\text{vacuum pump}}(V_0)$ is the static pressure function of flow rate for the vacuum pump. p_{ref} is the reference pressure that

ensure that the absolute pressure is set to $p_{\text{vacuum pump}}$. Equation 4-25 corresponds to the compressible formulation. For incompressible flows, the term $-(2/3)\mu(\nabla \cdot \mathbf{u})$ vanishes. In 2D the thickness in the third direction, Dz , is used to define the flow rate. Vacuum pumps are modeled as rectangles in this case.

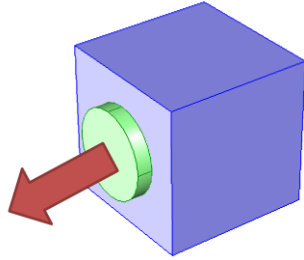


Figure 4-3: A vacuum pump at the outlet. The arrow represents the flow direction, the green circle represents the vacuum pump (that should not be part of the model), and the blue cube represents the modeled domain with an outlet boundary condition described by a lumped curve for the attached vacuum pump.

Mass Sources for Fluid Flow

There are two types of mass sources in a Single-Phase Flow interface: point sources and line sources.



These features require at least one of the following licenses: Batteries & Fuel Cells Module, CFD Module, Chemical Reaction Engineering Module, Corrosion Module, Electrochemistry Module, Electrodeposition Module, Microfluidics Module, Pipe Flow Module, or Subsurface Flow Module.

POINT SOURCE

A point source is theoretically formed by taking a mass injection/ejection, \dot{Q} (SI unit: $\text{kg}/(\text{m}^3 \cdot \text{s})$), in a small volume δV and then letting the size of the volume tend to zero while keeping the total mass flux constant. Given a point source strength, \dot{q}_p (SI unit: kg/s), this can be expressed as

$$\lim_{\delta V \rightarrow 0} \int_{\delta V} \dot{Q} = \dot{q}_p \quad (4-26)$$

An alternative way to form a point source/sink is to assume that mass is injected/extracted through the surface of a small object. Letting the object surface area tend to zero while keeping the mass flux constant, results in the same point source. For this alternative approach, effects resulting from the physical object volume, such as drag and fluid displacement, need to be neglected.

The weak contribution

$$\dot{q}_p \text{test}(p)$$

is added to a point in the geometry. As can be seen from Equation 4-26, \dot{Q} must tend to plus or minus infinity as δV tends to zero. This means that in theory the pressure also tends to plus or minus infinity.

Observe that “point” refers to the physical representation of the source. A point source can therefore only be added to points in 3D components and to points on the symmetry axis in 2D axisymmetry components. Other geometrical points in 2D components represent physical lines.

The finite element representation of Equation 4-26 corresponds to a finite pressure in a point with the effect of the point source spread out over a region around the point. The size of the region depends on the mesh and on the strength of the source. A finer mesh gives a smaller affected region, but also a more extreme pressure value. It is important not to mesh too finely around a point source since the resulting pressure can result in unphysical values for the density, for example. It can also have a negative effect on the condition number for the equation system.

LINE SOURCE

A line source can theoretically be formed by assuming a source of strength \dot{Q} (SI unit: $\text{kg}/(\text{m}^3 \cdot \text{s})$), located within a tube with cross-sectional area δS and then letting δS tend to zero, while keeping the total mass flux per unit length constant. Given a line source strength, \dot{q}_l (SI unit: $\text{kg}/(\text{m} \cdot \text{s})$), this can be expressed as

$$\lim_{\delta S \rightarrow 0} \int_{\delta S} \dot{Q} = \dot{q}_l \quad (4-27)$$

As in the point source case, an alternative approach is to assume that mass is injected/extracted through the surface of a small object. This results in the same mass



source, but requires that effects on the fluid resulting from the physical object volume are neglected.

The weak contribution


$$\dot{q}_1\text{test}(p)$$

is added to lines in 3D or to points in 2D (which represent cut-through views of lines). Line sources can also be added to the axisymmetry line in 2D axisymmetry components. It cannot, however, be added to geometrical lines in 2D because they represent physical planes.

As with a point source, it is important not to mesh too finely around the line source.

	For feature node information, see Line Mass Source and Point Mass Source in the <i>COMSOL Multiphysics Reference Manual</i> .
	<p>For the Reacting Flow in Porous Media, Diluted Species interface, which is available with the CFD Module, Chemical Reaction Engineering Module, or Batteries & Fuel Cells Module, these shared physics nodes are renamed as follows:</p> <ul style="list-style-type: none">• The Line Mass Source node is available as two nodes, one for the fluid flow (Fluid Line Source) and one for the species (Species Line Source).• The Point Mass Source node is available as two nodes, one for the fluid flow (Fluid Point Source) and one for the species (Species Point Source).

Numerical Stability—Stabilization Techniques for Fluid Flow

The momentum equation ([Equation 4-9](#) or [Equation 4-11](#)) is a (nonlinear) convection-diffusion equation. Such equations can easily become unstable if discretized using the Galerkin finite element method. Stabilized finite element methods are usually necessary in order to obtain physical solutions. The stabilization settings are found in the main Fluid Flow interface features. To display this section, click the **Show** button () and select **Stabilization**.

There are three types of stabilization methods available for Navier-Stokes—*streamline diffusion*, *crosswind diffusion*, and *isotropic diffusion*. Streamline diffusion and

crosswind diffusion are consistent stabilization methods, whereas isotropic diffusion is an inconsistent stabilization method.

For optimal functionality, the exact weak formulations of and constants in the streamline diffusion and crosswind diffusion methods depend on the order of the shape functions (basis functions) for the elements. The values of constants in the streamline diffusion and crosswind diffusion methods follow [Ref. 5](#) and [Ref. 6](#).

STREAMLINE DIFFUSION

For strongly coupled systems of equations, the streamline diffusion method must be applied to the system as a whole rather than to each equation separately. These ideas were first explored by Hughes and Mallet ([Ref. 7](#)) and were later extended to Galerkin least-squares (GLS) applied to the Navier-Stokes equations ([Ref. 8](#)). This is the streamline diffusion formulation that COMSOL Multiphysics supports. The time-scale tensor is the diagonal tensor presented in [Ref. 9](#).

Streamline diffusion is active by default because it is necessary when convection is dominating the flow.

The governing equations for incompressible flow are subject to the Babuska-Brezzi condition, which states that the shape functions (basis functions) for pressure must be of lower order than the shape functions for velocity. If the incompressible Navier-Stokes equations are stabilized by streamline diffusion, it is possible to use equal-order interpolation. Hence, streamline diffusion is necessary when using first-order elements for both velocity and pressure. This applies also if the model is solved using geometric multigrid (either as a solver or as a preconditioner) and at least one multigrid hierarchy level uses linear Lagrange elements.

CROSSWIND DIFFUSION

Crosswind diffusion can also be formulated for systems of equations, and when applied to the Navier-Stokes equations it becomes a shock-capturing operator. COMSOL Multiphysics supports the formulation in [Ref. 8](#) with a shock-capturing viscosity of the Hughes-Mallet type [Ref. 7](#).

Incompressible flows do not contain shock waves, but crosswind diffusion is still useful for introducing extra diffusion in sharp boundary layers and shear layers that otherwise would require a very fine mesh to resolve.

Crosswind diffusion is active by default as it makes it easier to obtain a solution even if the problem is fully resolved by the mesh. Crosswind diffusion also enables the iterative

solvers to use inexpensive presmoothers. If crosswind diffusion is deactivated, more expensive preconditioners must be used instead.

ISOTROPIC DIFFUSION

Isotropic diffusion adds diffusion to the Navier-Stokes equations. Isotropic diffusion significantly reduces the accuracy of the solution but does a very good job at reducing oscillations. The stability of the continuity equation is not improved.



[Numerical Stabilization](#) and [Iterative](#) in the *COMSOL Multiphysics Reference Manual*

Solvers for Laminar Flow

The Navier-Stokes equations constitute a nonlinear equation system. A nonlinear solver must hence be applied to solve the problem. The nonlinear solver iterates to reach the final solution. In each iteration, a linearized version of the nonlinear system is solved using a linear solver. In the time-dependent case, a time marching method must also be applied. The default suggestions for each of these solver elements are discussed below.

NONLINEAR SOLVER

The nonlinear solver method depends on if the model solves a stationary or a time-dependent problem.

Stationary Solver

In the stationary case, a fully coupled, damped Newton method is applied. The initial damping factor is low since a full Newton step can be harmful unless the initial values are close to the final solution. The nonlinear solver algorithm automatically regulates the damping factor in order to reach a converged solution.

For advanced models, the automatically damped Newton method might not be robust enough. A pseudo time-stepping algorithm can then be invoked. See [Pseudo Time Stepping for Laminar Flow Models](#).

Time-Dependent Solver

In the time-dependent case, the initial guess for each time step is (loosely speaking) the previous time step, which is a very good initial value for the nonlinear solver. The automatic damping algorithm is then not necessary. The damping factor in the

Newton method is instead set to a constant value slightly smaller than one. Also, for the same reason, it suffices to update the Jacobian once per time-step.

It is seldom worth the extra computational cost to update the Jacobian more than once per time step. For most models it is more efficient to restrict the maximum time step or possibly lower the damping factor in the Newton method.

LINEAR SOLVER

The linearized Navier-Stokes equation system has saddle point character, unless the density depends on the pressure. This means that the Jacobian matrix has zeros on the diagonal. Even when the density depends on the pressure, the equation system effectively shares many numerical properties with a saddle point system.

For small 2D and 3D models, the default solver suggestion is a direct solver. Direct solvers can handle most nonsingular systems and are very robust and also very fast for small models. Unfortunately, they become slow for large models and their memory requirement scales as somewhere between $N^{1.5}$ and N^2 , where N is the number of degrees of freedom in the model. The default suggestion for large 2D and 3D models is therefore the iterative GMRES solver. The memory requirement for an iterative solver optimally scales as N .

Geometric Multigrid (GMG) is used to accelerate GMRES. GMG needs smoothers but the saddle point character of the linear system restricts the number of applicable smoothers. The choices are further restricted by the anisotropic meshes frequently encountered in fluid-flow problems. Pointwise smoothers, such as SOR, are not very efficient on anisotropic meshes.

The efficiency of the smoothers is highly dependent on the numerical stabilization. Iterative solvers perform at their best when both [Streamline Diffusion](#) and [Crosswind Diffusion](#) are active.

The default smoother for P1+P1 elements is SCGS. This is an efficient and robust smoother specially designed to solve saddle point systems on meshes that contain anisotropic elements. The SCGS smoother works well even without crosswind diffusion. SCGS can sometimes work for higher-order elements, especially if **Method** in the SCGS settings is set to **Mesh element lines**. But there is no guarantee for this, so the default smoother for P2+P1 elements and P3+P2 elements is an SOR Line smoother. SOR Line handles mesh anisotropy but does not formally address the saddle point character. It does, however, function in practice provided that streamline diffusion and crosswind diffusion are both active.

A different kind of saddle point character can arise if the equation system contains ODE variables. Some advanced boundary conditions can add equations with such variables. These variables must be treated with the Vanka algorithm. SCGS includes an option to invoke Vanka. Models with higher-order elements must apply SCGS or use the Vanka smoother. The latter is the default suggestion for higher-order elements, but it does not work optimally for anisotropic meshes.

TIME-DEPENDENT SOLVERS

The default time-dependent solver for Navier-Stokes is the BDF method with maximum order set to two. Higher BDF orders are not stable for transport problems in general nor for Navier-Stokes in particular.

BDF methods have been used for a long time and are known for their stability. However, they can have severe damping effects, especially the lower-order methods. Hence, if robustness is not an issue, a model can benefit from using the generalized- α method instead. Generalized- α is a solver which has properties similar to those of the second-order BDF solver but it is much less diffusive.

Both BDF and generalized- α are per default set to automatically adjust the time step. While this works well for many models, extra efficiency and accuracy can often be gained by specifying a maximum time step. It is also often beneficial to specify an initial time step to make the solver progress smoothly in the beginning of the time series.



In the *COMSOL Multiphysics Reference Manual*:

- [Time-Dependent Solver](#)
- [Multigrid](#), [Direct](#), [Iterative](#), [SCGS](#), [SOR Line](#), and [Vanka](#)
- [Stationary Solver](#)

Pseudo Time Stepping for Laminar Flow Models

A stationary formulation has per definition no time derivatives and [Equation 4-11](#) reduces to:

$$\rho(\mathbf{u} \cdot \nabla)\mathbf{u} = \nabla \cdot [-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F} \tag{4-28}$$

Solving [Equation 4-28](#) requires a starting guess that is close enough to the final solution. If no such guess is at hand, the fully transient problem can be solved instead. This is, however, a rather costly approach in terms of computational time. An intermediate approach is to add a fictitious time derivative to [Equation 4-28](#):

$$\rho \frac{\mathbf{u} - \text{nojac}(\mathbf{u})}{\Delta \tilde{t}} + \rho(\mathbf{u} \cdot \nabla)\mathbf{u} = \nabla \cdot [-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F}$$

where $\Delta \tilde{t}$ is a *pseudo time step*. Since $\mathbf{u} - \text{nojac}(\mathbf{u})$ is always zero, this term does not affect the final solution. It does, however, affect the discrete equation system and effectively transforms a nonlinear iteration into a step of size $\Delta \tilde{t}$ of a time-dependent solver.

Pseudo time stepping is not active per default. The pseudo time step $\Delta \tilde{t}$ can be chosen individually for each element based on the local CFL number:

$$\Delta \tilde{t} = \text{CFL}_{\text{loc}} \frac{h}{|\mathbf{u}|}$$

where h is the mesh cell size. A small CFL number means a small time step. It is practical to start with a small CFL number and gradually increase it as the solution approaches steady state.

If the automatic expression for CFL_{loc} is set to the built-in variable `CFLCMP`, then the automatic setting suggests a PID regulator for the pseudo time step in the default solver. The PID regulator starts with a small CFL number and increases CFL_{loc} as the solution comes closer to convergence.

The default manual expression is

$$\begin{aligned} & 1.3^{\min(\text{niterCMP}, 9)} + \\ & \text{if}(\text{niterCMP} > 20, 9 \cdot 1.3^{\min(\text{niterCMP} - 20, 9)}, 0) + \\ & \text{if}(\text{niterCMP} > 40, 90 \cdot 1.3^{\min(\text{niterCMP} - 40, 9)}, 0) \end{aligned} \quad (4-29)$$

The variable `niterCMP` is the nonlinear iteration number. It is equal to one for the first nonlinear iteration. CFL_{loc} starts at 1.3 and increases by 30% each iteration until it reaches $1.3^9 \approx 10.6$. It remains there until iteration number 20 at which it starts to increase until it reaches approximately 106. A final increase after iteration number 40 then takes it to 1060. Equation 4-29 can, for some advanced flows, increase CFL_{loc} too slowly or too quickly. CFL_{loc} can then be tuned for the specific application.



For details about the CFL regulator, see [Pseudo Time Stepping](#) in the *COMSOL Multiphysics Reference Manual*.

Discontinuous Galerkin Formulation

Some boundary conditions are implemented using a discontinuous Galerkin formulation. These boundary conditions include

- [Wall – Slip](#)
- [Periodic Flow Condition](#)
- [Flow Continuity](#)

The formulation used in the Fluid Flow interfaces in COMSOL Multiphysics is the Symmetric Interior Penalty Galerkin method (SIPG). The SIPG method can be regarded to satisfy the boundary conditions in an integral sense rather than pointwise. More information on SIPG can be found in [Ref. 13](#).

In particular, the SIPG formulation includes a penalty parameter that must be large enough for the formulation to be coercive. The higher the value, the better the boundary condition is fulfilled, but a too high value results in an ill-conditioned equation system. The penalty parameter in COMSOL Multiphysics is implemented according to [Ref. 14](#).

Particle Tracing in Fluid Flow

The Particle Tracing Module is available to assist with these types of modeling problems.

It is possible to model particle tracing with COMSOL Multiphysics provided that the impact of the particles on the flow field is negligible. First compute the flow field, and then, as an analysis step, calculate the motion of the particles. The motion of a particle is defined by Newton's second law

$$m \frac{d^2 \mathbf{x}}{dt^2} = \mathbf{F} \left(t, \mathbf{x}, \frac{d\mathbf{x}}{dt} \right)$$

where \mathbf{x} is the position of the particle, m the particle mass, and \mathbf{F} is the sum of all forces acting on the particle. Examples of forces acting on a particle in a fluid are the drag force, the buoyancy force, and the gravity force. The drag force represents the force that a fluid exerts on a particle due to a difference in velocity between the fluid and the particle. It includes the viscous drag, the added mass, and the Basset history term. Several empirical expressions have been suggested for the drag force. One of those is the one proposed by Khan and Richardson ([Ref. 11](#)). That expression is valid for

spherical particles for a wide range of particle Reynolds numbers. The particle Reynolds number is defined as

$$\text{Re}_p = \frac{|\mathbf{u} - \mathbf{u}_p| 2r \rho}{\mu}$$

where \mathbf{u} is the velocity of the fluid, \mathbf{u}_p the particle velocity, r the particle radius, ρ the fluid density, and μ the dynamic viscosity of the fluid. The empirical expression for the drag force according to Khan and Richardson is

$$\mathbf{F} = \pi r^2 \rho |\mathbf{u} - \mathbf{u}_p| (\mathbf{u} - \mathbf{u}_p) [1.84 \text{Re}_p^{-0.31} + 0.293 \text{Re}_p^{0.06}]^{3.45}$$




The model *Flow Past a Cylinder* (Application Library path **COMSOL_Multiphysics/Fluid_Dynamics/cylinder_flow**) demonstrates how to add and set up particle tracing in a plot group using the **Particle Tracing with Mass** node. It uses the predefined Khan-Richardson model for the drag force and neglects gravity and buoyancy forces.

References for the Single-Phase Flow, Laminar Flow Interfaces

1. G.G. Stokes, Trans. Camb. Phil. Soc., 8, 287-305, 1845
2. P.M. Gresho and R.L. Sani, *Incompressible Flow and the Finite Element Method, Volume 2: Isothermal Laminar Flow*, John Wiley & Sons, 2000.
3. G.K. Batchelor, *An Introduction To Fluid Dynamics*, Cambridge University Press, 1967.
4. R.L. Panton, *Incompressible Flow*, 2nd ed., John Wiley & Sons, 1996.
5. I. Harari and T.J.R. Hughes, "What are C and h ? Inequalities for the Analysis and Design of Finite Element Methods," *Comp. Meth. Appl. Mech. Engrg*, vol. 97, pp. 157–192, 1992.
6. Y. Bazilevs, V.M. Calo, T.E. Tezduyar, and T.J.R. Hughes, "YZ β Discontinuity Capturing for Advection-dominated Processes with Application to Arterial Drug Delivery," *Int.J.Num. Meth. Fluids*, vol. 54, pp. 593–608, 2007.
7. T.J.R. Hughes and M. Mallet, "A New Finite Element Formulation for Computational Fluid Dynamics: III. The Generalized Streamline Operator for

- Multidimensional Advective-Diffusive System,” *Comp. Meth. Appl. Mech. Engrg*, vol. 58, pp. 305–328, 1986.
8. G. Hauke and T.J.R. Hughes, “A Unified Approach to Compressible and Incompressible Flows,” *Comp. Meth. Appl. Mech. Engrg*, vol. 113, pp. 389–395, 1994.
 9. G. Hauke, “Simple Stabilizing Matrices for the Computation of Compressible Flows in Primitive Variables,” *Comp. Meth. Appl. Mech. Engrg*, vol. 190, pp. 6881–6893, 2001.
 10. D.J. Tritton, *Physical Fluid Dynamics*, 2nd ed., Oxford University Press, 1988.
 11. J.M. Coulson and J.F. Richardson, “Particle Technology and Separation Processes,” *Chemical Engineering, Volume 2*, Butterworth-Heinemann, 2002.
 12. J.L. Guermond, P. Minev, and J. Shen, “An overview of projection methods for incompressible flows,” *Comp. Meth. Appl. Mech. Engrg*, vol. 195, pp. 6011–6045, 2006.
 13. B. Rivière, *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations*, SIAM, 2008.
 14. Y. Epshteyn and B. Rivière, “Estimation of penalty parameters for symmetric interior penalty Galerkin methods,” *J. Computational and Applied Mathematics*, vol. 206, pp. 843–872, 2007.
 15. G. Kariadakis, A. Beskok, and N. Aluru, *Microflows and Nanoflows*, Springer Science and Business Media, 2005.
 16. R.P. Chhabra and J.F. Richardson, *Non-Newtonian Flow and Applied Rheology*, 2nd ed., Elsevier, 2008.



The Boltzmann Equation, Two-Term Approximation Interface

This chapter describes the Boltzmann Equation, Two-Term Approximation interface, found under the **Plasma** branch () when adding a physics interface.

In this chapter:

- [The Boltzmann Equation, Two-Term Approximation Interface](#)
- [Theory for the Boltzmann Equation, Two-Term Approximation Interface](#)

The Boltzmann Equation, Two-Term Approximation Interface

The **Boltzmann Equation, Two-Term Approximation (be)** interface () , found under the **Plasma** branch () when adding a physics interface, studies the electron energy distribution function (EEDF) and calculates it from a set of collision cross sections for some mean discharge conditions. The physics interface is different because the geometry and mesh are fixed and automatically generated when the physics interface is added to a model. The x -axis represents an energy coordinate rather than physical space, enabling sets of collision cross sections to be loaded and the properties of the distribution function set. Additionally, electron source coefficients and transport properties can be computed. The Boltzmann Equation, Two-Term Approximation interface is intended as a preprocessing step before solving space-dependent models. This physics interface is available for 1D components.

When this physics interface is added, these default nodes are also added to the **Model Builder: Boltzmann Model** and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions. You can also right-click **Boltzmann Equation, Two-Term Approximation** to select physics features from the context menu.



The [Reduced Electric Fields](#) study is available for this physics interface and described in the *COMSOL Multiphysics Reference Manual*.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (`_`) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `be`.

DOMAIN SELECTION



There is always only one domain for the physics interface. The position and length of the domain are automatically generated and rarely need to be changed.

CROSS SECTION IMPORT

Browse to a file containing a set of collision cross sections.



The file must follow the format documented in [Importing Collision Cross-Section Data](#).

ELECTRON ENERGY DISTRIBUTION FUNCTION

Select an **Electron energy distribution function** (EEDF)—**Maxwellian** (the default), **Druyvesteyn**, **Generalized**, or **Boltzmann**.

- Select **Maxwellian** to assume a Maxwellian EEDF, which takes the form:

$$f(\epsilon) = \phi^{-3/2} \beta_1 \exp(-(\epsilon \beta_2 / \phi))$$

where

$$\beta_1 = \Gamma(5/2)^{3/2} \Gamma(3/2)^{-5/2}, \beta_2 = \Gamma(5/2) \Gamma(3/2)^{-1}$$

and ϕ is the mean electron energy (eV), ϵ is the electron energy (eV) and Γ is the upper incomplete gamma function.

- Select **Druyvesteyn** to assume a Druyvesteyn EEDF, which takes the form:

$$f(\epsilon) = 2\phi^{-3/2} \beta_1 \exp(-(\epsilon \beta_2 / \phi)^2)$$

where

$$\beta_1 = \Gamma(5/4)^{3/2} \Gamma(3/4)^{-5/2}, \beta_2 = \Gamma(5/4) \Gamma(3/4)^{-1}$$

- Select **Generalized** for a generalized distribution function where the EEDF is somewhere between Maxwellian and Druyvesteyn. For **Generalized** enter a value for the **Power law** g (dimensionless). The default is 1, and this number should be between 1 and 2. Mathematically, the EEDF takes the form:

$$f(\varepsilon) = g\varphi^{-3/2}\beta_1 \exp(-(\varepsilon\beta_2/\varphi)^g)$$

where

$$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2}, \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}, 1 \leq g \leq 2$$

So, when $g = 1$ the distribution function is Maxwellian and when $g = 2$ it is Druyvesteyn.

- Select **Boltzmann** to solve a two-term approximation to the Boltzmann equation. When selected, the EEDF is computed from a partial differential equation instead of by taking an assumed function. The two-term Boltzmann equation is a complicated, nonlocal integral equation.

BOLTZMANN PROPERTIES

This section is available when **Boltzmann** is selected as the **Electron energy distribution function**. These settings also enable available settings on the **Settings** window for **Boltzmann Model**.

- Select **On** or **Off** (the default) from the **Electron-electron collisions** list. If the ionization degree of the discharge is high then set this property to **On**.
Electron-electron collisions tend to make the distribution function more Maxwellian and they also increase the complexity of the problem.
- Select an option from the **Secondary electron energy sharing** list—**Equal** (the default) or **Zero energy**.


This describes how the energy is split between two electrons when an ionization collision occurs. If set to **Equal**, then both electrons take an equal energy after the collision. If the property is set to **Zero energy** then the secondary electron created in an ionization collision has zero energy and the ionizing electron carries all the excess energy.


- Select **On** or **Off** (the default) from the **Oscillating field** list. Only select **On** if the reduced angular frequency of the discharge is high, which is typically only true for microwave discharges. The reduced angular frequency is the ratio of the angular frequency and the neutral number density, ω/N .


DEPENDENT VARIABLES

The dependent variable (field variable) for the **EEDF**. The name can be changed but the names of fields and dependent variables must be unique within a model.

DISCRETIZATION


To display this section, click the **Show** button () and select **Discretization**.

	<ul style="list-style-type: none">• Domain, Boundary, and Pair Nodes for the Boltzmann Equation, Two-Term Approximation Interface• Theory for the Boltzmann Equation, Two-Term Approximation Interface
---	---


	<i>Argon Boltzmann Analysis:</i> Application Library path Plasma_Module/Two-Term_Boltzmann_Equation/boltzmann_argon
---	--

Domain, Boundary, and Pair Nodes for the Boltzmann Equation, Two-Term Approximation Interface

The [Boltzmann Equation, Two-Term Approximation Interface](#) has these domain, boundary, and pair nodes, listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).

	In general, to add a node, go to the Physics toolbar, no matter what operating system you are using. Subnodes are available by clicking the parent node and selecting it from the Attributes menu.
---	--

- | | |
|---|--|
| <ul style="list-style-type: none">• Boltzmann Model• Boundary Settings• Collision | <ul style="list-style-type: none">• Initial Values• Zero Energy Flux• Zero Probability |
|---|--|

	In the <i>COMSOL Multiphysics Reference Manual</i> see Table 2-3 for links to common sections and Table 2-4 to common feature nodes. You can also search for information: press F1 to open the Help window or Ctrl+F1 to open the Documentation window.
---	---



This node is available when **Boltzmann** is selected as the [Electron Energy Distribution Function](#).

In the **Boltzmann Model** node, values for specific discharge parameters are set. The number of parameters needed depends on the property settings for the physics interface.

DOMAIN SELECTION



There is always only one domain for the physics interface. The position and length of the domain are automatically generated and rarely needs to be changed.

BOLTZMANN SETTINGS

- Enter a **Gas temperature** T_g (SI unit: K). The default is 300 K. The gas temperature influences the amount of energy loss that an electron suffers during an elastic collision. The higher the gas temperature, the lower the energy loss for an electron during an elastic collision.
- If the **Electron-electron collisions** property is set to **On** in the [Boltzmann Properties](#) section, enter an estimate for the **Electron density** n_e (SI unit: $1/\text{m}^3$).

There is a weak dependence on the electron density in [Equation 5-1](#) which accounts for the screening of the Coulomb potential by space charge effects ([Ref. 1](#)).

- If the **Electron-electron collisions** property is set to **On** in the [Boltzmann Properties](#) section, enter an **ionization degree** β (dimensionless).

The ionization degree can have a significant effect on the shape of the electron energy distribution function. When the ionization degree is very low, the tail of the distribution function tends to decrease very rapidly. As the ionization degree increases, the distribution function tends to become more Maxwellian, resulting in a greater proportion of electrons with higher energies. This can have a profound effect on the rate coefficients for ionization and excitation collisions because of their high energy threshold.

- If the **Oscillating field** property is set to **On** in the [Boltzmann Properties](#) section, enter a **Reduced angular frequency** ω/N (SI unit: m^3/s). The default is $10^{-13} \text{ m}^3/\text{s}$.

If the reduced angular frequency is high, the proportion of electrons with high energies is substantially increased for the same mean electron energy. This is because in DC fields, collisional momentum transfer impedes electrons acquiring higher energies but high frequency fields have the opposite effect.

MOLE FRACTION SETTINGS

Specify the mole fraction of each of the target species in the set of collision cross sections.

Since the sum of all the mole fractions must be equal to one, select the **Sum equals 1** check box to compute the mole fraction of the first species based on the mole fractions of all the other species. The **Mole fraction** of the first species cannot be edited when this option is selected.

Mole fraction. There are N text fields available for setting the mole fractions of each species, where N is the total number of target species. Enter the **Mole fraction** expected to be observed in the discharge for each target species. The mole fraction of electronically excited species can have a strong effect on the shape of the electron energy distribution function.

Initial Values



The **Initial Values** node is only available if **Boltzmann** is selected as the [Electron Energy Distribution Function](#). The **Initial Values** node allows you to set an initial guess for the mean electron energy, the Townsend reduced electric field, and the initial condition for a Lagrange multiplier.

DOMAIN SELECTION



There is always only one domain for the physics interface. The position and length of the domain are automatically generated and should not be changed.

INITIAL VALUES

- Enter an **Initial mean electron energy** e_0 (SI unit: V). The default is 2 V.

This value should be equal to the mean electron energy of the first parameter in the parametric sweep. COMSOL Multiphysics constructs an initial guess at the EEDF by assuming a Maxwellian distribution function with the specified mean electron energy.

- Enter an **Initial reduced electric field** E/N_0 (dimensionless). The default is 1[Td] and this rarely needs to be changed. This is the initial reduced electric field used to compute the distribution function.
- Enter an **Initial condition for lam** λ_0 (dimensionless). The default is 1e-27 and this rarely needs to be changed.

This value is an initial guess for a Lagrange multiplier, which is used to force the mean electron energy to be equal to the specified mean electron energy.

Collision

Use the **Collision** node to add collisions to a model by entering a formula and some properties of the collision. The cross-section data can be entered manually or by loading it from a file.



Load in multiple cross sections from a file in the **Boltzmann Equation, Two-Term Approximation** settings. In this case, the collision features are automatically generated.

DOMAIN SELECTION



There is always only one domain for the physics interface. The position and length of the domain are automatically generated and should not be changed.

COLLISION


Enter a **Formula** for the collision.

Select a **Collision type**: **Elastic** (the default), **Excitation**, **Attachment**, or **Ionization**.

- For **Elastic** enter a value for the **Electron mass ratio** m_r (dimensionless) (the ratio of the mass of the electron to the target species).

- Select **Excitation** for excitation reactions where the collision is inelastic. Enter the **Energy loss $\Delta\epsilon$** (SI unit: V) for the reaction. This is also known as threshold energy for the reaction. Select this option for electronic excitation, vibrational and rotational excitation, and superelastic collisions. For superelastic collisions, the activation energy is negative.
- Select **Attachment** for collisions which result in the electron attaching to the target species, forming a negative ion. No user input is required.
- Select **Ionization** for electron impact reactions which result in secondary electrons being formed. Enter the **Energy loss $\Delta\epsilon$** (SI unit: V) for the reaction. This is also known as threshold energy for the reaction.

CROSS SECTION DATA

Enter the cross section data in two columns. Enter the **Electron energy** (SI unit: eV) and the **Collision cross section data** (SI unit: m^2). Also click the **Load from file** button () to import data from a text file. The file must contain the two columns of data (**Electron energy** (SI unit: eV) in the first column and **Collision cross section data** (SI unit: m^2) in the second column).

Boundary Settings

The boundary conditions are always the same, **Zero energy flux** at $\epsilon = 0$ and **Zero probability** as $\epsilon \rightarrow \infty$.

Zero Energy Flux

The **Zero Energy Flux** boundary condition should always be applied at $x = 0$. Physically the boundary condition specifies that no electrons can have energy less than zero electron volts.

Zero Probability

The **Zero Probability** boundary condition should always be applied at $\epsilon = 100$. Physically the boundary condition states that there are no electrons with infinite energy. Instead of solving the problem in an infinite domain, the maximum value of ϵ is chosen to be large enough so it does not affect the results. The default is $\epsilon = 100$.

Theory for the Boltzmann Equation, Two-Term Approximation Interface

The Boltzmann Equation, Two-Term Approximation Interface theory is described in this section:

- [The Electron Energy Distribution Function](#)
- [Boltzmann Equation, Two-Term Approximation](#)
- [References for the Boltzmann Equation, Two-Term Approximation Interface](#)

The Electron Energy Distribution Function

For computing the electron density and electron energy according to [Equation 6-5](#) and [Equation 6-6](#) transport properties and source coefficients are required. These coefficients can be calculated from collision cross-section data and the (unknown) *electron energy distribution function* (EEDF). The equation formulation used in the physics interface depends on whether the EEDF is an assumed function (Maxwellian, Druyvesteyn or Generalized) or is to be explicitly computed by the Boltzmann equation, using the two-term approximation.

In order to remain consistent with [Ref. 1](#), the unit of electron energy and mean electron energy is specified in volts, rather than electron volts. The two quantities can be converted to each other via the unit charge, which is the built-in variable `e_const` in COMSOL. So, the mean electron energy (in volts) is equal to the mean electron energy (in eV) divided by the unit charge.

Obtaining a physical solution for a fluid model of a plasma largely depends on how self-consistently the transport properties and source coefficients are calculated. By solving the two-term Boltzmann equation, maximum consistency is obtained between the kinetic and fluid description of the electron dynamics.

Solving the *Boltzmann equation* also makes it possible to calculate the drift velocity. The drift velocity is an important quantity because it depends on all the collisions which make up the plasma chemistry. The drift velocity is easy to measure, and comparing the experimental and simulated drift velocity is therefore straightforward. Indeed, collision cross sections are often inferred from drift velocity data.

Verifying that the experimental drift velocity agrees with the computed drift velocity is a good indicator that a given set of cross sections are correct. This physics interface makes it easy to compare the computed drift velocity to experimental data. A treatise on experimental measurements of the drift velocity is available in [Ref. 2](#).

Consider the following expression

$$f(\epsilon) = g\varphi^{-3/2}\beta_1 \exp(-(\epsilon\beta_2/\varphi)^g)$$

where the following definitions apply

$$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2}, \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}, 1 \leq g \leq 2$$

and φ is the mean electron energy. Depending on the value of the variable, g , the distribution function is either *Maxwellian* ($g = 1$), *Druyvesteyn* ($g = 2$), or *Generalized* ($1 < g < 2$). Mathematically, the distribution functions are defined as:

MAXWELLIAN EEDF

$$f(\epsilon) = \varphi^{-3/2}\beta_1 \exp(-(\epsilon\beta_2/\varphi))$$

$$\beta_1 = \Gamma(5/2)^{3/2}\Gamma(3/2)^{-5/2}, \beta_2 = \Gamma(5/2)\Gamma(3/2)^{-1}$$

DRUYVESTeyN EEDF

$$f(\epsilon) = 2\varphi^{-3/2}\beta_1 \exp(-(\epsilon\beta_2/\varphi)^2)$$

$$\beta_1 = \Gamma(5/4)^{3/2}\Gamma(3/4)^{-5/2}, \beta_2 = \Gamma(5/4)\Gamma(3/4)^{-1}$$

GENERALIZED EEDF

$$f(\epsilon) = g\varphi^{-3/2}\beta_1 \exp(-(\epsilon\beta_2/\varphi)^g)$$

$$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2}, \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}, 1 \leq g \leq 2$$

Boltzmann Equation, Two-Term Approximation

The Boltzmann equation describes the evolution of a distribution function, f , in six-dimensional phase space:

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f - \frac{e}{m} (\mathbf{E} \cdot \nabla_{\mathbf{v}} f) = C[f].$$

To be able to solve the Boltzmann equation and thus compute the EEDF, drastic simplifications are necessary. A common approach is to expand the distribution function in spherical harmonics. For high precision, six or more terms might be needed. If the EEDF can be assumed almost spherically symmetric, the series can be truncated after the second term. The EEDF becomes symmetric because in elastic collisions with neutral atoms, the direction of motion of the electrons is changed and their loss of energy is small due to large mass difference. The two-term approximation is valid for all values of reduced electric field (the ratio of the electric field to the number density of the background gas) if inelastic scattering can be neglected. There is no minimum in the cross section, and the cross section for momentum transfer is related to the electron energy by a power-law dependence (Ref. 3).

When the Boltzmann equation option is selected, it becomes necessary to solve a 1D reaction/convection/diffusion equation to compute the EEDF. The equation is somewhat special because the source term is nonlocal and the convection and diffusion coefficients depend on the integral of the solution. The stationary two-term Boltzmann equation, including the effects of *electron-electron collisions* is:

$$\frac{\partial}{\partial \epsilon} \left(Wf - D \frac{\partial f}{\partial \epsilon} \right) = S$$

where

$$W = -\gamma \epsilon^2 \sigma_\epsilon - 3a \left(\frac{n_e}{N_n} \right) A_1$$

$$D = \frac{\gamma}{3} \left(\frac{E}{N_n} \right)^2 \left(\frac{\epsilon}{\sigma_m} \right) + \frac{\gamma k_b T}{q} \epsilon^2 \sigma_\epsilon + 2a \left(\frac{n_e}{N_n} \right) (A_2 + \epsilon^{3/2} A_3)$$

The following definitions apply

$$a = \frac{q^2 \gamma}{24 \pi \epsilon_0^2} \ln \Lambda$$

$$\Lambda = \frac{12 \pi (2 \epsilon_0 q \bar{\epsilon} / 3)^{3/2}}{q^3 n_e^{1/2}} \quad (5-1)$$

$$\sigma_m = \sum_{k = \text{all}} x_k \sigma_k$$

$$\sigma_\varepsilon = \sum_{k = \text{elastic}} 2 \left(\frac{m_e}{M} \right) x_k \sigma_k$$

$$\tilde{\sigma}_m = \sigma_m + \frac{\lambda}{\varepsilon^{1/2}}$$

$$A_1 = \int_0^\varepsilon u^{1/2} f(u) du$$

$$A_2 = \int_0^\varepsilon u^{3/2} f(u) du$$

$$A_3 = \int_\varepsilon^\infty f(u) du$$

Here:

- $\gamma = (2q/m_e)^{1/2}$ (SI unit: $C^{1/2}/kg^{1/2}$)
- m_e is the electron mass (SI unit: kg)
- ε is energy (SI unit: V)
- σ_ε is the total elastic collision cross section (SI unit: m^2)
- σ_m is the total collision cross section (SI unit: m^2)
- q is the electron charge (SI unit: C)
- ε_0 is the permittivity of free space (SI unit: F/m)
- T is the temperature of the background gas (SI unit: K)
- k_b is the Boltzmann constant (SI unit: J/K)
- n_e is the electron density (SI unit: $1/m^3$)
- N_n is the background gas density (SI unit: $1/m^3$)
- Λ is the Coulomb logarithm, and
- M is the mass of the target species (SI unit: kg).

The source term, S represents energy loss due to inelastic collisions. Because the energy loss due to an inelastic collision is quantized, the source term is nonlocal in

energy space. The source term can be decomposed into four parts where the following definitions apply:

$$\begin{aligned}
S &= \sum_{k = \text{inelastic}} C_k - \gamma \lambda \varepsilon^{1/2} f \\
C_{k = \text{excitation}} &= -\gamma x_k [\varepsilon \sigma_k(\varepsilon) f(\varepsilon) - (\varepsilon + \Delta \varepsilon_k) \sigma_k(\varepsilon + \Delta \varepsilon_k) f(\varepsilon + \Delta \varepsilon_k)] \\
C_{k = \text{att}} &= -\gamma x_k \varepsilon \sigma_k(\varepsilon) f(\varepsilon) \\
C_{\text{ion}} &= -\gamma x_k [\varepsilon \sigma_k(\varepsilon) f(\varepsilon) - (\varepsilon + \Delta \varepsilon_k) \sigma_k(\varepsilon + \Delta \varepsilon_k) f(\varepsilon + \Delta \varepsilon_k)] \\
&\quad + \delta \gamma x_k \int_0^\infty u \sigma_k(u) f(u) du
\end{aligned}$$

where x_k is the mole fraction of the target species for reaction k , σ_k is the collision cross section for reaction k , $\Delta \varepsilon_k$ is the energy loss from collision k , and δ is the delta function at $\varepsilon = 0$. The term, λ is a scalar valued renormalization factor which ensures that the EEDF has the following property:

$$\int_0^\infty f \varepsilon^{1/2} d\varepsilon = 1 \quad (5-2)$$

An ODE is implemented to solve for the value of λ such that [Equation 5-2](#) is satisfied. The rate coefficients are computed from the EEDF by way of the following integral

$$k_k = \gamma \int_0^\infty \varepsilon \sigma_k(\varepsilon) f(\varepsilon) d\varepsilon$$

The mean electron energy is defined by the integral

$$\bar{\varepsilon} = \int_0^\infty f \varepsilon^{3/2} d\varepsilon \quad (5-3)$$

In order to set the mean electron energy to a specific setpoint, a second Lagrange multiplier is introduced to solve for the reduced electric field, such that [Equation 5-3](#) is satisfied. The weak form of the constraint is:

$$\left(\int_0^\infty f \varepsilon^{3/2} d\varepsilon - \varepsilon_{sp} \right) \left(\frac{\bar{E}}{N_n} \right) = 0$$

where tilde denotes test function. The drift velocity is computed from the following integral:

$$w = -\left(\frac{\gamma}{3}\right) \left(\frac{E}{N_n}\right) \int_0^\infty \frac{\varepsilon}{\sigma_m} \left(\frac{\partial f}{\partial \varepsilon}\right) d\varepsilon$$

The *drift velocity* is an important quantity for two reasons. Firstly, it provides a convenient way of comparing the results of the Boltzmann equation to experimental data. The second reason is that it allows the *Townsend coefficients* to be computed using:

$$\alpha_k = \frac{k_k}{w}$$

When the distribution function is assumed, the reduced electric field must be computed in a completely different manner. The reduced electric field is not required to compute the rate coefficients or transport properties. It only needs to be calculated so that the Townsend coefficients and the drift velocity can be computed. The following expression is defined:

$$\left(\frac{E}{N_n}\right) = \sqrt{\frac{\Upsilon}{\mu_e N_n}}$$

where

$$\Upsilon = \sum_k x_k k_k \Delta \varepsilon_k$$

and $\Delta \varepsilon_k$ is the energy loss for collision k . If the collision is elastic, the energy loss is defined as:

$$\Delta \varepsilon_k = 2\tilde{\varepsilon} \left(\frac{m_e}{M}\right)$$

A plot of the drift velocity for different distribution functions versus the reduced electric field for oxygen is shown in [Figure 5-1](#). Experimental data is also included in the plot.

The *reduced transport properties* are computed using the following integrals

$$\mu_e N_n = -\left(\frac{\gamma}{3}\right) \int_0^\infty \frac{\varepsilon}{\sigma_m} \left(\frac{\partial f}{\partial \varepsilon}\right) d\varepsilon$$

$$D_e N_n = \left(\frac{\gamma}{3} \right) \int_0^\infty \frac{\varepsilon}{\sigma_m} f d\varepsilon$$

$$\mu_\varepsilon N_n = - \left(\frac{\gamma}{3\varepsilon} \right) \int_0^\infty \frac{\varepsilon^2}{\sigma_m} \left(\frac{\partial f}{\partial \varepsilon} \right) d\varepsilon$$

$$D_\varepsilon N_n = \left(\frac{\gamma}{3\varepsilon} \right) \int_0^\infty \frac{\varepsilon^2}{\sigma_m} f d\varepsilon$$

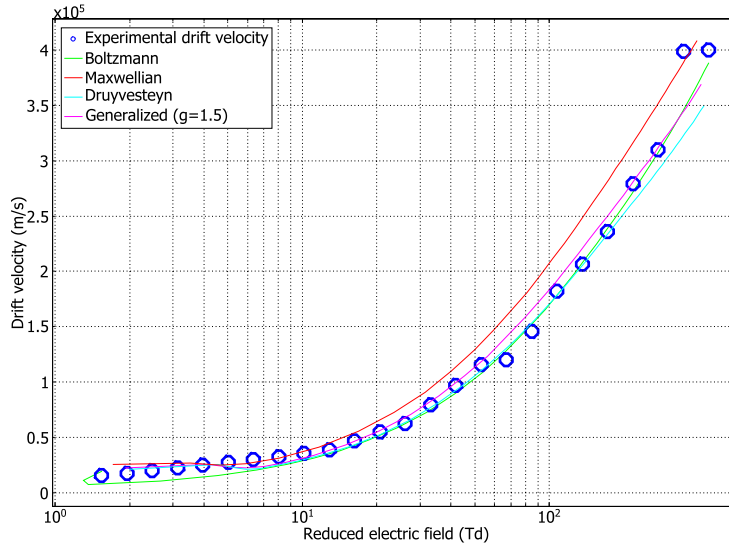


Figure 5-1: Plot of experimental and computed drift velocity for different distribution functions.

The computed transport coefficients have little dependence on the type of EEDF. However, the rate coefficients for excitation and ionization processes are highly dependent on the shape of the EEDF, due to the exponential drop off in the population of electrons at energies exceeding the activation threshold. [Figure 5-2](#) plots the ionization rate coefficient for oxygen for the types of distribution function.

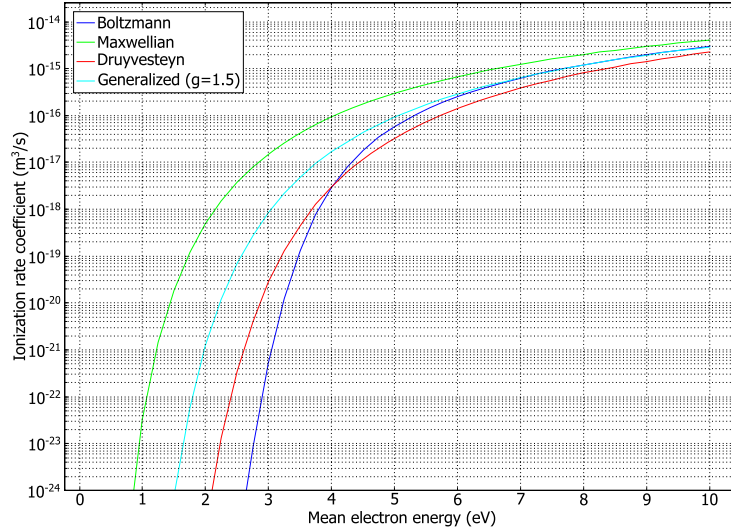



Figure 5-2: Plot of ionization coefficient vs. mean electron energy for different distribution functions.

References for the Boltzmann Equation, Two-Term Approximation Interface

1. G.J.M. Hagelaar and L. C. Pitchford, "Solving the Boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models," *Plasma Sources Science and Technology*, vol. 14, pp. 722–733, 2005.
2. J. Dutton, "A Survey of Electron Swarm Data," *J. Phys. Chem. Ref. Data*, vol. 4, pp. 577–866, 1975.
3. H.B. Milloy and R.O. Watts, "On the Validity of the Two-term Approximation in the Solution of Boltzmann's Equation for Electron Motion" *Aust. J. Phys.*, vol 30, pp 73, 1977.



The Drift Diffusion Interface

This chapter describes the underlying electron transport theory for the Drift Diffusion interface, found under the **Plasma** branch () when adding a physics interface.

In this chapter:

- [The Drift Diffusion Interface](#)
- [Theory for the Drift Diffusion Interface](#)

The Drift Diffusion Interface

The **Drift Diffusion (dd)** interface () , found under the **Plasma** branch () when adding a physics interface, solves for the electron density and mean electron energy for any type of plasma. A wide range of boundary conditions are available to handle secondary emission, thermionic emission, and wall losses.

When this physics interface is added, these default nodes are also added to the **Model Builder—Drift Diffusion Model**, **Insulation** (the default boundary condition), and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and velocity. You can also right-click **Drift Diffusion** to select physics features from the context menu.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (`_`) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `dd`.

ELECTRON PROPERTIES

The following check boxes are available—**Use reduced electron transport properties**, **Compute tensor electron transport properties**, **Include thermal diffusion**, or **Fix mean electron energy**.

Use Reduced Electron Transport Properties

Select the **Use reduced electron transport properties** check box to specify the electron mobility, diffusivity, energy mobility and energy diffusivity in reduced form. The neutral number density is then specified in the **Drift Diffusion Model** node. The electron transport properties are computed from the reduced transport properties using:

$$\mu_e = \frac{\mu_{\text{red}}}{N_n}$$

where N_n is the user-defined neutral number density.



It is sometimes more convenient to use reduced transport properties, since they can be output from [The Boltzmann Equation, Two-Term Approximation Interface](#).

Compute Tensor Electron Transport Properties

Select the **Compute tensor electron transport properties** check box to automatically compute the tensor form of the electron mobility, diffusivity, energy mobility and energy diffusivity when a static magnetic field is present. In this case the magnetic flux density is specified in the [Drift Diffusion Model](#) feature.

Include Thermal Diffusion

Select the **Include thermal diffusion** check box to add an additional term to the definition of the electron current due to gradients in the electron diffusivity. If the diffusivity is a constant then including this does not effect the solution. It is only necessary to include this term if the electron diffusivity is a function of the electron temperature, and there are significant gradients in the electron temperature.

Fix Mean Electron Energy

Select the **Fix mean electron energy** check box to fix the mean electron energy to its initial value. This can be useful in some situations because the strong coupling between the mean electron energy and the electromagnetic fields is removed. This allows for non-self consistent models to be created quickly, since problems where the mean electron energy is fixed are easier to solve numerically.

STABILIZATION

If the **Equation formulation** is set to **Log** then the solver can run into difficulties as the electron density or electron energy density approach zero. The **Source stabilization** check box (selected by default) adds an additional source term to the equation for the electron density and electron energy density. In the ζ field, enter a tuning parameter for the source stabilization. The default value is 1. This value is usually good enough. If the plasma is high pressure (atmospheric) then it can help to lower this number to somewhere in the range of 0.25-0.5. For information on stabilization see [Stabilization](#).

ADVANCED SETTINGS


Use the **Equation formulation** to chose whether to solve the drift-diffusion equations in log or linear form. The default is set to log which means that COMSOL Multiphysics



solves for the log of the electron density and electron energy density. This is generally more numerically stable than the linear form of the equations.

DEPENDENT VARIABLES

The dependent variables (field variables) for the **Electron density** and **Electron energy density**. The name can be changed but the names of fields and dependent variables must be unique within a model.



DISCRETIZATION

To display this section, click the **Show** button () and select **Discretization**.

	<ul style="list-style-type: none">• Domain, Boundary, and Pair Nodes for the Drift Diffusion Interface• Stabilization• Theory for the Drift Diffusion Interface
	<i>Drift Diffusion Tutorial:</i> Application Library path Plasma_Module/Direct_Current_Discharges/drift_diffusion_tutorial

Domain, Boundary, and Pair Nodes for the Drift Diffusion Interface

The [Drift Diffusion Interface](#) has these domain, boundary, and pair nodes, listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).

	In general, to add a node, go to the Physics toolbar, no matter what operating system you are using. Subnodes are available by clicking the parent node and selecting it from the Attributes menu.
	In the <i>COMSOL Multiphysics Reference Manual</i> see Table 2-3 for links to common sections and Table 2-4 to common feature nodes. You can also search for information: press F1 to open the Help window or Ctrl+F1 to open the Documentation window.



The boundary conditions are used for the electron density and mean electron energy, which make it easy to specify how the electrons interact with the wall. Secondary emission, thermionic emission and reflection can all be specified.

It is also possible to impose the specific value of the electron density or mean electron energy on a boundary. This, however is not recommended as imposing such a value on a solid surface tends not to fit the physics of the interaction between a solid surface and electrons.

- [Drift Diffusion Model](#)
- [Initial Values](#)
- [Electron Density and Energy](#)
- [Electron Outlet](#)
- [Electron Production Rate](#)
- [Flux](#)
- [General Power Deposition](#)
- [Inductive Power Deposition](#)
- [Insulation](#)
- [Microwave Power Deposition](#)
- [Velocity](#)
- [Wall](#)

Drift Diffusion Model

The **Drift Diffusion Model** adds the equations for electron transport in a plasma. Specify expressions for the electron diffusivity, mobility, electron energy diffusivity and electron energy mobility. There are also model inputs for the electric potential and collisional power loss.

MODEL INPUTS

This section contain fields and values that are inputs to expressions that define material properties. If such user-defined property groups are added, the model inputs are listed here.

- **Electric potential** V (SI unit: V). Used to compute the electric field which results in migration of the electrons.
- **Collisional power loss** S_{en} (SI unit: W/m^3). Represents the energy loss due to elastic and inelastic collisions.

- When the [Use Reduced Electron Transport Properties](#) check box is selected, chose **Neutral number density** N_n (SI unit: $1/\text{m}^3$) to compute the electron transport.
- When the [Compute Tensor Electron Transport Properties](#) check box is selected, chose **Magnetic flux density** B (SI unit: T) to compute the tensor form of the electron transport properties when a static magnetic field is present.

The electric potential typically comes from an **Electrostatics** interface and the collisional power loss is typically defined by [The Heavy Species Transport Interface](#).

ELECTRON DENSITY AND ENERGY

This section is not available if the [Compute Tensor Electron Transport Properties](#) check box is selected.

For all other choices, select an option from the **Electron transport properties** list—[Specify Mobility Only](#) (the default), [Specify All](#), or [Use Lookup Tables](#).



The other transport properties are computed using Einstein's relation, see [Tensor Transport Properties](#) for more details.

Specify Mobility Only

When no check boxes, or the [Include Thermal Diffusion](#) or [Fix Mean Electron Energy](#) check boxes are selected, and if **Specify mobility only** is selected, enter a value or expression for the **Electron mobility** μ_e (SI unit: $\text{m}^2/(\text{V}\cdot\text{s})$). Select **Isotropic**, **Diagonal**, **Symmetric** or **Anisotropic** based on the model. The electron diffusivity, energy mobility and energy diffusivity are automatically computed using Einstein's relation for a Maxwellian EEDF:

$$\mathbf{D}_e = \mu_e T_e, \mu_\epsilon = \left(\frac{5}{3}\right) \mu_e, \mathbf{D}_\epsilon = \mu_\epsilon T_e \quad (6-1)$$

If the [Use Reduced Electron Transport Properties](#) check box is selected, enter the **Reduced electron mobility** $\mu_e N_n$ (SI unit: $1/(\text{V}\cdot\text{m}\cdot\text{s})$). The actual electron mobility is then computed by dividing the reduced electron mobility by the neutral number density. The electron diffusivity, energy mobility and energy diffusivity are then computed as in [Equation 6-1](#).

Specify All

For **Specify all** enter values or expressions for all of the properties, which can be either scalars or tensors. The appropriate values for the transport properties can be computed with [The Boltzmann Equation, Two-Term Approximation Interface](#). For all of the

properties, select **Isotropic**, **Diagonal**, **Symmetric** or **Anisotropic** from the list based on the model, then enter values or expressions for all of these properties:

- **Electron mobility** μ_e (SI unit: $\text{m}^2/(\text{V}\cdot\text{s})$)
- **Electron diffusivity** D_e (SI unit: m^2/s)
- **Electron energy diffusivity** D_{en} (SI unit: m^2/s)
- **Electron energy mobility** μ_{en} (SI unit: $\text{m}^2/(\text{V}\cdot\text{s})$)

If the [Use Reduced Electron Transport Properties](#) check box is selected, then enter:

- **Reduced electron mobility** $\mu_e N_n$ (SI unit: $1/(\text{V}\cdot\text{m}\cdot\text{s})$)
- **Reduced electron diffusivity** $D_e N_n$ (SI unit: $1/(\text{m}\cdot\text{s})$)
- **Reduced electron energy diffusivity** $D_{en} N_n$ (SI unit: $1/(\text{m}\cdot\text{s})$)
- **Reduced electron energy diffusivity** $\mu_{en} N_n$ (SI unit: $1/(\text{V}\cdot\text{m}\cdot\text{s})$)

Use Lookup Tables

For **Use lookup tables** enter or load a lookup table with the transport properties as listed above versus mean electron energy (eV).



If the [Use Reduced Electron Transport Properties](#) check box is selected, the appropriate values for the transport properties can be computed with [The Boltzmann Equation, Two-Term Approximation Interface](#) and written out to a text file. The text file can then be loaded in for each electron transport property.

DC ELECTRON MOBILITY

This section is available if the [Compute Tensor Electron Transport Properties](#) check box is selected. Enter a value for the **DC electron mobility**, μ_{dc} (SI unit: $\text{m}^2/(\text{V}\cdot\text{s})$).

The section is also available when both the [Compute Tensor Electron Transport Properties](#) and the [Use Reduced Electron Transport Properties](#) check boxes are chosen. Then enter a value or expression for the **Reduced DC electron mobility**, $\mu_{dc} N_n$ (SI unit: $1/(\text{V}\cdot\text{m}\cdot\text{s})$).

Initial Values

The **Initial Values** node adds initial values for the electron density and mean electron energy.

INITIAL VALUES

Enter values or expressions for the initial value of the **Initial electron density** $n_{e,0}$ (SI unit: $1/\text{m}^3$) and **Initial mean electron energy** e_0 (SI unit: V). Enter an approximate guess for the mean electron energy in the reactor. COMSOL Multiphysics automatically computes the exact mean electron energy after solving but the closer the guess can be set to the expected value the easier it is to find a solution.

Electron Production Rate

Use the **Electron Production Rate** node to add sources and sinks to the number density of electrons. Ionization processes lead to an increase in the number density of electrons. Similarly, attachment leads to a decrease in the number density of electrons. A production rate for new electrons can be specified with this feature.

ELECTRON PRODUCTION RATE

Select or enter a value for a **Production rate** R_e (SI unit: $1/\text{m}^3\cdot\text{s}$) of new electrons. The value typically comes from [The Heavy Species Transport Interface](#).

General Power Deposition

Use the **General Power Deposition** to add an arbitrary heat source to the electrons.

GENERAL POWER DEPOSITION

Enter a value or expression for the **General heat source** Q (SI unit: W/m^3). This could be a constant number or come from another physics interface.

Velocity

Use the **Velocity** node to include the effect of convection due to a moving gas. In nearly all practical situations, convection of electrons due to gas movement is negligible.

VELOCITY

The **Velocity field** μ (SI unit: m/s) value or expression can be user-defined or come from another physics interface, typically **Laminar Flow**.

Inductive Power Deposition



The **Inductive Power Deposition** node requires a license for the AC/DC Module. Use it to add a heat source to the electrons from induction currents. This is typically used in inductively coupled plasmas.

INDUCTIVE POWER DEPOSITION

Enter a **Heat source** Q (SI unit: W/m^3) for the electrons due to induction currents. Typically this comes from a **Magnetic Fields** interface.

Microwave Power Deposition



The **Microwave Power Deposition** node requires a license for the RF Module. Use it to add microwave heating to the electrons. This is typically used in wave heated discharges.

MICROWAVE POWER DEPOSITION

Enter a **Heat source** Q (SI unit: W/m^3) for the electrons due to electromagnetic waves. This typically comes from an **Electromagnetic Waves** interface.

Insulation

Use the **Insulation** boundary condition for boundaries that correspond to inlets or outlets for the background gas and it is the default boundary condition for the **Drift Diffusion** interface. It is also available on the **Pairs** menu. This boundary condition requires no user input. The boundary condition sets the normal component of the electron and electron energy flux to zero:

$$-\mathbf{n} \cdot \Gamma_e = 0 \text{ and } -\mathbf{n} \cdot \Gamma_e = 0$$

Electron Density and Energy

The **Electron Density and Energy** boundary condition fixes the electron density to a specific number density or the mean electron energy to a specific energy. It is also available on the **Pairs** menu. This boundary condition should, in general, be avoided in favor of the **Wall** boundary condition.

$$n_e = n_{ew} \text{ and } \bar{\epsilon} = \bar{\epsilon}_0$$

Due to the fact that the log of the electron density is solved for, specifying an electron density of zero is not allowed.



For background theory see [The Electron Density and Energy Boundary Condition](#).

ELECTRON DENSITY AND ENERGY

Select the **Fix electron density** check box to enter a value for the **Electron density at the wall** n_{ew} (SI unit: $1/\text{m}^3$).

Select the **Fix mean electron energy** check box to enter a value for the **Mean electron energy at the wall** e (SI unit: V).

Wall

Use the **Wall** boundary condition when the plasma is in contact with a solid surface. This also allows for secondary and thermionic emission effects from the solid surfaces. Use the **Wall** boundary condition to describe how the electrons interact with the wall

GENERAL WALL SETTINGS

Enter a value for the **Reflection coefficient** r_e (dimensionless) on the selected boundaries. This must be a number between 0 and 1.

The **Include migration effects** check box should be checked if the electron flux to the wall is due to the electric field. This is usually not the case since the plasma potential is positive with respect to the walls so the electron flux to the wall due to the electric field is zero. In models where the electrostatic field changes very rapidly (like with dielectric barrier discharges for example) the electron flux to the wall can be due to the electric in which case this option should be activated.

ELECTRON DENSITY WALL SETTINGS

The **Use wall for electron density** check box is selected by default and deactivates the wall boundary condition for the electron density only. When activated, the flux of electrons lost to the wall is self consistently computed based on the electron temperature and electron density adjacent to the wall. When this is deactivated, the flux term is no longer applied for the corresponding equation, and the software will impose a zero flux boundary condition for that dependent variable.

Select or enter a value for the **Secondary emission flux** (SI unit: $1/\text{m}^2\cdot\text{s}$). The secondary emission flux can either be a user-defined expression or come from another physics interface, typically [The Heavy Species Transport Interface](#).

Select or enter a value for the **Thermal emission flux** (SI unit: $1/\text{m}^2\cdot\text{s}$). If electrons are being emitted from the wall due to thermal emission then enter an expression for the thermionic flux. Typically this only occurs when the wall is at a very high temperature.

ELECTRON ENERGY WALL SETTINGS

The **Use wall for electron energy** check box is selected by default and deactivates the wall boundary condition for the electron energy. This is useful in order to use the wall condition for the electron density but impose the value of the mean electron energy on the same boundaries.

Select or enter a value for the **Secondary emission energy flux**, which can either be a user-defined expression or come from another physics interface, typically [The Heavy Species Transport Interface](#).

Select or enter a value for the **Mean thermionic energy**. If electrons are being emitted from the wall due to thermionic emission then enter an expression for the mean energy of the emitted electrons. The mean electron energy of the emitted electrons is usually a function of the work function of the surface.



For background theory, see [The Wall Boundary Condition](#).

Flux

Use the **Flux** node to select whether the boundary condition is imposed for electron density or electron energy or both. The wall boundary condition should be used in preference to the flux boundary conditions for walls. A user-defined electron **Flux** can be specified by the following:

$$-\mathbf{n} \cdot \mathbf{\Gamma}_e = \Gamma_{e, \text{in}} \quad \text{and} \quad -\mathbf{n} \cdot \mathbf{\Gamma}_\epsilon = \Gamma_{\epsilon, \text{in}}$$

FLUX

Select the **Use flux for electron density** check box to activate the boundary condition for electron density. Then enter a value for the **Electron density influx** r_e (SI unit: $1/(\text{m}^2\cdot\text{s})$).

Select the **Use flux for electron energy density** to activate the boundary condition for electron energy. Then enter a value for the **Electron energy density influx** r_{en} (dimensionless).

Electron Outlet

Use the **Electron Outlet** boundary condition for boundaries which correspond to outlets for the background gas. This boundary condition requires no user input. The boundary condition sets the normal component of the electron and electron energy flux to zero:

$$-\mathbf{n} \cdot \Gamma_e = 0 \text{ and } -\mathbf{n} \cdot \Gamma_\epsilon = 0$$

Theory for the Drift Diffusion Interface

The Drift Diffusion Interface theory is described in this section:

- [Introduction to Drift Diffusion Theory](#)
- [Electron Transport Theory](#)
- [Source Coefficients](#)
- [Stabilization](#)
- [Tensor Transport Properties](#)
- [The Wall Boundary Condition](#)
- [The Insulation Boundary Condition](#)
- [The Electron Density and Energy Boundary Condition](#)
- [Reference for the Drift Diffusion Interface](#)

Introduction to Drift Diffusion Theory

In general, electron transport is described by the Boltzmann equation, which is a non-local continuity equation in phase space (\mathbf{r}, \mathbf{u}) . The Boltzmann equation is an extremely complicated integrodifferential equation and solving it in an efficient manner is not currently possible. The Boltzmann equation can be approximated by fluid equations by multiplying by a weighting function and then integrating over velocity space. This reduces the governing equations to a three-dimensional, time-dependent problem. The fluid equations describe the electron number density, the mean electron momentum and the mean electron energy as a function of configuration space and time. The rate of change of the electron density is described by:

$$\frac{\partial}{\partial t}(n_e) + \nabla \cdot \Gamma_e = R_e \quad (6-2)$$

where n_e is the electron density, Γ_e is the electron flux vector and R_e is either a source or a sink of electrons. The rate of change of the electron momentum is described by:

$$\frac{\partial}{\partial t}(n_e m_e \mathbf{u}_e) + \nabla \cdot n_e m_e \mathbf{u}_e \mathbf{u}_e^T = -(\nabla \cdot \mathbf{p}_e) + q n_e \mathbf{E} - n_e m_e \mathbf{u}_e \mathbf{v}_m \quad (6-3)$$

where m_e is the electron mass (SI unit: kg), \mathbf{u}_e is the drift velocity of the electrons (SI unit: m/s), \mathbf{p}_e is the electron pressure tensor (SI unit: Pa), q is the electron charge (SI unit: s A), \mathbf{E} is the electric field (SI unit: V/m) and ν_m is the momentum transfer frequency (SI unit: 1/s). The rate of change of the electron energy density is described by:

$$\frac{\partial}{\partial t}(n_e) + \nabla \cdot \mathbf{\Gamma}_e + \mathbf{E} \cdot \mathbf{\Gamma}_e = S_{\text{en}} \quad (6-4)$$

where n_e is the electron energy density (V/m³) and S_{en} is the energy loss or gain due to inelastic collisions (V/m³ s).

The expression for the flux, $\mathbf{\Gamma}_e$ in Equation 6-2 is derived from the momentum conservation, Equation 6-3. Under the assumptions that the ionization and attachment frequencies and also the angular frequency are much less than the momentum transfer frequency, the first term on the left hand side of Equation 6-3 can be neglected. Under the assumption that the electron drift velocity is smaller than the thermal velocity, the second term on the left hand side of Equation 6-3 can also be neglected. For a Maxwellian distribution the pressure term, \mathbf{p}_e can be replaced using the equation of state:

$$\mathbf{p}_e = n_e k_B T_e \mathbf{I}$$

where \mathbf{I} is the identity matrix and T_e is the electron temperature. With these assumptions an expression for the electron drift velocity can be derived:

$$\mathbf{u}_e = -\frac{k_B}{m_e \nu_m} \nabla T_e - \frac{k_B T_e}{n_e m_e \nu_m} \nabla n_e + \frac{q}{m_e \nu_m} \mathbf{E}.$$

Defining the electron flux as:

$$\mathbf{\Gamma}_e = n_e \mathbf{u}_e = -(\mu_e \bullet \mathbf{E}) n_e - \nabla (D_e n_e)$$

where the electron mobility, μ_e (SI unit: m²/(V s)) is defined as:

$$\mu_e = \frac{q}{m_e \nu_m}$$

and the electron diffusivity, D_e (SI unit: m²/s) is defined as:

$$D_e = \frac{k_B T_e}{m_e \nu_m}$$

The drift-diffusion approximation is suitable when the gas pressure is above about 1 mtorr (0.133 Pa) and the reduced electric field is not too high (typically less than 500 Townsends). Furthermore, the number density of charged species should be much less than the number density of the background gas. In other words, the discharge must be *weakly ionized*. The plasma must also be *collisional* which means that the mean free path between electrons and the background gas must be much less than the characteristic dimension of the system. This means the drift diffusion approximation is not suitable for modeling fusion plasmas.

The Drift Diffusion interface solves one equation for the electron density and one for the electron energy density.



The Drift Diffusion interface requires specification of the electron transport properties which are typically computed using [The Boltzmann Equation, Two-Term Approximation Interface](#).

Electron Transport Theory

The equation for the electron density is given by

$$\frac{\partial}{\partial t}(n_e) + \nabla \cdot \mathbf{\Gamma}_e = R_e - (\mathbf{u} \cdot \nabla)n_e \quad (6-5)$$

$$\mathbf{\Gamma}_e = -(\mu_e \bullet \mathbf{E})n_e - \nabla(D_e n_e)$$

Here (the subscript e refers to an electron:

- n_e denotes the electron density (SI unit: $1/\text{m}^3$)
- R_e is the electron rate expression (SI unit: $1/(\text{m}^3 \cdot \text{s})$)
- μ_e is the electron mobility which is either a scalar or tensor (SI unit: $\text{m}^2/(\text{V} \cdot \text{s})$)
- \mathbf{E} is the electric field (SI unit: V/m)
- D_e is the electron diffusivity which is either a scalar or a tensor (SI unit: m^2/s), and
- \mathbf{u} is the neutral fluid velocity vector (SI unit: m/s). This usually comes from a Navier-Stokes interface and in the case of electrons is almost always negligible.

The *electron energy density* is given by the following equation

$$\frac{\partial}{\partial t}(n_\epsilon) + \nabla \cdot \mathbf{\Gamma}_\epsilon + \mathbf{E} \cdot \mathbf{\Gamma}_e = S_{\text{en}} - (\mathbf{u} \cdot \nabla)n_\epsilon + (Q + Q_{\text{gen}})/q \quad (6-6)$$

$$\Gamma_\epsilon = -(\boldsymbol{\mu}_\epsilon \cdot \mathbf{E})n_\epsilon - \nabla(\mathbf{D}_\epsilon n_\epsilon)$$

where (the subscript ϵ refers to electron energy):

- n_ϵ is the electron energy density (SI unit: V/m^3)
- S_{en} is the energy loss/gain due to inelastic collisions (SI unit: $\text{V}/(\text{m}^3 \cdot \text{s})$)
- Q is an external heat source (SI unit: W/m^3)
- Q_{gen} is a generalized heat source (SI unit: W/m^3)
- μ_ϵ is the electron energy mobility which is either a scalar or a tensor (SI unit: $\text{m}^2/(\text{V} \cdot \text{s})$)
- \mathbf{E} is the electric field (SI unit: V/m), and
- D_ϵ is the electron energy diffusivity (SI unit: m^2/s).

The mean electron energy, $\bar{\epsilon}$ (SI unit: V) is computed through the expression

$$\bar{\epsilon} = \frac{n_\epsilon}{n_e} \quad (6-7)$$

Because of the high degree of nonlinearity inherent in the drift diffusion equation, the electron number density can span 10 orders of magnitude over a very small distance. In this region (the *plasma sheath*), the difference in the mobility and diffusivity between the ions and electrons creates a separation of space charge. This in turn produces a large electric field which can lead to a substantial increase in the mean electron energy. The best way of handling with this from a numerical point of view is to solve for the log of the electron number and energy density. This also prevents a divide by zero from occurring when equation Equation 6-7 is evaluated. So, letting $N_e = \ln n_e$, equation Equation 6-5 becomes

$$e^{N_e} \frac{\partial}{\partial t}(N_e) + \nabla \cdot [-e^{N_e}(\boldsymbol{\mu}_e \cdot \mathbf{E}) - e^{N_e}(\nabla(\mathbf{D}_e N_e))] = R_e - (\mathbf{u} \cdot \nabla)e^{N_e}$$

or

$$n_e \frac{\partial}{\partial t}(N_e) + \nabla \cdot [-n_e(\boldsymbol{\mu}_e \cdot \mathbf{E}) - n_e(\mathbf{D}_e \cdot \nabla N_e) - n_e(\nabla \cdot \mathbf{D}_e)] = R_e - n_e(\mathbf{u} \cdot \nabla)N_e$$

similarly, for the electron energy density:

$$n_\epsilon \frac{\partial}{\partial t}(E_n) + \nabla \cdot [-n_\epsilon(\boldsymbol{\mu}_\epsilon \cdot \mathbf{E}) - n_\epsilon(\mathbf{D}_\epsilon \cdot \nabla E_n) - n_\epsilon(\nabla \cdot \mathbf{D}_\epsilon)] + \mathbf{E} \cdot \boldsymbol{\Gamma}_e = S_{\text{en}} - n_\epsilon(\mathbf{u} \cdot \nabla)E_n$$

The resulting equation system is inherently more stable than the original equation system. A useful quantity for results and analysis is the electron “temperature” (SI unit: V) which is defined as:

$$T_e = \left(\frac{2}{3}\right)\bar{\epsilon}$$

Source Coefficients

The source coefficients in the equations in the [Electron Transport Theory](#) section are determined by the plasma chemistry and are written using either rate or Townsend coefficients. Suppose that there are M reactions which contribute to the growth or decay of electron density and P inelastic electron-neutral collisions. In general $P \gg M$. In the case of *rate coefficients*, the electron source term is given by

$$R_e = \sum_{j=1}^M x_j k_j N_n n_e$$

where x_j is the mole fraction of the target species for reaction j , k_j is the rate coefficient for reaction j (SI unit: m^3/s) and N_n is the total neutral number density (SI unit: $1/\text{m}^3$). When *Townsend coefficients* are used, the source term becomes

$$R_e = \sum_{j=1}^M x_j \alpha_j N_n |\Gamma_e|$$

where α_j is the Townsend coefficient for reaction j (SI unit: m^2) and Γ_e is the electron flux as defined above (SI unit: $1/(\text{m}^2 \cdot \text{s})$). Townsend coefficients can increase the stability of the numerical scheme when the electron flux is field driven as is the case with DC discharges.

The electron energy loss is obtained by summing the collisional energy loss over all reactions:

$$S_{\text{en}} = \sum_{j=1}^P x_j k_j N_n n_e \Delta \epsilon_j$$

where $\Delta \epsilon_j$ is the energy loss from reaction j (SI unit: V). In the case of Townsend coefficients, the energy loss is given by

$$S_{\text{en}} = \sum_{j=1}^P x_j \alpha_j N_n |\Gamma_e| \Delta \epsilon_j$$

The energy loss due to *inelastic collisions* is the most important input parameter in the Drift Diffusion interface. If this is not correctly defined, severe numerical problems result.

The rate and Townsend coefficients k_j and α_j depend exponentially on the mean electron energy, $\bar{\epsilon}$. When a Maxwellian EEDF is assumed, the rate and Townsend coefficients can be fitted with a function of the form:

$$k_j = A \bar{\epsilon}^\beta e^{-E/\bar{\epsilon}}$$

The physics interface does not provide a way of calculating the rate or Townsend coefficients. The multiphysics interfaces for the most common types of reactor automatically compute the rate coefficients for each electron impact reaction based on plasma chemistry.

Stabilization

When the source stabilization is on, additional source terms are added to [Equation 6-5](#) and [Equation 6-6](#). For the electron density, the following source is added:

$$R_e = N_A \exp(-\zeta \ln(n_e))$$

where ζ is the user-defined tuning parameter. When the electron density is very low, this acts as a source term which prevents the electron density from approaching zero. As the electron density increases this term becomes exponentially smaller, eventually becoming negligible for high electron densities. A similar source is added for the electron energy density:

$$S_{\text{en}} = N_A \exp(-\zeta \ln(n_e))$$

Tensor Transport Properties

If a strong DC magnetic field is present then the electron mobility can be a tensor:

$$\mu_e^{-1} = \begin{bmatrix} \frac{1}{\mu_{dc}} & -B_z & B_y \\ B_z & \frac{1}{\mu_{dc}} & -B_x \\ -B_y & B_x & \frac{1}{\mu_{dc}} \end{bmatrix}$$

where the inverse of the mobility has been used because the actual expression for the electron mobility cannot be written in a compact form. The quantity μ_{dc} is the electron mobility in the absence of a magnetic field. The electron diffusivity, energy mobility, and energy diffusivity are then calculated using:

$$\mathbf{D}_e = \mu_e T_e, \mu_\epsilon = \left(\frac{5}{3}\right) \mu_e, \mathbf{D}_\epsilon = \mu_\epsilon T_e$$

These equations are exact for a Maxwellian EEEDF, a constant momentum transfer frequency and a constant kinetic pressure.

The Wall Boundary Condition

At the [Wall](#), the exchange of electrons occurs through the following mechanisms:

- Loss of electrons due to a net flux of electrons from the *plasma bulk* to the wall.
- Loss of electrons due to the random motion of electrons within the mean free path of the wall.
- Gain of electrons due to secondary emission; a wall emits an electron with some probability when it is struck by a positive ion.
- Gain of electrons due to *thermionic emission*.

The resulting equation for the normal component of the electron flux at the wall is given by:

$$\mathbf{n} \cdot \Gamma_e = \frac{1-r_e}{1+r_e} \left(\frac{1}{2} v_{e,th} n_e \right) - \left[\sum_i \gamma_i (\Gamma_i \cdot \mathbf{n}) + \Gamma_t \cdot \mathbf{n} \right] \quad (6-8)$$

and for the normal component of the electron energy density:

$$\mathbf{n} \cdot \Gamma_\epsilon = \frac{1-r_e}{1+r_e} \left(\frac{5}{6} v_{e,th} n_\epsilon \right) - \left[\sum_i \gamma_i \bar{\epsilon}_i (\Gamma_i \cdot \mathbf{n}) + \bar{\epsilon}_t \Gamma_t \cdot \mathbf{n} \right] \quad (6-9)$$

here:

- r is the reflection coefficient (usually 0)
- α is 1 when the electron flux is directed toward the wall and zero otherwise
- $v_{e,\text{th}}$ is the thermal velocity (SI unit: m/s)
- n_γ is the number density of back scattered electrons (SI unit: $1/\text{m}^3$)
- γ_p is the secondary emission coefficient from the p th positive ion species
- Γ_p is the ion flux of the p th positive ion species at the wall (SI unit: $1/(\text{m}^2 \cdot \text{s})$)
- Γ_t is the thermal emission flux (SI unit: $1/(\text{m}^2 \cdot \text{s})$)
- ε_p is the mean energy of the p th species of emitted electrons (SI unit: V)
- ε_t is the mean energy of thermally emitted electrons (SI unit: V), and
- \mathbf{n} is the outward normal.

The thermal velocity is defined as:

$$v_{e,\text{th}} = \sqrt{\frac{8k_b T_e}{\pi m_e}}.$$

When the **Include migration effects** is activated the first term on the right-hand side of [Equation 6-8](#) becomes:

$$\frac{1-r_e}{1+r_e} \left(\frac{1}{2} v_{e,\text{th}} n_e + n_e (\boldsymbol{\mu}_e \cdot \mathbf{E}) \cdot \mathbf{n} \right)$$

and the first term on the right-hand side of [Equation 6-9](#) becomes:

$$\frac{1-r_e}{1+r_e} \left(\frac{5}{6} v_{e,\text{th}} n_e + n_e (\boldsymbol{\mu}_e \cdot \mathbf{E}) \cdot \mathbf{n} \right)$$

where \mathbf{E} is the electric field on the wall.

The Insulation Boundary Condition

The [Insulation](#) boundary condition simply sets the normal component of the electron and electron energy flux to zero:

$$\mathbf{n} \cdot \Gamma_e = 0 \text{ and } \mathbf{n} \cdot \Gamma_\varepsilon = 0$$

The Electron Density and Energy Boundary Condition

The [Electron Density and Energy](#) boundary condition fixes the electron density to a specific number density or the mean electron energy to a specific energy. This boundary condition should, in general, be avoided in favor of the [Wall](#) boundary condition.


$$n_e = n_{ew} \text{ and } \bar{\epsilon} = \bar{\epsilon}_0$$

Due to the fact that the log of the electron density is solved for, specifying an electron density of zero is not allowed.

Reference for the Drift Diffusion Interface

1. G.J.M. Hagelaar and L. C. Pitchford, “Solving the Boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models,” *Plasma Sources Science and Technology*, vol. 14, pp. 722–733, (2005).
2. E. Gogolides and H.H. Sawin, “Continuum modeling of radio-frequency glow discharges. I. Theory and results for electropositive and electronegative gases,” *J. Appl. Phys.*, vol. 72, 3971, 1992.



The Heavy Species Transport Interface

This chapter describes the Heavy Species Transport interface, which is a mass balance interface for all non-electron species. This physics interface also makes it possible to add electron impact reactions, chemical reactions, surface reactions, and species, including charged, neutral, and electronically excited species. The physics interface is found under the **Plasma** branch (.

In this chapter:

- [The Heavy Species Transport Interface](#)
- [Theory for the Heavy Species Transport Interface](#)

The Heavy Species Transport Interface

The **Heavy Species Transport (hs)** interface () , found under the **Plasma** branch () when adding a physics interface, adds electron impact reactions, gas phase reactions, and species and surface reactions to plasma models. The most convenient way to do this is to load a set of collision cross sections from a file in the **Cross Section Import** section from the **Settings** window for **Heavy Species Transport**.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (`_`) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `hs`.

TRANSPORT SETTINGS

Select a **Diffusion model**—**Mixture-averaged** (the default) or **Fick's law**. When using the **Mixture-averaged** model, the binary diffusion coefficients are automatically computed based on the data specified for each species.

Select an **Equation formulation**—**Log** (the default) to solve the equations in logarithmic form or **Linear** to solve equations in the original form. **Log** solves for the log of the dependent variables, ensuring that the mass fraction of any of the species is never lower than zero. This makes it more numerically stable but increases the nonlinearity of the equation system, and as such the model might take slightly longer to solve.

The **Migration in electric field** check box is selected by default. The **Migration** term is part of the total mass flux vector. Select the check boxes for other transport mechanisms to **Include**—**Convection**, **Calculate thermodynamic properties**, **Full expression for diffusivity**, or **Compute tensor ion transport properties**. The selection changes the

number of **Model Inputs** requiring values on the **Settings** window for **Diffusion, Migration**. Note the following:

- **Convection** of heavy species present in a plasma can often be neglected due to the low operating pressure.
- For **Calculate thermodynamic properties** select the thermodynamic properties of each reaction and species are computed automatically based on the thermodynamic properties of each species.
- For **Full expression for diffusivity** it computes a more accurate expression for the binary diffusion coefficients. Often the additional correction terms (the collision term, Ω_D given by Equation 7-17 and used in Equation 7-16) are negligible in which case the expressions are much simpler and the time taken to assemble the Jacobian matrix is reduced.
- For **Compute tensor ion transport properties** the tensor form of the ion transport properties when a static magnetic field is present are computed. This option only needs to be activated when a strong DC magnetic field exists and the operating pressure is very low (on the order of millitorr). When this option is activated an expression must be provided for the magnetic flux density which would typically be computed by another physics interface. This is set in the **Convection, Migration, Diffusion** feature.

ELECTRON ENERGY DISTRIBUTION FUNCTION SETTINGS

If cross-section data is used to define source coefficients in the model then an electron energy distribution function (EEDF) needs to be selected. These options are available.



For more information on the electron energy distribution function (EEDF), see the [Theory for the Boltzmann Equation, Two-Term Approximation Interface](#).

- **Maxwellian**. This option assumes a Maxwellian EEDF which takes the form:

$$f(\epsilon) = \phi^{-3/2} \beta_1 \exp(-(\epsilon \beta_2 / \phi))$$

where

$$\beta_1 = \Gamma(5/2)^{3/2} \Gamma(3/2)^{-5/2}, \beta_2 = \Gamma(5/2) \Gamma(3/2)^{-1}$$

where ϕ is the mean electron energy (eV), ϵ is the electron energy (eV) and Γ is the incomplete gamma function.

- **Druyvesteyn.** This option assumes a Druyvesteyn EEDF which takes the form:

$$f(\epsilon) = 2\varphi^{-3/2}\beta_1 \exp(-(\epsilon\beta_2/\varphi)^2)$$

where

$$\beta_1 = \Gamma(5/4)^{3/2}\Gamma(3/4)^{-5/2}, \beta_2 = \Gamma(5/4)\Gamma(3/4)^{-1}$$

- For **Generalized**, it is a generalized distribution function where the EEDF is somewhere between Maxwellian and Druyvesteyn. Specify a power law; the number must be between 1 and 2. Mathematically, the EEDF takes the form:

$$f(\epsilon) = g\varphi^{-3/2}\beta_1 \exp(-(\epsilon\beta_2/\varphi)^g)$$

where

$$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2}, \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}, 1 \leq g \leq 2$$

- **Function.** If a two-dimensional interpolation function has been added to the model, it can be used for the EEDF. In this case, the x -data should be the electron energy (eV) and the y -data should be the mean electron energy (eV).



The two-dimensional interpolation function can be computed using a parametric sweep in [The Boltzmann Equation, Two-Term Approximation Interface](#). This allows for modeling of discharges where the EEDF is far from Maxwellian. For step-by-step instructions on how to do this, refer to this blog entry:
<https://www.comsol.com/blogs/the-boltzmann-equation-two-term-approximation-interface/>

In all these cases the rate constants in the model are automatically computed based on the selected EEDF using the formula:

$$k_f = \gamma \int_0^{\infty} \epsilon \sigma_k(\epsilon) f(\epsilon) d\epsilon \quad (7-1)$$

The rate coefficients when computed using cross section data are a highly nonlinear function of the mean electron energy. COMSOL Multiphysics automatically computes the integral in [Equation 7-1](#) and makes the result available for evaluation of the rate coefficient. The variation of the rate coefficient for any particular model can be plotted using `<name>.kf_<reaction_number>`. For example, for reaction number 3 in a

Heavy Species interface, with name `hs`, the rate coefficient would be plotted using `hs.kf_3`.


CHEMKIN IMPORT FOR SPECIES PROPERTY



Import thermodynamic and transport properties for the reacting species into the model.

STABILIZATION

If the **Equation formulation** is set to **Log** then the solver can run into difficulties when the species mass fractions approach zero. The **Reaction source stabilization** check box (selected by default) adds an additional source term to the rate expression for each species. In the `t` text field, enter a tuning parameter for the source stabilization. The default value is 1. This value is usually good enough. If the plasma is high pressure (atmospheric) then it can help to lower this number to somewhere in the range of 0.25–0.5.

DISCRETIZATION

To display this section, click the **Show** button () and select **Discretization**.

	<ul style="list-style-type: none">• Stabilization• Domain, Boundary, and Pair Nodes for the Heavy Species Transport Interface• Theory for the Heavy Species Transport Interface• Interpolation in the <i>COMSOL Multiphysics Reference Manual</i>
	<p><i>Surface Chemistry Tutorial</i>: Application Library path Plasma_Module/Chemical_Vapor_Deposition/surface_chemistry_tutorial</p>

Domain, Boundary, and Pair Nodes for the Heavy Species Transport Interface

The [Heavy Species Transport Interface](#) has these domain, boundary, and pair nodes available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).



In general, to add a node, go to the **Physics** toolbar, no matter what operating system you are using. Subnodes are available by clicking the parent node and selecting it from the **Attributes** menu.



The boundary conditions are available for both charged and neutral species. Surface reactions can be added to the model—COMSOL Multiphysics then automatically generates the appropriate expressions for the fluxes for each species on the boundaries. Boundary conditions do not need to be applied for a species that comes from the mass constraint condition.

- [Cross Section Import](#)
- [Diffusion and Migration](#)
- [Electron Impact Reaction](#)
- [Flux](#)
- [Inlet](#)
- [Outflow](#)
- [Reaction](#)
- [Species](#)
- [Surface Reaction](#)
- [Surface Species](#)
- [Symmetry](#)



Due to the overwhelming number of species that can exist in a plasma, the **Flux**, **Inlet**, **Outflow**, and **Symmetry** boundary condition subnodes are available from the context menu (right-click the **Species** parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model.



In the *COMSOL Multiphysics Reference Manual* see [Table 2-3](#) for links to common sections and [Table 2-4](#) to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

Cross Section Import

The **Cross Section Import** node allows you to import collision cross section data into the model. The cross section data must be in a specific format. Browse to a file containing a set of collision cross sections.



The file must follow the format documented in [Importing Collision Cross-Section Data](#).

Diffusion and Migration

The **Diffusion and Migration** node sets up the model inputs for the physics interface. Choose how to define the density of the gas mixture.



The node name changes to **Diffusion** or **Convection, Diffusion, and Migration** based on the check boxes selected in the **Transport Settings** section.

MODEL INPUTS

This section has fields and values that are inputs to expressions that define material properties. If such user-defined property groups are added, the model inputs display here. Initially, this section is empty. The input depends on which **Transport Settings** are selected on [The Heavy Species Transport Interface](#).

- Enter a **Temperature** T (SI unit: K) for the gas mixture used to compute the gas density and species transport properties.
- Enter an **Absolute pressure** p_A (SI unit: Pa) to compute the gas density and the binary diffusion coefficients.
- When the default **Migration in electric field** check box is selected, enter an **Electric potential** V (SI unit: V). This is used to compute the electric field that results in migration of any charged species.

- If the **Convection** check box is selected, enter coordinates for the **Velocity field μ** (SI unit: m/s).
- If the **Compute tensor electron transport properties** check box is selected, enter coordinates for the **Magnetic flux density \mathbf{B}** (SI unit: T). This usually comes from a magnetostatic model, and is used to compute anisotropic transport properties for electrons.

DENSITY

Select an option from the **Mixture density** list—Ideal gas (the default) or **User defined**. For **User defined** enter a value for ρ (SI unit: kg/m³).


For **Ideal gas** it computes the expression for the gas density automatically using the ideal gas law, based on the composition of the mixture. The density is a function of the mean molecular weight

$$\rho = \frac{p_A M_n}{RT}$$

where p_A is the absolute pressure, T is the temperature, R is the universal gas constant, and M_n is the mean molar mass given by

$$\frac{1}{M_n} = \sum_{k=1}^Q \frac{w_k}{M_k}$$

Electron Impact Reaction

The **Electron Impact Reaction** node () sets up the reaction rate for a reaction involving an electron and a target species.

REACTION FORMULA

Enter chemical reaction formulas in the **Formula** text field. The physics interface examines the species taking part in the reaction and automatically adds the associated **Species** features to the **Model Builder**.



Electron impact reactions are always assumed to be irreversible. However, the reverse reaction is automatically created if it is specified that a detailed balance should be constructed in the cross section data file. See [Importing Collision Cross-Section Data](#).

Entering a Formula



The letter “e” is reserved to represent an electron. Therefore if a formula such as $e + \text{Ar} \Rightarrow e + \text{Ar}^+$ is entered, the species **e** is automatically assigned as an electron.

Valid species names consist of a case-sensitive combination of letters, numbers, and “+” or “-” signs. With the exception of the “+” and “-” signs, which are used for ions, special characters are not allowed as species names or part of species names (for example, (), [], *, #, and _). In addition, the first character in a species name must be a letter to avoid confusion with the stoichiometric coefficients in the reaction formula. Ions are labeled with one plus or minus sign per species charge added directly after the species name. For example, O^{++} represents an oxygen atom with a twofold positive charge; O_2^+ represents molecular oxygen with a single positive charge; and 2O^+ represents two oxygen atoms, each with a single positive charge. This automatically fills in the charge of the species ion the **Charge** text field in the associated **Species** node.

COLLISION TYPE



This section is available when a valid formula for the electron impact reaction is entered.

Select a **Collision type**— **Elastic** (the default), **Excitation**, **Attachment**, or **Ionization**.

- For **Elastic** enter a value for the **Electron mass ratio** m_r (dimensionless) (the ratio of the mass of the electron to the target species).
- Select **Excitation** for excitation reactions where the collision is inelastic. Enter the **Energy loss** Δe (SI unit: V) for the reaction. This is also known as threshold energy for the reaction. Select this option for electronic excitation, vibrational and rotational excitation, and superelastic collisions. For superelastic collisions, the activation energy is negative.
- Select **Attachment** for collisions which result in the electron attaching to the target species, forming a negative ion. No user input is required.
- Select **Ionization** for electron impact reactions which result in secondary electrons being formed. Enter the **Energy loss** Δe (SI unit: V) for the reaction. This is also known as threshold energy for the reaction.

COLLISION



This section is available when a valid formula for the electron impact reaction is entered.

Specify how to specify the reaction rate constant. There are these options:

- For **Cross section data** enter cross section data directly into a table for the electron impact reaction.
- For **Arrhenius parameters** use Arrhenius coefficients to specify the reactions dependence on the electron “temperature,” T_e .
- For **Rate constant** specify an arbitrary expression for the rate constant. This can be a numeric value or a complicated function of the electron “temperature”, gas temperature or any other expression. This is the default option for specifying the reaction.
- For **Use lookup table** either enter or load in a lookup table of the source coefficient versus mean electron energy.

CROSS SECTION DATA



This section is available when a valid formula is entered for the electron impact reaction and the **Specify reaction using** option is set to **Cross section data** in the **Collision** section.

By default the electron energy distribution function is taken from the physics interface property. It can be changed to **Maxwellian**, **Druyvesteyn**, or **Generalized**. In this case, the selected distribution function is only used in the current electron impact reaction. The **Electron energy** in the **Collision cross section data** table should have units of electronvolts (eV). The units of the **Collision cross section data** has units of m^2 .



If a set of collision cross sections are loaded from a file with the [Cross Section Import](#) node, the software automatically generates the electron impact collisions and enters the cross section data into the **Cross section data** table.



ARRHENIUS PARAMETERS

For **Arrhenius parameters** specify the reaction then COMSOL Multiphysics automatically generates an expression of the form

$$k_f = A_f T_e^{n_f} \exp(-E_f/T_e)$$

for the rate constant. Here, A_f is the forward frequency factor, n_f is the electron temperature exponent and E_f is the activation energy (eV). The quantity T_e is the electron “temperature” (eV) which is defined as

$$T_e = \frac{2}{3} \bar{\epsilon}$$

where $\bar{\epsilon}$ is the mean electron energy.

KINETIC EXPRESSIONS

If the **Specify reaction using** option is set to **Rate constant** then enter in an expression for the reaction forward rate constant in the **Forward rate constant** text field. The units of the rate constant depend on the order of the reaction.


TABLE 7-1: SI UNITS FOR THE RATE CONSTANT FOR DIFFERENT REACTION ORDERS

ORDER	EXAMPLE	SI UNIT
1	<chem>O2s=>O2</chem>	1/s
2	<chem>O+O2=>O3</chem>	m ³ /(s·mol)
3	<chem>3O=>O2+O</chem>	m ⁶ /(s·mol ²)

SOURCE COEFFICIENT DATA

If the **Specify reaction using** option is set to **Use lookup table** then the **Source coefficient** data section appears. Chose between **Rate coefficients** and **Townsend coefficients** to specify the source coefficients. For **Rate coefficients** either enter in the lookup table or load in from file a table of mean electron energy (eV) versus rate coefficient (SI unit: m³/s). For **Townsend coefficients** enter in or load data from file a table of mean electron energy (eV) versus Townsend coefficient (SI unit: m²).

Reaction

Use the **Reaction** node () to enter reversible or irreversible reactions which don't involve electrons, for example for gas phase reactions. Enter the **Formula** for the reaction into the field and then fill in the appropriate information about the reaction.

DOMAIN SELECTION

COMSOL Multiphysics automatically ensures that the **Reaction** selection is the same as the [Diffusion and Migration](#) selection.

REACTION FORMULA

See the Reaction Formula settings for the [Electron Impact Reaction](#) node.

ARRHENIUS PARAMETERS



This section is available when a valid formula for the reaction is entered.

Choose whether or not to use Arrhenius expressions to compute the reaction forward and reverse rate constants using:

$$k = AT^n \exp\left(-\frac{E}{R_g T}\right)$$

To activate the **Arrhenius Parameters** section and make the kinetic expressions use these parameters, select the **Use Arrhenius expressions** check box. In the **Arrhenius Parameters** section, A denotes the frequency factor, n the temperature exponent, E the activation energy (SI unit: J/mol), and R_g the gas constant, 8.314 J/(mol·K). The pre-exponential factor, including the frequency factor A and the temperature factor T^n , is given the units (m³/mol) ^{α -1}/s, where α is the order of the reaction (with respect to the concentrations). text fields are available for each of the Arrhenius parameters in the equation above.


KINETIC EXPRESSIONS

Modify the default expressions for the rate constant. This field is only available if the **Use Arrhenius** check box is cleared. The units of the rate constant depend on the order of the reaction.

TABLE 7-2: SI UNITS FOR THE RATE CONSTANT FOR DIFFERENT REACTION ORDERS

ORDER	EXAMPLE	SI UNIT
1	$O_2 \rightarrow O_2$	1/s
2	$O + O_2 \rightarrow O_3$	$m^3/(s \cdot mol)$
3	$3O \rightarrow O_2 + O$	$m^6/(s \cdot mol^2)$

Species

Use the **Species** node () to add additional species to the model even if they do not participate in any reactions.



Due to the overwhelming number of species that can exist in a plasma, the **Flux**, **Inlet**, **Outflow**, and **Symmetry** boundary condition subnodes are available from the context menu (right-click the **Species** parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model.



Species are automatically added to the model when reactions are added.

DOMAIN SELECTION

COMSOL Multiphysics automatically ensures that the **Species** selection is the same as the [Diffusion and Migration](#) selection.

SPECIES FORMULA

Species Name

Enter basic information about the name and nature of the species. When a new species is created, the **Species name** field is editable. A valid name for the species must be entered, for example, o2, Ar, Ar+. After the species name has been entered the text field cannot be edited (as long as a valid species name is entered).

Species Type

The **Species type** is either **Neutral**, **Ion**, or **Electron**.

- Enter “e” to automatically create a species as an **Electron**. The **Species** node is renamed to include the name entered in the **Species type** field. Also enter the [Electron Parameters](#).
- Enter a neutral species, for example, o2 to automatically create a species as **Neutral**.
- Enter a charged species, for example, Ar+ or o2- to automatically create a species as an **Ion**. The **Species** node is renamed to include the name entered in the **Species type** field. When the species created is an **Ion**, also select the **Initial value from electroneutrality constraint** check box to indicate that the initial mass fraction of this species is automatically computed based on the initial electron density specified in the electron species properties. This ensures that the solver can find consistent initial conditions when starting because the plasma is initially electrically neutral. Also enter the [General Parameters](#) described in the next section.
- If **Neutral** is selected as the **Species type**, enter the [General Parameters](#) described in the next section.

For **Neutral** select the **From mass constraint** check box to indicate which species should be used as the mass constraint and therefore not solved for. This is typically the species with the highest mass fraction. It should not be a species whose mass fraction is a very small value (as is the case with ionic species).

GENERAL PARAMETERS

This section is available when the **Species type** under **Species Formula** selected is **Ion** or **Neutral**. Enter the physical properties of the species. Enter the following properties:

- **Preset species data.** The default value is **User defined** which means that the transport properties are computed from the values given in the text fields below. There is preset transport data available for the most common gases encountered in plasma processing. Selecting one of these gases fills in the correct values for the species transport properties automatically.
- **Molecular weight M_w** (SI unit: kg/mol) of the species. The default value for the molecular weight is 0.032. The molecular weight must be a nonzero number.
- **Charge z** (dimensionless) number of the species. The charge number is automatically computed from the formula specified via either a reaction or by manually creating a species.

- **Potential characteristic length** σ (SI unit: m). The potential characteristic length is used to compute the binary diffusion coefficients used in the Mixture-Averaged transport setting.
- **Potential energy minimum** e/k_b (SI unit: m). This is used to compute the binary diffusion coefficients.
- **Dipole moment** μ_D (SI unit: C·m). If the dipole moment is known, enter its value and obtain a more accurate expression for the binary diffusion coefficients.
- **Thermal diffusion coefficient** D^T (SI unit: m^2/s). If thermophoretic effects are important in the model then enter a value for the thermal diffusion coefficient.
- If **Neutral** is selected as the **Species type**, enter an **Initial mole fraction** x_0 (dimensionless). Set the initial mole fraction of the selected species in the gas mixture. If the **From mass constraint** check box is selected, this field is not available.
- If **Ion** is selected as the **Species type**, enter an **Initial number density** n_0 (dimensionless) to set the initial number density of the selected species in the gas mixture. If the **Initial condition from electroneutrality constraint** check box is selected, this field is not available.
- **Diffusion coefficient** D_f (SI unit: m^2/s). If the **Diffusion model** property is set to **Fick's law** then the diffusion coefficient is not computed automatically from the transport properties, its value is taken from this text field.

ELECTRON PARAMETERS

This section is available when the **Species type** under **Species Formula** selected is **Electron**. Enter the physical properties of the species. Enter the following properties:

- **Electron density** n_e (SI unit: $1/m^3$).
- **Mean electron energy** e (SI unit: V).
- **Initial electron density** $n_{e,0}$ (SI unit: $1/m^3$).

SPECIES THERMODYNAMIC PARAMETERS

This section is available if the **Calculate thermodynamic properties** is check box is selected under [Transport Settings](#) on the Heavy Species Transport interface.

Enter thermodynamic properties for the species using the NASA polynomial format. Any coefficients for the thermodynamic polynomials are entered into the $\alpha_{low,k}$ fields and apply to the temperatures in the range **T_{int}/lower** to **T_{int}/midpoint**; coefficients

entered into the $a_{hi,k}$ fields apply to temperatures in the range $T_{int}/midpoint$ to $T_{int}/upper$ range.

- Enter the **Lower temperature limit** T_{lo} , **Middle temperature limit** T_{mid} , and **Upper temperature limit** T_{hi} . The defaults are 300 K, 1000 K, and 5000 K respectively.
- Enter the **Polynomial coefficients** $a_{low,k}$ and $a_{hi,k}$.
- Enter an **Additional enthalpy contribution** Δh (SI unit: V) as required.

MOBILITY AND DIFFUSIVITY EXPRESSIONS

This section is available when the **Species type** under **Species Formula** selected is **Ion**. The default specification is to compute the diffusivity and mobility from the binary diffusion coefficients and Einstein's relation.

Select a **Specification—Compute mobility and diffusivity** or **Specify mobility, compute diffusivity**.

The **Mobility** is computed using Einstein's relation by default. If **Mixture averaged** is selected as the **Diffusion model** under **Transport Settings** on the Heavy Species Transport interface, then the mobility is computed using:

$$\mu = \frac{D_m q}{k_B T}$$

and if the Fick's law property is selected, then the mobility is defined as:


$$\mu = \frac{D_f q}{k_B T}$$

Select an **Ion temperature—Use gas temperature** or **Specify ion temperature**. For **Specify ion temperature** enter an **Ion temperature** T_{ion} (SI unit: K). The default is 300 K.

MOBILITY SPECIFICATION

This section is available if **Specify mobility, compute diffusivity** is selected under **Mobility and Diffusivity Expressions**.


Select an option from the **Specify using** list—**Constant value** or **Lookup table**.

- For **Constant value** enter a value or expression for the **Mobility** u_m (SI unit: $m^2/(V \cdot s)$). This can be a function of any of the variables in the model.
- For **Lookup table** the mobility is then a function of the electric field. Select an option from the **Mobility a function of** list—**Electric field** or **Reduced electric field**. Enter values into the table or click the **Load from file** button () to import data from a text file.

The file must contain the two columns of data, the **Electric field** (SI unit: V/m) or the **Reduced electric field** (SI unit: $\text{V}\cdot\text{m}^2$) as the x -data and the **ion mobility** (SI unit: $\text{m}^2/(\text{V}\cdot\text{s})$) or **reduced ion mobility** (SI unit: $1/(\text{V}\cdot\text{m}\cdot\text{s})$) as the y -data.

Select the **Use local field approximation for ion temperature** check box to automatically compute the ion temperature using the local field approximation. The local field approximation can be useful for high pressure discharges where the electric field is very high, as is the case for dielectric barrier discharges.

Surface Reaction

The **Surface Reaction** node () sets up reactions involving gas phase, surface, and bulk species on surfaces. Surface reactions play an important role in the overall characteristics of a discharge. This typically involves first order reactions on surfaces where excited or ionic species transition back to their ground state.

REACTION FORMULA

The first thing to input after creating a new reaction is the **Reaction Formula**. The reaction can only be irreversible. Valid species names consist of a case-sensitive combination of letters, numbers, and “+” or “-” signs. With the exception of the “+” and “-” signs, which are used for ions, special characters are not allowed as species names or part of species names (for example, [], *, #, and _). In addition, the first character in a species name must be a letter to avoid confusion with the stoichiometric coefficients in the reaction formula. Ions are labeled with one plus or minus sign per species charge added directly after the species name.



For surface reactions, use the notation (s) to denote surface species or (b) to denote bulk species. This is bulk surface species, not to be confused with gas phase species. The species is then called `name_surf` or `name_bulk` depending on the species type. For example, using `Si(s)` in the reaction formula creates a surface species with the name `Si_surf` and `Si(b)` creates a surface species with the name `Si_bulk`.

Under **Specify reaction using** select whether to specify the surface reaction using **Sticking coefficient** or **Rate coefficient**. Sticking coefficients are typically easy to estimate intuitively and must be between 0 and 1. Rate coefficients require specification of a rate constant, which can be obtained in the appropriate literature.

Under **Motz-Wise correction** select whether to include this correction in the expression for the rate constant. This option is only available when the reaction is specified using

sticking coefficients. When the **Motz-Wise correction** is set to **On**, Equation 7-19 is used to compute the rate constant. When set to **Off**, Equation 7-20 is used.

Only irreversible surface reactions are supported.

ARRHENIUS PARAMETERS



This section is available when a valid formula for the surface reaction is entered.

Choose whether or not to use Arrhenius expressions to compute the sticking coefficients. When this option is activated the expression for the sticking coefficient is automatically computed using:


$$\gamma = aT^b \exp\left(-\frac{e}{R_g T}\right)$$

where a is the frequency factor, b is the temperature exponent, and e is the activation energy (SI unit: J/mol).

KINETIC EXPRESSIONS

Enter the **Forward sticking coefficient** for the surface reaction. The option is available if the **Use Arrhenius expressions** check box is not selected.

Surface Species

The **Surface Species** () node adds detailed surface chemistry to the model. The species can either be surface (for example, Si(s)) or bulk species (for example, Si(b)). This is usually generated by default when a formula for a surface reaction is entered.

SPECIES FORMULA

Enter a formula for the surface species. The symbols (s) and (b) are reserved to denote surface species or bulk species, respectively. So, to create a silicon surface species simply type Si(s). To create a silicon bulk species type Si(b). COMSOL Multiphysics automatically renames the species to a valid name, for example Si_surf or Si_bulk.

GENERAL PARAMETERS

Enter the molecular weight (SI unit: kg/mol) of the surface or bulk species. The default value is **User defined** which means that the surface species properties are computed from the values given in the text fields below. There is preset transport data

available some of the typical surface species encountered in plasma processing. Selecting one of these fills in the correct values for the molecular weight and thermodynamic properties automatically.

SURFACE SPECIES PARAMETERS

If the species is a surface species then this section is available.

- Enter the **Site occupancy number** σ (dimensionless) of the species. The default value is 1. This accounts for the fact that larger molecules can occupy more than one active site on a surface.
- Enter the **Initial density of surface sites** $\Gamma_{n,0}$ (SI unit: mol/m²) of the species. The default value is 1.95E-5. This surface site concentration remains constant, with this initial value if every surface species has the same site occupancy number.
- The **Initial site fraction** $Z_{k,0}$ (dimensionless) is the initial fraction of the total sites which is occupied by the species. This must be a number between 0 and 1.

BULK SPECIES PARAMETERS

If the species is a bulk species then enter the density (SI unit: kg/m³). The default value is 2329 kg/m³ corresponding to the density of silicon.

Flux

Due to the overwhelming number of species which can exist in a plasma, the **Flux** boundary condition subnode is available from the context menu (right-click the **Species** parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model.

The **Flux** subnode prescribes the mass flux of the selected species on selected boundaries. A user-defined normal flux is available via the equation:

$$-\mathbf{n} \cdot \rho w_k (\mathbf{V}_k + \mathbf{u}) = \Gamma_k$$

Inlet

Due to the overwhelming number of species which can exist in a plasma, the **Inlet** boundary condition subnode is available from the context menu (right-click the **Species** parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model.

The **Inlet** subnode adds a boundary condition for an inflow boundary, where a condition for all species is specified. The condition can be specified using these quantities:

- The mass fraction: $\omega = \omega_0$.
- The mole fraction: $x = x_0$.
- The molar concentration (SI unit: mol/m^3) $c = c_0$.
- The number density, which describes the number of particles per volume (SI unit: $1/\text{m}^3$): $n = n_0$.

A concentration quantity other than the mass fractions can only be used when all species are defined, as in the Inlet condition. The other quantities are composition dependent and therefore unambiguous only when all species are defined.

INLET

Select the type of input from the **Mixture specification** list:

- Select **Mass fraction** (the default) to enter mass fractions ($\omega_{0,\omega 1}$, for example).
- Select **Mole fraction** to enter mole fractions ($x_{0,\omega 1}$, for example).
- Select **Molar concentration** to enter molar concentrations ($c_{0,\omega 1}$, for example).
- Select **Number density** to enter number densities ($n_{0,\omega 1}$, for example).

Type the value or expression in the text fields for each species except the one computed from the mass constraint.



When using the Logarithmic form of the equations, setting a zero value for any of the options listed above results in an error when attempting to solve. To avoid the solver taking $\log(0)$ a positive, nonzero value must be entered.

Outflow

Due to the overwhelming number of species that can exist in a plasma, the **Outflow** boundary condition subnode is available from the context menu (right-click the **Species** parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model.

The **Outflow** subnode is used on boundaries where the species mass is to be transported out of the model domain.

If an imaginary slice is drawn through the cross section of a geometry, then there is no net flux across the slice due to diffusion, since diffusion corresponds to random motion of the atoms and molecules. There can, however, be a net flux due to convection, migration or thermophoresis, but outlet boundary conditions for temperature and electric field also impose zero gradient. An appropriate “open” boundary condition is to set the normal component of the diffusive flux to zero, which implies that diffusive transport of flux across the imaginary slice is zero. The boundary condition becomes:

$$-\mathbf{n} \cdot \mathbf{j}_k = 0$$

which assumes that all mass passing through the boundary is due to convection.

Symmetry

Due to the overwhelming number of species which can exist in a plasma, the **Symmetry** boundary condition subnode is available from the context menu (right-click the [Species](#) parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model.

The Symmetry subnode is used on all boundaries where the selected species mass fraction is symmetric. This is identical to the No Flux boundary condition. The symmetry boundary imposes a zero flux boundary condition.

Theory for the Heavy Species Transport Interface

The Heavy Species Transport Interface theory is described in this section:

- [About the Transport Equations](#)
- [Multicomponent Diffusion Equations](#)
- [Additional Quantities of Interest](#)
- [Auxiliary Relations](#)
- [Source Coefficients](#)
- [Thermodynamic Properties](#)
- [Species Transport Properties](#)
- [Mixture Transport Properties](#)
- [Scaling](#)
- [Stabilization](#)
- [Surface Reactions](#)
- [Flux Matching](#)
- [About Surface Species](#)
- [Limitations of the Heavy Species Transport Equations](#)
- [References for the Heavy Species Transport Interface](#)

About the Transport Equations

There are a number of different formulations available for modeling mass transport, ranging from a simple reaction, convection, diffusion equation to the Maxwell-Stefan (sometimes called Stefan-Maxwell) equations. The Maxwell-Stefan equations are the only formulation that conserves the total mass in the system and also satisfies a number of auxiliary constraints. The main drawback of the Maxwell-Stefan equations is the amount of computational resources needed to solve them. In many situations, simplified forms of the Maxwell Stefan equations can be applied, which do not rigorously satisfy the laws of multicomponent diffusion but are much easier to implement and consume less computational resources.

Neutral and excited species in a plasma require a suitable transport equation. Typically the concentration of excited species is very low and they can be considered dilute. This is true with atomic gases, where the concentration of ions and electronically excited species is much lower than the concentration of the ground state atoms. Molecular gases can dissociate into stable neutral fragments via *electron impact dissociation*. These fragments can have a long lifetime and accumulate inside a reactor. In this case, the dilute approximation is no longer valid and a more rigorous description of the mass

transport is necessary. Conservation of mass dictates that the sum of the mass fractions of all the species must equal one. The most accurate technique is to solve the Maxwell-Stefan equations which correctly take into account diffusive transport due to mole fraction, pressure and temperature gradients. The Maxwell-Stefan equations quickly become computationally very expensive when the number of species becomes large (>6). In a plasma, there might be 20+ neutral and excited species, so solving the full Maxwell-Stefan equations is not practical.

The Heavy Species Transport interface provides two different Diffusion models to choose from, Mixture averaged and Fick's law. The Mixture averaged option fulfills all the criteria for mass conservation. It is much less computationally expensive to solve than the full Maxwell-Stefan equations, which are not available in the Heavy Species Transport interface. The Fick's law option uses an even simpler diffusion model, which is less accurate than the Mixture averaged model, but is computationally less expensive.

Multicomponent Diffusion Equations

Suppose a reacting flow consists of $k = 1, \dots, Q$ species and $j = 1, \dots, N$ reactions. The equation for the first $Q - 1$ species is given by

$$\rho \frac{\partial}{\partial t}(w_k) + \rho(\mathbf{u} \cdot \nabla)w_k = \nabla \cdot \mathbf{j}_k + R_k \quad (7-2)$$

where:

- \mathbf{j}_k is the diffusive flux vector
- R_k is the rate expression for species k (SI unit: $\text{kg}/(\text{m}^3 \cdot \text{s})$)
- \mathbf{u} is the mass averaged fluid velocity vector (SI unit: m/s)
- ρ denotes the density of the mixture (SI unit: kg/m^3), and
- w_k is the mass fraction of the k th species (1).

The diffusive flux vector is defined as:

$$\mathbf{j}_k = \rho w_k \mathbf{V}_k$$

where \mathbf{V}_k is the multicomponent diffusion velocity for species k . The definition of \mathbf{V}_k depends on the option chosen for the Diffusion Model property. If Mixture Averaged is chosen then:

$$\mathbf{V}_k = D_{k,m} \frac{\nabla w_k}{w_k} + D_{k,m} \frac{\nabla M_n}{M_n} + \frac{D_k^T}{\rho w_k} \frac{\nabla T}{T} - z_k \mathbf{u}_{k,m} \mathbf{E}$$

where

- $D_{k,m}$ is the mixture averaged diffusion coefficient (SI unit: m^2/s)
- M_n is the mean molar mass of the mixture (SI unit: kg/mol)
- T is the gas temperature (SI unit: K)
- D_k^T is the thermal diffusion coefficient for species k (SI unit: $\text{kg}/(\text{m}\cdot\text{s})$)
- z_k is the charge number for species k (SI unit: dimensionless)
- $\mu_{k,m}$ is the mixture averaged mobility for species k (SI unit: $\text{m}^2/(\text{V}\cdot\text{s})$)
- \mathbf{E} is the electric field (SI unit: V/m)

The mixture averaged diffusion coefficient, $D_{k,m}$ for species k is defined as:

$$D_{k,m} = \frac{1 - w_k}{Q \sum_{j \neq k} x_j / D_{kj}} \quad (7-3)$$

The mean molar mass, M_n is computed using the sum of the mass fractions and molecular weights of all species:

$$\frac{1}{M_n} = \sum_{k=1}^Q \frac{w_k}{M_k}.$$

The mole fraction for species k , which is required in [Equation 7-3](#) can be computed from the mass fraction of species k and the mean molar mass:

$$x_k = \frac{w_k}{M_k} M_n.$$

The quantity, D_{kj} is the binary diffusion coefficient between species k and j and are rather complicated functions of the physical properties of the species. They are defined later, in [Equation 7-12](#).

By default, the mixture averaged mobility, $\mu_{k,m}$ is computed using *Einstein's relation*:

$$\mu_{k,m} = \frac{q D_{k,m}}{k_B T}$$

where q is the unit charge (SI unit: s A) and k_B is Boltzmann's constant (SI unit: J/K). For ions, it is possible to specify an arbitrary expression for the mobility in the Mobility

and Diffusivity Expressions and Mobility Specification sections. One option is to use something called the *Local Field Approximation*. In this case, the mobility is specified via a lookup table, the local field approximation can be used to compute the ion temperature and thus the diffusivity through Einstein's relation, see [Ref. 7](#). The ion temperature is computed using:

$$T_i = T + \left(\frac{m_k + m_n}{5m_k + 3m_n} \right) \frac{m_n}{k_b} (\boldsymbol{\mu} \cdot \mathbf{E}) \cdot (\boldsymbol{\mu} \cdot \mathbf{E})$$

where M_k is the ion mass (SI unit: kg), T is the background gas temperature (SI unit: K), M_n is the mean mass of the mixture (SI unit: kg), and \mathbf{E} is the electric field (SI unit: V/m)

If the Diffusion Model is changed to Fick's law then the diffusion coefficient must be given as an input for each species. In this case, the diffusion velocity is defined as:

$$\mathbf{V}_k = D_{k,f} \frac{\nabla w_k}{w_k} + D_{k,f} \frac{\nabla M_n}{M_n} + \frac{D_k}{\rho w_k} \frac{\nabla T}{T} - z_k \mu_{k,f} \mathbf{E}$$

where $D_{k,f}$ is the user-defined diffusion coefficient for species k . The mobility for species k is then given by:

$$\mu_{k,f} = \frac{q D_{k,f}}{k_B T}.$$

This approach is much less computationally expensive than the Mixture averaged model since the diffusion coefficients are not computed from the complicated expression in [Equation 7-3](#).

Additional Quantities of Interest

The Heavy Species Transport interface also defines a number of additional variables for quantities which are typically of interest during results processing. The number density of species k is given by

$$n_k = \left(\frac{p}{k_B T} \right) x_k$$

where p (SI unit: Pa) is the absolute pressure of the fluid. The molar concentration of species k is given by

$$c_k = \frac{w_k \rho}{M_k};$$

The density of the mixture is given by the ideal gas law by default:

$$\rho = \frac{p M_n}{R T}$$

where R is the universal gas constant (SI unit: J/(mol·K)).

A useful quantity in systems where large thermal gradients exist (that is, chemical vapor deposition processes) is the *thermal drift velocity* (SI unit: m/s), which is defined as

$$\mathbf{v}^T = \frac{D_k^T}{\rho w_k} \nabla \ln T$$

Auxiliary Relations

Only $Q - 1$ of the Maxwell-Stefan equations are independent and so a *closure law* is required. One way of ensuring a unique solution is the use the fact that the sum of the diffusive fluxes must be zero:

$$\sum_{k=1}^Q \rho w_k \mathbf{V}_k = 0 \quad (7-4)$$

An alternative is to use the fact that the sum of the mass fractions must be equal to one:

$$w_Q = 1 - \sum_{k=1}^{Q-1} w_k \quad (7-5)$$

In the Heavy Species Transport interface, [Equation 7-5](#) is used to fulfill the mass constraint so only $Q - 1$ species are solved for.

Source Coefficients

The rate expression, R_k is determined by the stoichiometry of the system through the formula

$$R_k = M_k \sum_{j=1}^N \nu_{kj} r_j \quad (7-6)$$

where ν_{kj} is the stoichiometric matrix and r_j is the reaction rate corresponding to reaction j . The reaction rates depend on the order of the reaction and the molar concentration of each species. In compact form, the equation for the j th reaction rate is

$$r_j = k_{f,j} \prod_{k=1}^Q c_k^{\nu'_{kj}} - k_{r,j} \prod_{k=1}^Q c_k^{\nu''_{kj}}$$

where:

- c_k is the molar concentration of species k (SI unit: mol/m³)
- $k_{f,j}$ is the rate coefficient for the j th forward reaction (SI unit: 1/s for first order reactions, m³/(mol·s) for second order reactions or m⁶/(mol²·s) for third order reactions)
- ν'_{kj} is the stoichiometric matrix corresponding to forward reactions, and
- ν''_{kj} is the stoichiometric matrix corresponding to reverse reactions.

The molar concentration of the k th species is computed via:

$$c_k = \frac{w_k \rho}{M_k}.$$

The rate coefficients k_j depend exponentially on the gas temperature through Arrhenius law

$$k_j = A_j T^{\beta_j} \exp\left(-\frac{E_j}{RT}\right)$$

where A_j is the frequency factor for reaction j , β_j is the temperature exponent and E_j is the activation energy for reaction j .

Thermodynamic Properties

The species' heat capacity, C_p , the molar enthalpy, h , and the molar entropy, s , are computed using the polynomial format of Gordon and McBride (Ref. 1). This manual also refers to these expressions as *NASA polynomials*:

$$C_{p,i} = R_g(a_1 + a_2T + a_3T^2 + a_4T^3 + a_5T^4) \quad (7-7)$$

$$h_i = R_g\left(a_1T + \frac{a_2}{2}T^2 + \frac{a_3}{3}T^3 + \frac{a_4}{4}T^4 + \frac{a_5}{5}T^5 + a_6\right) \quad (7-8)$$

$$s_i = R_g\left(a_1\ln T + a_2T + \frac{a_3}{2}T^2 + \frac{a_4}{3}T^3 + \frac{a_5}{4}T^4 + a_7\right) \quad (7-9)$$

Here, $C_{p,i}$ denotes the species' heat capacity (SI unit: J/(mol·K)), T the temperature (SI unit: K), and R_g the ideal gas constant, 8.314 J/(mol·K). Further, h_i is the species' molar enthalpy (SI unit: J/mol), and s_i represents its molar entropy (SI unit: J/(mol·K)), at standard state.

The thermodynamic properties for a given reaction are computed from the enthalpy and entropy of the participating species. The enthalpy of reaction (SI unit: J/mol) is calculated using:

$$H_j = \sum_i \nu_{ij} h_i \quad (7-10)$$

Similarly, the entropy of reaction (SI unit: J/(mol·K)) comes from the relationship

$$S_j = \sum_i \nu_{ij} s_i \quad (7-11)$$

In these equations, h_i and s_i are the species' molar enthalpy (SI unit: J/mol) and entropy (SI unit: J/(mol·K)), respectively. The heat source of reaction (SI unit: J/(m³·s)) is given as:

$$Q_j = -H_j r_j$$

The sum of Q_j over all reactions is the total heat source due to chemical reaction:

$$Q = \sum_j Q_j$$

Species Transport Properties

The diffusion coefficients are computed from the following expression based on kinetic gas theory:

$$D_{kj} = 2.66 \cdot 10^{-2} \cdot \frac{\sqrt{T^3(M_k + M_j)/(2 \cdot 10^3 M_k M_j)}}{p \sigma_k \sigma_j \Omega_D} \quad (7-12)$$

Here:

- D_{kj} is the binary diffusion coefficient (SI unit: m²/s)
- M equals the molecular weight (SI unit: kg/mol)
- T represents the temperature (SI unit: K)
- p is the pressure (SI unit: Pa), and
- σ equals the characteristic length (SI unit: Å) of the Lennard-Jones/Stockmayer potential.
- In addition, Ω_D is the collision integral, given by the following equation ([Ref. 2](#) and [Ref. 3](#)):

$$\begin{aligned} \Omega_D = & A(T^*)^{-B} + C[\exp(-DT^*)] + E[\exp(-FT^*)] \\ & + G[\exp(-HT^*)] + \frac{0.19\delta_{kj}^2}{T^*} \end{aligned} \quad (7-13)$$

where

$$\begin{aligned} T^* &= k_b \frac{T}{\epsilon_{kj}} \\ \epsilon_{kj} &= (\epsilon_k \epsilon_j)^{1/2} \\ \delta_{kj} &= (\delta_k \delta_j)^{1/2} \\ \sigma_{kj} &= (\sigma_k \sigma_j)^{1/2} \\ \delta &= \frac{1}{2} \frac{\mu^2}{\epsilon \sigma^3} \end{aligned} \quad (7-14)$$

In [Equation 7-14](#), μ is the species dipole moment (SI unit: Debye). For gases at moderate pressure, the binary diffusivity can be used for the multicomponent Maxwell-Stefan diffusivities.

The predefined expression for the dynamic viscosity is given by the kinetic gas theory ([Ref. 2](#) and [Ref. 3](#)) as

$$\mu_i = 2.669 \cdot 10^{-6} \frac{\sqrt{T(M_i \cdot 10^3)}}{\sigma_i^2 \Omega_v} \quad (7-15)$$

Here μ represents the dynamic viscosity (SI unit: N·s/m²), and Ω_v is the collision integral, given by

$$\Omega_v = A(T^*)^{-B} + C[\exp(-DT^*)] + E[\exp(-FT^*)] + \frac{0.20\delta^2}{T^*}$$

with

$$T^* = k \frac{T}{\epsilon}$$

$$\delta = \frac{1}{2} \frac{\mu^2}{\epsilon \sigma^3} \quad (7-16)$$

In Equation 7-16, μ is the species dipole moment (SI unit: Debye). The predefined expression for thermal conductivity comes from the Stiel-Thodos equation (Ref. 4)

$$k = \frac{\eta}{M} (1.15C_p + 0.88R_g) \quad (7-17)$$

where the thermal conductivity k is in (SI unit: W/(m·K)). Here C_p denotes the heat capacity (SI unit: J/(mol·K)).

Mixture Transport Properties

The transport properties of the mixture are computed using the following for specific heat capacity (SI unit: J/(kg·K))

$$c_{p\text{mix}} = \frac{\sum_i x_i C_{p,i}}{\sum_i x_i M_i}$$

The thermal conductivity (SI unit: W/(m·K)) is computed using:

$$k_{\text{mix}} = \frac{1}{2} \left[\sum_{i=1} x_i k_i + \left(\sum_{i=1} \frac{x_i}{k_i} \right)^{-1} \right]$$

In these equations:

- T is the temperature (SI unit: K)
- x is the mole fraction
- M is the molecular weight (SI unit: kg/mol)
- c_p is the heat capacity per unit mass (SI unit: J/(kg·K))
- C_p represents the heat capacity per mole (SI unit: J/(mol·K)), and
- k is the thermal conductivity (SI unit: W/(m·K)).

The dynamic viscosity (SI unit: N·s/m²) is computed using:

$$\mu_{\text{fluid}} = \sum_{i=1} \frac{x_i \mu_i}{\sum_{j=1} x_j \phi_{ij}}$$

where

$$\phi_{ij} = \frac{\left[1 + \left(\frac{\eta_i}{\eta_j} \right)^{0.5} \left(\frac{M_j}{M_i} \right)^{0.25} \right]^2}{\left[8 \left(1 + \frac{M_i}{M_j} \right) \right]^{0.5}}$$

In these equations μ denotes the dynamic viscosity (SI unit: Ns/m²).

Scaling

Because there can be a large variation in mass fraction of the components of a plasma, a logarithmic scaling of the equations is sometimes necessary. So, starting with the equation for a generalized mass transport template

$$\rho \frac{\partial}{\partial t}(w_k) + \rho(\mathbf{u} \cdot \nabla)w_k = \nabla \cdot \rho w_k \mathbf{V}_k + R_k \quad (7-18)$$

and letting $W_k = \ln w_k$, [Equation 7-18](#) becomes

$$\rho w_k \frac{\partial}{\partial t}(W_k) + \rho w_k(\mathbf{u} \cdot \nabla)W_k = \nabla \cdot [\rho w_k \mathbf{V}_k] + R_k$$

The first term on the right-hand side depends on which formulation is used. For the Mixture averaged diffusion model:

$$\mathbf{V}_k = D_{k,m} \nabla W_k + D_{k,m} \nabla \ln M + D^T \nabla \ln T - Z_k \mu_{k,m} \mathbf{E}$$

and for the **Fick's law** diffusion model:

$$\mathbf{V}_k = D_{k,f} \nabla W_k + D_{k,f} \nabla \ln M + D^T \nabla \ln T - Z_k \mu_{k,f} \mathbf{E}.$$

You can switch between the standard and logarithmic formulations of the mass balance equation. In this module the default implementation is logarithmic.

Stabilization

When the source stabilization is on, an additional term is added to the rate expression, [Equation 7-6](#). The expression takes the form:

$$R_{k,\text{tot}} = R_k + \exp(-\mathfrak{t} \ln(n_k))$$

where \mathfrak{t} is the user-defined tuning parameter. When the species number density is very low, this acts as a source term which prevents the mass fraction from approaching zero. As the number density increases this term becomes exponentially smaller, eventually becoming negligible for high number densities.

Surface Reactions

The equations solved for the transport for each species need to be supplemented by appropriate boundary conditions. This is often accomplished by way of a set of suitable surface reactions, which describe how species are created or consumed on surfaces. Assume a surface mechanism with I surface reactions and K chemical species

$$\sum_{k=1}^K \nu_{ki}^f c_k \Leftrightarrow \sum_{k=1}^K \nu_{ki}^r c_k$$

where ν_{ki} are the stoichiometric coefficients and c_k is the concentration of the k th species. The surface rate expression (SI unit: $\text{mol}/(\text{m}^2 \cdot \text{s})$) for species k is defined as

$$R_{\text{surf},k} = \sum_{i=1}^I \nu_{ki} q_i$$

where $\nu_{ki} = \nu_{ki}^r - \nu_{ki}^f$. The rate of progress of the i th reaction is governed by the law of mass action:

$$q_i = k_{f,i} \prod_{k=1}^K c_k^{\gamma_{ki}^f} - k_{r,i} \prod_{k=1}^K c_k^{\gamma_{ki}^r}$$

where the following definitions apply for c_k .

Gas Phase Species

The molar concentration is given by:

$$c_k = \frac{w_k \rho}{M_k}$$

where w_k is the mass fraction of species k , ρ is the gas density, and M_k is the molecular weight of species k . When the Motz-Wise correction option is set to On, the forward rate constant is related to the sticking coefficients via:

$$k_{f,i} = \left(\frac{\gamma_i}{1 - \gamma_i/2} \right) \frac{1}{(\Gamma_{\text{tot}})^m} \left(\frac{1}{4} \right) \sqrt{\frac{8RT}{\pi M_n}} \quad (7-19)$$

When the Motz-Wise correction option is set to Off, the forward rate constant is given via:

$$k_{f,i} = \frac{\gamma_i}{(\Gamma_{\text{tot}})^m} \left(\frac{1}{4} \right) \sqrt{\frac{8RT}{\pi M_n}}. \quad (7-20)$$

where Γ_{tot} is the total surface site concentration (SI unit: mol/m²). The *sticking coefficient* can be a constant or even temperature dependent through an Arrhenius expression:

$$\gamma_i = a_i T^{b_i} \exp\left(-\frac{e_i}{RT}\right)$$

where a , b , and e are constants. Since γ is a probability it must be between 0 and 1 to be physically meaningful.

Flux Matching

Flux matching must be performed for each species in order for the total molar mass in the system to be conserved. If a model also contains a physics interface to compute the fluid velocity and pressure (the Laminar Flow interface for example) and the gas temperature (the Heat Transfer in Fluids interface for example) then those physics interfaces require appropriate flux matching boundary conditions. The Heavy Species

Transport interface automatically computes flux matching variables for the total mass flux and heat source of reaction on surfaces.



Surface Chemistry Tutorial: Application Library path
Plasma_Module/Chemical_Vapor_Deposition/surface_chemistry_tutorial

FLUX MATCHING FOR EACH SPECIES

The inward mass flux for species k on a reacting surface is given by:

$$-\mathbf{n} \cdot \mathbf{j}_k = M_k R_{\text{surf}, k} \quad (7-21)$$

This is the natural boundary condition for [Equation 7-3](#) and results in either an inwards or outward mass flux for species k , depending on the surface chemical mechanism stoichiometry. If the species is an ion, then there is an additional flux contribution due to migration. In this case, [Equation 7-21](#) becomes:

$$-\mathbf{n} \cdot \mathbf{j}_k = M_k R_{\text{surf}, k} + M_k c_k u_{m, k} z_k (\mathbf{n} \cdot \mathbf{E}) [(z_k \mathbf{n} \cdot \mathbf{E}) > 0]$$

This additional flux contribution ensures that when the normal electric field is directed to the wall, there is an outflux of ions due to migration.

FLUX MATCHING FOR THE TOTAL MASS FLUX

If a model contains the Laminar Flow interface, an expression for the mass averaged velocity is needed on reacting surfaces. The Heavy Species Transport interface creates an appropriate variable for the inward mass flux, which can be accepted as a feature input for the Mass Flow option in the Inlet boundary condition of the Laminar Flow interface.

The inward mass flux, M_f is defined as:

$$M_f = \sum_{k=1}^{K_g} M_k R_{\text{surf}, k}$$

where $R_{\text{surf}, k}$ is the surface rate expression for each species which comes from summing the surface reaction rates multiplied by their stoichiometric coefficients over all surface reactions:

$$R_{\text{surf},k} = \sum_{i=1}^I \nu_{ki} q_i.$$

FLUX MATCHING FOR THE TOTAL HEAT FLUX

If a model contains the Heat Transfer in Fluids interface, an expression for the total heat source of reactions is needed. The Heavy Species Transport interface creates an appropriate variable for the total heat source of reactions, which can be accepted as a feature input in the Reaction Heat Flux boundary condition of the Heat Transfer in Fluids interface.

The total heat source due to reactions is defined as:

$$Q_b = \sum_{i=1}^I q_i h_i$$

where h_i is the molar enthalpy change due to reaction i . Depending on the surface chemical mechanism stoichiometry, this can either cause heating (exothermic) or cooling (endothermic).

About Surface Species

For each of the k [Surface Species](#) the following equations are solved:

$$\frac{d\Gamma_k}{dt} = \sum_{i=1}^N q_i \Delta\sigma_i$$

where Γ_k is the surface site concentration (SI unit: mol/m²), q_i is the reaction rate for reaction i (SI unit: mol/m²), and $\Delta\sigma_i$ is the change in site occupancy number for reaction i (dimensionless).

$$\frac{dZ_k}{dt} = \frac{R_{\text{surf},k}}{\Gamma_{\text{tot}}}$$

where Z_k is the site fraction (dimensionless), and $R_{\text{surf},k}$ is the surface rate expression (SI unit: mol/m²). The quantity Z_k is the surface equivalent of the mole fraction on the volumetric level. That is, the sum of the site fractions must always equal one:

$$\sum Z_k = 1.$$

For each of the bulk surface species, the following equation is solved for the deposition height:

$$\frac{dh_k}{dt} = -\frac{R_{\text{surf},k} M_k}{\rho_k}$$

where h_k is the total growth height (SI unit: m), M_k is the molecular weight (SI unit: kg/mol), and ρ is the density of the bulk species (SI unit: kg/m³).

Limitations of the Heavy Species Transport Equations

Before continuing, some comments on the validity of the equations are necessary. Despite their relatively complicated form, the equations for transport of the heavy species are formulated under the following assumptions:


- The diffusion velocity must be much less than the thermal velocity; that is, $\mathbf{V}_k \ll v_k$. In the case of very low pressure CCP reactors, the ion drift velocity can approach or exceed the ion thermal velocity. In this case, the heavy species transport equations fail to fit the physics of the discharge.
- A single temperature is assumed for heavy species. In the case of reactors with high reduced electric fields, the ions can have a different temperature than the surrounding gas.

References for the Heavy Species Transport Interface

1. S. Gordon and B.J. McBride, *Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouquet Detonations*, NASA-SP-273, 1971.
2. P.D. Neufeld, A.R. Janzen, and R.A. Aziz, "Empirical Equations to Calculate 16 of the Transport Collision Integrals $\Omega^{(l,s)*}$ for the Lennard-Jones (12-6) Potential," *J. Chem. Phys.*, vol. 57, p. 1100, 1972.
3. R.S. Brokaw, "Predicting Transport Properties of Dilute Gases," *Ind. Eng. Process Design Develop.*, vol. 8, p. 240-253, 1969.
4. L.I. Stiel and G. Thodos, "The Viscosity of Polar Substances in the Dense Gaseous and Liquid Regions," *AIChE J.*, vol. 10, p. 26, 1964.

5. R.J. Kee, M.E. Coltrin, and P. Glarborg, *Chemically Reacting Flow Theory and Practice*, Wiley, 2003.
6. R.B. Bird, W.E. Stewart, and E.N. Lightfoot, *Transport Phenomena*, John Wiley & Sons, 2002.
7. G.J.M. Hagelaar, *Modeling Of Microdischarges for Display Technology*, PhD thesis, Universiteitsdrukkerij Technische Universiteit Eindhoven, 2000.



Plasma Interfaces

This chapter describes the Plasma Module interfaces found under the **Plasma** branch ()

In this chapter:

- [The Plasma Interface](#)
- [The Inductively Coupled Plasma Interface](#)
- [The Microwave Plasma Interface](#)
- [Plasma Reactors Theory](#)
- [Capacitively Coupled Plasmas](#)
- [DC Discharges](#)
- [Theory for Inductively Coupled Plasmas](#)

The Plasma Interface

The **Plasma (plas)** interface () found under the **Plasma** branch () couples the Drift Diffusion, Heavy Species Transport, and Electrostatics interfaces into an integrated multiphysics interface to model capacitively coupled plasmas (CCP) and DC discharges.

When this physics interface is added, these default nodes are also added to the **Model Builder**—**Plasma Model**, **Zero Charge**, **Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and velocity. You can also right-click **Plasma** to select physics features from the context menu.



Except where described in this section, this physics interface shares its physics nodes with [The Drift Diffusion Interface](#), [The Heavy Species Transport Interface](#), and [The Electrostatics Interface](#).

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (`_`) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `plas`.

OUT-OF-PLANE THICKNESS (2D, 2D AXISYMMETRIC, AND 1D)

- For 2D components, enter a **Thickness** d (SI unit: m). The default is 1 m.
- For 2D axisymmetric components, enter a **Vertical height** d_z (SI unit: m). The default is 0.01 m.
- For 1D components, enter a **Cross-section area** A (SI unit: m^2). The default is 0.01 m^2 .

TRANSPORT SETTINGS

Select a **Diffusion model**—**Mixture-averaged** (the default) or **Fick's law**. When using the **Mixture-averaged** model, the mixture averaged diffusion coefficients are automatically computed based on the data specified for each species.

Select the check boxes for which transport mechanisms to **Include**—**Convection**, **Migration in electric field**, **Calculate thermodynamic properties**, **Full expression for diffusivity**, or **Compute tensor ion transport properties**. The selection changes the number of [Model Inputs](#) requiring values on the **Plasma Model** page. Note the following:

- The **Migration** term is part of the relative mass flux vector.
- **Convection** of heavy species present in a plasma can often be neglected due to the low operating pressure.
- For **Calculate thermodynamic properties** select that the thermodynamic properties of each reaction and species are computed automatically based on the thermodynamic properties of each species.
- For **Full expression for diffusivity** it computes a more accurate expression for the Maxwell-Stefan diffusivities. Often the additional correction terms are negligible in which case the expressions are much simpler and the time taken to assemble the Jacobian matrix is reduced.
- For **Compute tensor ion transport properties** the tensor form of the ion transport properties when a static magnetic field is present is computed. This option only needs to be activated when a strong DC magnetic field exists and the operating pressure is very low (on the order of millitorr). When this option is activated an expression must be provided for the magnetic flux density, which typically another physics interface computes. This is set in the **Plasma Model** node.

PLASMA PROPERTIES

Select the **Compute tensor electron transport properties**, **Use reduced electron transport properties** and **Fix mean electron energy** check boxes as needed.

Select **Compute tensor electron transport properties** to compute the tensor form of the electron mobility, electron diffusivity, energy mobility and energy diffusivity. This should only be used in cases where a strong DC magnetic field exists. Two quantities must be supplied, both of which are in the [Plasma Model](#) node. The DC mobility which is the value of the electron mobility in the absence of a DC magnetic field and the magnetic flux density which would typically be computed by another physics interface.

Select **Use reduced electron transport properties** to specify the electron mobility, diffusivity, energy mobility and energy diffusivity in reduced form. The neutral number density is then specified in the [Drift Diffusion Model](#) node. The electron transport properties are computed from the reduced transport properties using:

$$\mu_e = \frac{\mu_{\text{red}}}{N_n}$$

where N_n is the user-defined neutral number density.

The **Include thermal diffusion** check box adds an additional term to the definition of the electron current due to gradients in the electron diffusivity. If the diffusivity is a constant then including this does not effect the solution. It is only necessary to include this term if the electron diffusivity is a function of the electron temperature, and there are significant gradients in the electron temperature.

Select **Fix mean electron energy** to fix the mean electron energy to its initial value. This can be useful in some situations because the strong coupling between the mean electron energy and the electromagnetic fields is removed. This allows for non-self consistent models to be created quickly, since problems where the mean electron energy is fixed are easier to solve numerically.



For more information on the electron energy distribution function (EEDF), see the [Theory for the Boltzmann Equation, Two-Term Approximation Interface](#).

ELECTRON ENERGY DISTRIBUTION FUNCTION

If cross section data is used to define source coefficients in the model then an electron energy distribution function (EEDF) needs to be selected. Select one of these options.

- **Maxwellian.** This option assumes a Maxwellian EEDF which takes the form:

$$f(\epsilon) = \varphi^{-3/2} \beta_1 \exp(-(\epsilon \beta_2 / \varphi))$$

where

$$\beta_1 = \Gamma(5/2)^{3/2} \Gamma(3/2)^{-5/2}, \beta_2 = \Gamma(5/2) \Gamma(3/2)^{-1}$$

where φ is the mean electron energy (SI unit: eV), ϵ is the electron energy (SI unit: eV) and Γ is the incomplete gamma function:

$$\Gamma(s) = \int_0^{\infty} u^{s-1} e^{-u} du .$$

- **Druyvesteyn.** This option assumes a Druyvesteyn EEDF which takes the form:

$$f(\varepsilon) = 2\varphi^{-3/2}\beta_1 \exp(-(\varepsilon\beta_2/\varphi)^2)$$

where

$$\beta_1 = \Gamma(5/4)^{3/2}\Gamma(3/4)^{-5/2}, \beta_2 = \Gamma(5/4)\Gamma(3/4)^{-1}$$

- **Generalized.** Use this option for a generalized distribution function where the EEDF is somewhere between Maxwellian and Druyvesteyn. For this option, specify a power law. This number must be between 1 and 2. Mathematically, the EEDF takes the form:

$$f(\varepsilon) = g\varphi^{-3/2}\beta_1 \exp(-(\varepsilon\beta_2/\varphi)^g)$$

where

$$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2}, \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}, 1 \leq g \leq 2$$

- **Function.** If a two-dimensional interpolation function has been added to the model, it can be used for the EEDF. In this case, the x -data should be the electron energy (eV) and the y -data should be the mean electron energy (eV).



The two-dimensional interpolation function can be computed using a parametric sweep in [The Boltzmann Equation, Two-Term Approximation Interface](#). This allows for modeling of discharges where the EEDF is far from Maxwellian. For step-by-step instructions on how to do this, refer to this blog entry:

<https://www.comsol.com/blogs/the-boltzmann-equation-two-term-approximation-interface/>

See [Interpolation](#) in the *COMSOL Multiphysics Reference Manual*.

In all these cases the rate constants in the model are automatically computed based on the selected EEDF using the formula:

$$k_f = \gamma \int_0^{\infty} \epsilon \sigma_k(\epsilon) f(\epsilon) d\epsilon . \quad (8-1)$$

The rate coefficients when computed using cross section data are a highly nonlinear function of the mean electron energy. COMSOL Multiphysics automatically computes the integral in [Equation 8-1](#) and makes the result available for evaluation of the rate coefficient. The variation of the rate coefficient for any particular model can be plotted using `<name>.kf_<reaction number>`. For example, for reaction number 3 in the Plasma interface, with name `plas`, the rate coefficient is plotted using `plas.kf_3`.

CHEMKIN IMPORT FOR SPECIES PROPERTY

Import thermodynamic and transport properties for the reacting species into the model.




The thermodynamic and transport input files must follow the CHEMKIN[®] standard.

STABILIZATION

If the **Equation formulation** is set to **Log** then the solver can run into difficulties as the species mass fractions approach zero. The **Reaction source stabilization** check box (selected by default) adds an additional source term to the rate expression for each species. In the τ field, enter a tuning parameter for the source stabilization. The default value is 1. This value is usually good enough. If the plasma is high pressure (atmospheric) then it can help to lower this number to somewhere in the range of 0.25–0.5.

The solver can also run into difficulties as the electron density or electron energy density approach zero. The **Source stabilization** check box (selected by default) adds an additional source term to the equation for the electron density and electron energy density. In the ζ field, enter a tuning parameter for the source stabilization. The default value is 1. This value is usually good enough. If the plasma is high pressure (atmospheric) then it can help to lower this number to somewhere in the range of 0.25–0.5.

DISCRETIZATION

To display this section, click the **Show** button () and select **Discretization**.



Select a **Formulation**—**Finite element** (the default) or **Finite volume**. The **Finite element** option uses the Galerkin method to discretize the equations whereas the **Finite volume**

option creates degrees of freedom for the dependent variables which are piecewise constant within each mesh element. For the charged species, Scharfetter–Gummel upwinding is used. For charge neutral species and the electrostatic field, a centered difference scheme is used. In cases where the ion and electron flux are strongly driven by the electric field, the **Finite volume** option can be more stable. Examples where this is true include dielectric barrier and corona discharges.

When the **Formulation** is set to **Finite element**, select an **Equation formulation—Log** (the default) to solve the equations in logarithmic form or **Linear** to solve equations in the original form. **Log** solves for the log of the dependent variables, ensuring that the mass fraction of any of the species is never lower than zero. This makes it more numerically stable but increases the nonlinearity of the equation system, and as such the model might take slightly longer to solve.

DEPENDENT VARIABLES

The dependent variables (field variables) are the **Electron potential**, **Log of electron density**, and **Log of electron energy density**. The name can be changed but the names of fields and dependent variables must be unique within a model.

	<ul style="list-style-type: none"> • Domain, Boundary, Edge, Point, and Pair Nodes for the Plasma Interface • Plasma Reactors Theory and Capacitively Coupled Plasmas • Stabilization • Interpolation in the <i>COMSOL Multiphysics Reference Manual</i>
	<ul style="list-style-type: none"> • <i>GEC CCP Reactor, Argon Chemistry, 1D</i>: Application Library path Plasma_Module/Capacitively_Coupled_Plasmas/argon_gec_ccp_1d • <i>Harmonic Content of the Power Deposition into a Dual Frequency Capacitively Coupled Plasma</i>: Application Library path Plasma_Module/Capacitively_Coupled_Plasmas/harmonic_content • <i>DC Glow Discharge</i>: Application Library path Plasma_Module/Direct_Current_Discharges/positive_column_2d

Domain, Boundary, Edge, Point, and Pair Nodes for the Plasma Interface

The **Plasma Interface** has these domain, boundary, edge, point, and pair nodes, listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).



In general, to add a node, go to the **Physics** toolbar, no matter what operating system you are using. Subnodes are available by clicking the parent node and selecting it from the **Attributes** menu.



In the *COMSOL Multiphysics Reference Manual* see [Table 2-3](#) for links to common sections and [Table 2-4](#) to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

These nodes are described in this section:

- [Plasma Model](#)
- [Initial Values](#)

These nodes are described for the Drift Diffusion or Heavy Species Transport interface (listed in alphabetical order):

- [Cross Section Import](#)
- [Electron Density and Energy](#)
- [Electron Impact Reaction](#)
- [Electron Outlet](#)
- [Electron Production Rate](#)
- [Flux](#)
- [General Power Deposition](#)
- [Insulation](#)
- [Reaction](#)
- [Species](#)
- [Surface Reaction](#)
- [Surface Species](#)
- [Velocity](#)
- [Wall](#)

These nodes are described for the Electrostatics interface (listed in alphabetical order):

- Dielectric Contact
- Dielectric Shielding
- Distributed Capacitance
- Electric Displacement Field
- Electric Potential
- Electrostatic Point Dipole
- Floating Potential
- Ground
- Line Charge
- Metal Contact
- Periodic Condition
- Point Charge
- Surface Charge Accumulation
- Surface Charge Density
- Terminal
- Thin Low Permittivity Gap
- Zero Charge

Plasma Model

Use the **Plasma Model** node to set the electron transport properties, material properties, and model inputs.

MODEL INPUTS

This section has fields and values that are inputs to expressions that define material properties. If such user-defined property groups are added, the model inputs display here. Initially, this section is empty. The input depends on which **Transport Settings** are selected in the **Plasma** interface.

- Enter a **Temperature** T (SI unit: K) for the gas mixture used to compute the gas density and species transport properties.
- Enter an **Absolute pressure** p_A (SI unit: Pa) to compute the gas density and the Maxwell-Stefan diffusivities.
- If the **Convection** check box is selected, enter coordinates for the **Velocity field** \mathbf{u} (SI unit: m/s).
- If the **Compute tensor electron transport properties** or the **Compute tensor ion transport properties** property is activated enter an expression for the **Magnetic flux density** B (SI unit: T). This usually comes from a magnetostatic model, and is used to compute anisotropic transport properties for electrons and ions.

DENSITY

Select an option from the **Mixture density** list—**Ideal gas** or **User defined**. For **User defined** enter a value for ρ (SI unit: kg/m^3).

For **Ideal gas** it computes the expression for the gas density automatically using the ideal gas law, based on the composition of the mixture. The density is a function of the mean molecular weight:

$$\rho = \frac{p_A M}{RT}$$

where p_A is the absolute pressure, T is the temperature, R is the universal gas constant and M is the mean molecular weight given by:

$$\frac{1}{M} = \sum_{k=1}^Q \frac{w_k}{M_k}$$

ELECTRON DENSITY AND ENERGY

Select an option from the **Electron transport properties** list—**Specify mobility only** (the default), **Specify all**, or **Use lookup tables**. The options available depend on whether the **Use reduced electron transport properties** physics interface property is checked. If the physics interface property Compute tensor electron transport properties is checked then only one field appears for the DC or reduced DC electron mobility, $\mu_{dc} N_n$. The other transport properties are computed using Einstein's relation.



Tensor Transport Properties

- For **Specify mobility only** enter a value or expression for the **Electron mobility** μ_e (SI unit: $\text{m}^2/(\text{V}\cdot\text{s})$). Select **Isotropic**, **Diagonal**, **Symmetric** or **Anisotropic** based on the model. The electron diffusivity, energy mobility and energy diffusivity are automatically computed using Einstein's relation for a Maxwellian EEDF:

$$\mathbf{D}_e = \mu_e T_e, \mu_\varepsilon = \left(\frac{5}{3}\right) \mu_e, \mathbf{D}_\varepsilon = \mu_\varepsilon T_e \quad (8-2)$$

For **Use reduced electron transport properties** enter the **Reduced electron mobility** $\mu_e N_n$ (SI unit: $1/(\text{V}\cdot\text{m}\cdot\text{s})$). The actual electron mobility is then computed by dividing the reduced electron mobility by the neutral number density.

- For **Specify all** enter values or expressions for all of the properties, which can be either scalars or tensors. The appropriate values for the transport properties can be computed with [The Boltzmann Equation, Two-Term Approximation Interface](#). For all of the properties, select **Isotropic**, **Diagonal**, **Symmetric** or **Anisotropic** from the list based on the model, then enter values or expressions for all of these properties:

- **Electron mobility** μ_e (SI unit: $\text{m}^2/(\text{V}\cdot\text{s})$)
- **Electron diffusivity** D_e (SI unit: m^2/s)
- **Electron energy diffusivity** D_{en} (SI unit: m^2/s)
- **Electron energy mobility** μ_{en} (SI unit: $\text{m}^2/(\text{V}\cdot\text{s})$)

For **Use reduced electron transport properties** enter:

- **Reduced electron mobility** $\mu_e N_n$ (SI unit: $1/(\text{V}\cdot\text{m}\cdot\text{s})$)
- **Reduced electron diffusivity** $D_e N_n$ (SI unit: $1/(\text{m}\cdot\text{s})$)
- **Reduced electron energy diffusivity** $D_{\text{en}} N_n$ (SI unit: $1/(\text{m}\cdot\text{s})$)
- **Reduced electron energy diffusivity** $\mu_{\text{en}} N_n$ (SI unit: $1/(\text{V}\cdot\text{m}\cdot\text{s})$)

For **Use lookup tables** enter or load a lookup table with the transport properties as listed above versus mean electron energy (eV).



For the **Use reduced electron transport properties** physics interface property, the appropriate values for the transport properties can be computed with [The Boltzmann Equation, Two-Term Approximation Interface](#) and written out to a text file. The text file can then be loaded in for each electron transport property.

Initial Values

Use the **Initial Values** node to set the initial values of the electron density, mean electron energy, and electric potential.



Initial conditions for the heavy species are specified on the individual species features.

INITIAL VALUES

Enter values or expressions for each of the following:



- **Initial electron density** $n_{e,0}$ (SI unit: $1/\text{m}^3$). This value represents some initial seed electrons in the plasma. If this number is too low then the plasma might not sustain itself. If this number is too high then numerical instabilities can arise at early time steps.
- **Initial mean electron energy** e_0 (SI unit: V). Enter an approximate guess for the mean electron energy in the reactor. COMSOL Multiphysics automatically computes the exact mean electron energy after solving but the closer the guess is set to the expected value the easier it is to find a solution.
- **Electric potential** V (SI unit: V). In nearly all cases the value should be zero. This initial value must also be consistent with any boundary conditions on the electric potential in the model.

The Inductively Coupled Plasma Interface

The **Inductively Coupled Plasma** interface is used to study discharges that are sustained by induction currents. The induction currents are solved for in the frequency domain. The electron heating due to the induction currents is automatically handled by the software. This physics interface is available for 2D and 2D axisymmetric components.



With an AC/DC Module license, this physics interface is also available in 3D. This also enables additional features that are available with the AC/DC Module, and described in the *AC/DC Module User's Guide*.

When a predefined **Inductively Coupled Plasma** interface () is added from the **Plasma** branch () of the **Model Wizard** or **Add Physics** windows, **Plasma** and **Magnetic Fields** interfaces are added to the Model Builder. In addition, the **Multiphysics** node is added, which automatically includes the multiphysics coupling features **Plasma Conductivity Coupling** and **Electron Heat Source**.

When the **Plasma** physics interface is added, these default nodes are also added to the **Model Builder**—**Plasma Model**, **Zero Charge**, **Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, the plasma chemistry and boundary conditions.

When the **Magnetic Fields** interface is added, these default nodes are also added to the **Model Builder**—**Ampère's Law**, **Magnetic Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, a coil.

SETTINGS

The settings for the **Plasma** interface can be found in [The Plasma Interface](#) section. The settings for the **Magnetic Fields** interface can be found in the [The Magnetic Fields Interface](#) in the *AC/DC Module User's Guide*.

These nodes are described in this section:

- [Plasma Conductivity Coupling](#)
- [Electron Heat Source](#)
- [Collisionless Heating](#)

Plasma Conductivity Coupling

The **Plasma Conductivity Coupling** computes the electrical conductivity of the plasma based on the cold plasma approximation. The electron density and collision frequency are provided by the **Plasma** interface, and the angular frequency by the **Magnetic Fields** or **Electromagnetic Waves, Frequency Domain** interfaces.

COMPUTE TENSOR PLASMA CONDUCTIVITY

If a static magnetic field is present, and the pressure is suitably low, it may be necessary to use a full tensor for the plasma conductivity. This can result in a non-linear and anisotropic plasma conductivity, as is the case in electron cyclotron resonance. When **Compute tensor plasma conductivity** is active, enter a value for the **Doppler broadening parameter**. This can smooth resonance zones where the power deposition is focused over a very small region. The lower the value entered, the more smoothing occurs. More information can be found in the [resonance zone](#) section.

COUPLED INTERFACES

The selection for this multiphysics coupling is locked, and corresponds to the intersection of the **Magnetic Fields** or **Electromagnetic Waves, Frequency Domain** and **Plasma** interfaces specified in the **Coupled Interfaces** section.

Electron Heat Source

The **Electron Heat Source** coupling applies the heat source from the **Magnetic Fields** or **Electromagnetic Waves, Frequency Domain** to the electrons in the **Plasma** interface. This corresponds to inductive heating or wave-heating of the electrons respectively.

COUPLED INTERFACES

The selection for this multiphysics coupling is locked, and corresponds to the intersection of the **Magnetic Fields** or **Electromagnetic Waves, Frequency Domain** and **Plasma** interfaces specified in the **Coupled Interfaces** section.

Collisionless Heating

The **Collisionless Heating** multiphysics node includes the non-local heating effects via an effective viscosity approximation. Typically this is only important at very low pressures (less than 2 Pa) and frequencies above 10 MHz.

COUPLED INTERFACES



The selection for this multiphysics coupling is locked, and corresponds to the intersection of the **Magnetic Fields** and **Plasma** interfaces specified in the **Coupled Interfaces** section.

The Microwave Plasma Interface



This physics interface requires a RF Module license. For theory and physics interface feature descriptions, see the *RF Module User's Guide*.

The **Microwave Plasma** physics interface is suitable for studying discharges that are sustained by electromagnetic waves (wave-heated discharge). The electromagnetic waves are solved for in the frequency domain, and all plasma variables are solved for in the time-domain.

When a predefined **Microwave Plasma** interface () is added from the **Plasma** branch () of the **Model Wizard** or **Add Physics** windows, **Plasma** and **Electromagnetic Waves, Frequency Domain** interfaces are added to the Model Builder. In addition, the **Multiphysics** node is added, which automatically includes the multiphysics coupling features **Plasma Conductivity Coupling** and **Electron Heat Source**.

When the **Plasma** physics interface is added, these default nodes are also added to the **Model Builder**—**Plasma Model**, **Zero Charge**, **Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, the plasma chemistry and boundary conditions.

When the **Electromagnetic Waves, Frequency Domain** interface is added, these default nodes are also added to the **Model Builder**—**Wave Equation**, **Electric**, **Perfect Electric Conductor**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, a port.

SETTINGS

The settings for the **Plasma** interface can be found in [The Plasma Interface](#) section. The settings for the **Electromagnetic Waves, Frequency Domain** interface can be found in the [The Electromagnetic Waves, Frequency Domain Interface](#) in the *RF Module User's Guide*.

For a description of the multiphysics nodes, refer to:

- [Plasma Conductivity Coupling](#)
- [Electron Heat Source](#)

Plasma Reactors Theory

The Plasma interface couples the Drift Diffusion, Heavy Species and Electrostatic interfaces. A brief explanation of the underlying equations is given here.



For detailed information about electron transport see [Theory for the Drift Diffusion Interface](#).

For detailed information about the transport of the non-electron species see [Theory for the Heavy Species Transport Interface](#).

For detailed information about electrostatics see [The Electrostatics Interface](#).

The electron density and mean electron energy are computed by solving a pair of drift-diffusion equations for the electron density and mean electron energy. Convection of electrons due to fluid motion is neglected.

$$\frac{\partial}{\partial t}(n_e) + \nabla \cdot [-n_e(\boldsymbol{\mu}_e \bullet \mathbf{E}) - \nabla(\mathbf{D}_e n_e)] = R_e$$

$$\frac{\partial}{\partial t}(n_e) + \nabla \cdot [-n_e(\boldsymbol{\mu}_e \bullet \mathbf{E}) - \nabla(\mathbf{D}_e n_e)] + \mathbf{E} \cdot \boldsymbol{\Gamma}_e = R_e$$

The electron source R_e and the energy loss due to inelastic collisions R_e are defined later. If a strong DC magnetic field is present then the electron mobility can be a tensor:

$$\boldsymbol{\mu}_e^{-1} = \begin{bmatrix} \frac{1}{\mu_{dc}} & -B_z & B_y \\ B_z & \frac{1}{\mu_{dc}} & -B_x \\ -B_y & B_x & \frac{1}{\mu_{dc}} \end{bmatrix}$$

where the inverse of the mobility has been used because the actual expression for the electron mobility cannot be written in a compact form. The quantity μ_{dc} is the electron mobility in the absence of a magnetic field. The electron diffusivity, energy mobility and energy diffusivity are then calculated using:

$$\mathbf{D}_e = \mu_e T_e, \mu_\epsilon = \left(\frac{5}{3}\right) \mu_e, \mathbf{D}_\epsilon = \mu_\epsilon T_e$$

The source coefficients in the above equations are determined by the plasma chemistry and are written using either rate or Townsend coefficients. Suppose that there are M reactions which contribute to the growth or decay of electron density and P inelastic electron-neutral collisions. In general $P \gg M$. In the case of rate coefficients, the electron source term is given by

$$R_e = \sum_{j=1}^M x_j k_j N_n n_e$$

where x_j is the mole fraction of the target species for reaction j , k_j is the rate coefficient for reaction j (SI unit: m^3/s), and N_n is the total neutral number density (SI unit: $1/\text{m}^3$). When Townsend coefficients are used, the source term becomes

$$R_e = \sum_{j=1}^M x_j \alpha_j N_n |\Gamma_e|$$

where α_j is the Townsend coefficient for reaction j (SI unit: m^2) and Γ_e is the electron flux (SI unit: $1/(\text{m}^2 \cdot \text{s})$). *Townsend coefficients* can increase the stability of the numerical scheme when the electron flux is field driven as is the case with DC discharges.

The electron energy loss is obtained by summing the collisional energy loss over all reactions:

$$R_\epsilon = \sum_{j=1}^P x_j k_j N_n n_e \Delta \epsilon_j$$

where $\Delta \epsilon_j$ is the energy loss from reaction j (SI unit: V). In the case of Townsend coefficients, the energy loss is given by

$$R_\epsilon = \sum_{j=1}^P x_j \alpha_j N_n |\Gamma_e| \Delta \epsilon_j$$

The electron source and inelastic energy loss are automatically computed by the multiphysics interface. The *rate coefficients* can be computed from cross section data by the following integral:

$$k_k = \gamma \int_0^{\infty} \epsilon \sigma_k(\epsilon) f(\epsilon) d\epsilon$$

where:

- $\gamma = (2q/m_e)^{1/2}$ ($C^{1/2}/kg^{1/2}$)
- m_e is the electron mass (SI unit: kg)
- ϵ is the energy (SI unit: V)
- σ_k is the collision cross section (SI unit: m^2), and
- f is the electron energy distribution function. Choose between Maxwellian, Druyvesteyn, or a Generalized EEDF.

For non-electron species, the following equation is solved for the mass fraction of each species.

$$\rho \frac{\partial}{\partial t}(w_k) + \rho(\mathbf{u} \cdot \nabla)w_k = \nabla \cdot \mathbf{j}_k + R_k$$

The electrostatic field is computed using the following equation:

$$-\nabla \cdot \epsilon_0 \epsilon_r \nabla V = \rho$$

The *space charge density*, ρ is automatically computed based on the plasma chemistry specified in the model using the formula:

$$\rho = q \left(\sum_{k=1}^N Z_k n_k - n_e \right).$$

Capacitively Coupled Plasmas

Capacitively coupled plasmas (CCP) represent a monumental challenge when it comes to numerical modeling for the following reasons:

- The method of power deposition into the plasma from the electric field is a highly nonlinear process meaning that the problem must be solved in the time domain. This requires the RF cycle to be accurately resolved over time. Because a capacitively coupled plasma can take thousands of RF cycles to reach a periodic steady state solution, it can take hours or even days for a problem to solve.
- String electric fields, which result in the plasma sheath, can lead to very high *Peclet numbers*. This means that the numerical scheme can become unstable unless the mesh is sufficiently fine in the plasma sheath.
- The external circuit connecting a CCP reactor to a power supply nearly always needs to be included in the model.
- Sharp geometric corners lead to substantial local electric fields that can result in nonphysical ion fluxes.

When the [Metal Contact](#) feature is active, it is possible to include quick options for the most common types of electrical circuit. The **Circuit type** has the following options, which affect the value of the electric potential imposed on the electrode. When the **Circuit type** is anything other than **None**, the electric potential imposed on the electrode, V is not necessary equal to the user supplied voltage, V_0 .



The Electrostatics Interface

NONE

When the circuit type is set to None, the electric potential imposed at the electrode is equal to the user supplied electric potential:

$$V = V_0.$$

BALLAST RESISTOR

This option places a ballast resistor between the user supplied electric potential and the applied voltage. In this case, the imposed value of the electric potential on the electrode becomes:

$$V = V_0 - I_p R_b$$

where R_b is the value of the ballast resistor, and I_p is the total current flowing into the electrode. The current is computed by integrating the normal component of the electron and ion current densities over the electrode surface. This boundary condition should be used to prevent arcing in a DC discharge. If the plasma current becomes high, the magnitude of the applied voltage is reduced, thus avoiding a positive feedback loop where the plasma densities become extremely high.

SERIES RC CIRCUIT

This option places a ballast resistor and blocking capacitor between the user supplied electric potential and the applied voltage. In steady state, this has the same effect as the ballast resistor option. One advantage of this option is that the imposed voltage on the electrode is smoothly ramped up from 0 to the setpoint in a time constant determined by the value of the ballast resistor and blocking capacitor. This can help with convergence when solving because it results in initial conditions which are consistent with the boundary conditions. The imposed value of the electric potential on the electrode becomes:

$$V = V_0 - I_p R_b - R_b C_b \frac{dV}{dt}.$$

This option is the preferred option to use when modeling DC discharges.

BLOCKING CAPACITOR

Another less useful option is to include a blocking capacitor in between the electrode and voltage source. In this case, the imposed value of the electric potential is given by the equation:

$$C_b \frac{d}{dt}(V_0 - V) = I_p$$

where I_p is the current due to motion of charged carriers and the displacement current.

Computing the DC Bias

The DC bias in a capacitively coupled plasma can be computed using the **Terminal** boundary condition and setting the **Terminal type** to **Current**. In this case, the following integral constraint is enforced:

$$\int_{\partial\Omega} \frac{\partial}{\partial t} (\mathbf{n} \cdot \mathbf{D}) dS = I_0 .$$

where I_0 is the user defined expression for the current. This constraint is enforced by introducing a Lagrange multiplier for the electric potential on this boundary. The software adjusts the value of the electric potential such that the above constraint is satisfied. Since the constraint involves the time derivative of the electric displacement, the **Use weak constraints** check box needs to be selected. This produces a much more accurate value for the integral in the above equation.

Once the plasma has reached its periodic steady state solution, the DC bias can be computed by evaluating the time averaged value of the electric potential on the terminal over 1 RF cycle.

DC Discharges

The theory for modeling DC discharges is described in this section:

- [Introduction to DC Discharge Theory](#)
- [DC Discharge Boundary Conditions](#)
- [Tips for Modeling DC Discharges](#)

Introduction to DC Discharge Theory

Due to the complexity of coupling the electrostatic field to the transport of electrons and heavy species, the Plasma Module provides the Plasma interface for modeling DC discharges. The complicated coupling between the electron transport, heavy species transport, and electrostatic field is handled automatically by the software. Furthermore, the secondary emission flux from ion bombardment on an electrode is automatically computed and used in the boundary condition for electrons. This makes it easy to model things such as positive columns and DC glow discharges.

The physics of DC discharges can be further complicated when a strong, static magnetic field is present. This can lead to highly anisotropic transport properties for the electrons. The Plasma interface automatically computes the tensor form of the electron mobility if requested. This makes it possible to model things such as magnetron discharges and hall thrusters. DC discharges require reaction rates to be specified in terms of Townsend rather than rate coefficients. The main reason is that Townsend coefficients provide a better description of the physics in the cathode fall region. The discharge is sustained by secondary emission, rather than a time varying electric field. Therefore, secondary emission effects must be included in any DC discharge model otherwise the plasma simply self extinguishes.

It is important to be aware of the different time scales at which the components of a plasma can change:

TABLE 8-1: TABLE OF VARIOUS TIME SCALES IN A PLASMA

PHENOMENA	TIME SCALE
Electromagnetic field propagation	Instantaneous
Electron energy	< 1 ns
Electron transport	nanoseconds
Ion transport	microseconds

TABLE 8-1: TABLE OF VARIOUS TIME SCALES IN A PLASMA

PHENOMENA	TIME SCALE
Excited species transport	milliseconds
Neutral gas flow	10's milliseconds

DC Discharge Boundary Conditions

Since secondary emission effects are vital in sustaining DC discharges. Therefore the boundary conditions play an important role in DC discharges. The secondary emission flux for electrons is defined as:

$$\mathbf{n} \cdot \mathbf{\Gamma}_{e,s} = \sum_{k=1}^N \gamma_k (\mathbf{\Gamma}_k \cdot \mathbf{n})$$

Additionally, the secondary emission energy flux requires that the mean energy of the secondary electrons are specified:

$$\mathbf{n} \cdot \mathbf{\Gamma}_{\epsilon,s} = \sum_{k=1}^N \gamma_k \bar{\epsilon}_k (\mathbf{\Gamma}_k \cdot \mathbf{n})$$

The secondary emission fluxes are automatically computed by the Plasma interface when surface reactions are added to a model. The mean electron energy of the emitted electrons, ϵ_k is typically a function of the ionization energy of the impinging ion and the work function of the surface:

$$\bar{\epsilon}_k = \Delta\epsilon_k - 2W_f$$

where $\Delta\epsilon_k$ is the ionization energy (SI unit: V) and W_f is the work function of the surface (SI unit: V). On dielectric surfaces, charge accumulates due to the difference in fluxes between the electrons and ions:

$$\frac{d\rho_s}{dt} = \mathbf{n} \cdot \mathbf{J}_i + \mathbf{n} \cdot \mathbf{J}_e$$

where \mathbf{J}_i is the normal total ion current and \mathbf{J}_e is the normal total electron current. The surface charge density, ρ_s is used as a boundary condition in the Electrostatics interface:

$$-\mathbf{n} \cdot \mathbf{D} = \rho_s$$

Tips for Modeling DC Discharges

There are a number of key points to consider when modeling DC discharges:

- For atmospheric pressure DC discharges, the plasma might naturally try to arc. This typically causes the solver to terminate because it cannot reach the desired tolerances. Often this behavior is undesired, so in practical experimental setups a series RC circuit is added between the driven electrode and the power supply. This prevents arcing from occurring and results in a glow discharge. It is recommended to use the **Series RC circuit** option in the **Circuit Settings** section of the **Electric Potential** feature.
- Townsend coefficients should be used in preference to rate coefficients if possible. Townsend coefficients give a more realistic description of cathode fall regions in DC discharges.
- Secondary emission from a surface bombarded by ions must be included somewhere in the model in order for the discharge to be sustained.
- Although a DC discharge ultimately reaches a steady-state condition, the problem must be solved as a time-dependent problem so the plasma can naturally evolve into its equilibrium state.
- The initial conditions should enforce the electroneutrality constraint.

Theory for Inductively Coupled Plasmas

The Inductively Coupled Plasma Interface and The Microwave Plasma Interface theory is described in this section:

- [Introduction to Inductively Coupled Plasmas](#)
- [Domain Equations for the Inductively Coupled Plasma Interface](#)
- [Wave Heated Discharge Theory](#)
- [Plasma Conductivity in the Presence of a Static Magnetic Field](#)
- [Plasma Conductivity in the Presence of a Static Magnetic Field](#)

Introduction to Inductively Coupled Plasmas

Inductively coupled plasma (ICP) reactors are important in plasma processing and plasma sources because the plasma density can be considerably higher than in capacitively coupled discharges. ICP reactors are also attractive from the modeling perspective because they are relatively straightforward to model. Inductively coupled plasmas are relatively straightforward to model due to the fact that the induction currents can be solved for in the frequency domain. This means that the RF cycle applied to the driving coil does not need to be explicitly resolved. As such, the quasi steady-state solution is reached in relatively few time steps.

In a 2D axial symmetric ICP reactor, a coil driven at a high frequency (usually 13.56 MHz) creates a magnetic field with only an in-plane component (the r - z plane) and a high frequency out-of-plane component of the electric field (in the θ -direction). This results in oscillatory electron motion in the θ -direction only. Due to this effect, further simplifications can be made and the out-of-plane electric field can be solved for in the frequency domain. Consider the momentum conservation of electrons as given in [Equation 6-3](#):

$$\frac{\partial}{\partial t}(n_e m_e \mathbf{u}_e) + \nabla \cdot n_e m_e \mathbf{u}_e \mathbf{u}_e^T = -(\nabla \cdot \mathbf{p}_e) + q n_e \mathbf{E} - n_e m_e \mathbf{u}_e v_m$$

where m_e is the electron mass (SI unit: kg), \mathbf{u}_e is the drift velocity of the electrons (SI unit: m/s), \mathbf{p}_e is the electron pressure tensor (SI unit: Pa), q is the electron charge (SI unit: s A), \mathbf{E} is the electric field (SI unit: V/m) and v_m is the momentum transfer

frequency (SI unit: 1/s). Neglecting the inertial term and taking only the θ -component yields:

$$\frac{\partial}{\partial t}(n_e m_e u_{e,\theta}) = -\frac{\partial}{\partial \theta}(n_e k_B T_e) + q n_e E_\theta - n_e m_e u_{e,\theta} v_m.$$

Since n_e and T_e are uniform in the θ -direction, the first term on the right and side can be neglected, resulting in:

$$\frac{\partial}{\partial t}(n_e m_e u_{e,\theta}) = q n_e E_\theta - n_e m_e u_{e,\theta} v_m$$

which is a linear equation in for $u_{e,\theta}$ which can be Fourier transformed by letting:

$$E_\theta = \tilde{E}_\theta e^{j\omega t}$$

and

$$u_{e,\theta} = \tilde{u}_{e,\theta} e^{j\omega t}$$

the resulting equation in the frequency domain is:

$$j\omega n_e m_e \tilde{u}_{e,\theta} = q n_e \tilde{E}_\theta - n_e m_e \tilde{u}_{e,\theta} v_m.$$

Multiplying this equation by q and defining the out-of-plane electron current density as:

$$\tilde{J}_\theta = q n_e \tilde{u}_{e,\theta}$$

the following expression is obtained:

$$\tilde{J}_\theta = \frac{q^2 n_e}{m_e (v_m + j\omega)} \tilde{E}_\theta$$

which relates the current density to the electric field in the form of $J_\theta = \sigma E_\theta$ with the plasma conductivity, σ defined as:

$$\sigma = \frac{q^2 n_e}{m_e (v_m + j\omega)}$$

which is required by [Equation 8-3](#). The third term on the left hand side of [Equation 6-4](#) now has an additional contribution due to the out-of-plane motion of the electrons:

$$\mathbf{E} \cdot \boldsymbol{\Gamma}_e = E_\theta \cdot \frac{J_\theta}{q} = \frac{1}{q} \left(\frac{1}{2} \right) \text{real}(\tilde{E}_\theta \tilde{J}_\theta).$$

The presence of the driving coil in an ICP reactor also introduces capacitive coupling between the coil and the electrons in the plasma sheath. This is neglected in COMSOL Multiphysics. This is because the electron density in an ICP reactor is typically so high that the Debye length is very small, meaning that the in-plane component of the electric field cannot penetrate into the plasma.

When collisionless heating is included in the model an extra degree of freedom for the out-of-plane electron drift velocity, u_θ , is added:

$$n_e u_\theta j\omega t + n_e \nu_e u_\theta - \eta_{\text{eff}} \nabla \cdot (n_e \nabla u_\theta) = -\frac{q}{m_e} n_e E_\theta$$

where η_{eff} is the effective viscosity coefficient defined as:

$$\eta_{\text{eff}} = \left(\frac{c^2 V_{th}^4}{\pi^2 \omega_p^2 \omega} \right)^{\frac{1}{3}}$$

and V_{th} is the thermal velocity defined as:

$$V_{th} = \sqrt{\frac{3T_e q}{m_e}}$$

and ω_p is the plasma frequency defined as:

$$\omega_p = \sqrt{\frac{q^2 n_e}{\epsilon_0 m_e}}.$$

The plasma conductivity, σ , is set to zero and the out-of-plane current density, J_θ has the following term added to it:

$$J_\theta = n_e u_\theta q.$$

Domain Equations for the Inductively Coupled Plasma Interface



The **Inductively Coupled Plasma Interface** solves a system of coupled partial differential equations for:

- The electron density

- The mean electron energy
- The mass fraction of each of the heavy species
- The electrostatic potential
- The electric field due to induction currents

For a nonpolarized plasma, the induction currents are computed in the frequency domain using the following equation:

$$(j\omega\sigma - \omega^2 \epsilon_0 \epsilon_r) \mathbf{A} + \nabla \times (\mu_0^{-1} \nabla \times \mathbf{A}) = \mathbf{J}^e \tag{8-3}$$

	<ul style="list-style-type: none"> • Plasma Reactors Theory • The Magnetic Fields Interface in the <i>AC/DC User's Guide</i>
	<p>The links to items described in other user's guides or reference manuals do not work in the PDF, only from the on line help.</p>

Theory for Microwave Plasmas

Microwave plasmas, or wave-heated discharges, find applications in many industrial areas such as semiconductor processing, surface treatment, and the abatement of hazardous gases.

Introduction

Microwave plasmas are sustained when electrons can gain enough energy from an electromagnetic wave as it penetrates into the plasma. The physics of a microwave plasma is quite different depending on whether the TE mode (out-of-plane electric field) or the TM mode (in-plane electric field) is propagating. In both cases, it is not possible for the electromagnetic wave to penetrate into regions of the plasma where the electron density exceeds the critical electron density (around $7.6 \times 10^{16} \text{ 1/m}^3$ for 2.45 GHz). The critical electron density is given by the formula:

$$n_e = \frac{\epsilon_0 m_e \omega^2}{q^2}$$

where ϵ_0 is the permittivity of free space, m_e is the electron mass, ω is the angular frequency, and q is the electron charge. This corresponds to the point at which the angular frequency of the electromagnetic wave is equal to the plasma frequency. The pressure range for microwave plasmas is very broad. For electron cyclotron resonance (ECR) plasmas, the pressure can be on the order of around 1 Pa, while for non-ECR plasmas, the pressure typically ranges from 100 Pa up to atmospheric pressure. The power can range from a few watts to several kilowatts. Microwave plasmas are popular due to the cheap availability of microwave power.

Wave Heated Discharge Theory

In the Microwave Plasma interface, the electromagnetic waves are computed in the frequency domain and all other variables in the time domain. In order to justify this approach, we start from Maxwell's equations, which state that:

$$\nabla \times \tilde{\mathbf{E}} = -\frac{\partial \tilde{\mathbf{B}}}{\partial t} \quad (8-4)$$

$$\nabla \times \tilde{\mathbf{H}} = \tilde{\mathbf{J}}_p + \frac{\partial \tilde{\mathbf{D}}}{\partial t} \quad (8-5)$$

where $\tilde{\mathbf{E}}$ is the electric field (V/m), $\tilde{\mathbf{B}}$ is the magnetic flux density (T), $\tilde{\mathbf{H}}$ is the magnetic field (A/m), $\tilde{\mathbf{J}}_p$ is the plasma current density (A/m²), and $\tilde{\mathbf{D}}$ is the electrical displacement (C/m²). The tilde is used to denote that the field is varying in time with frequency $\omega/2\pi$. The plasma current density can be approximated by this expression:

$$\tilde{\mathbf{J}}_p = -qn_e \tilde{\mathbf{v}}_e \quad (8-6)$$

where q is the unit charge (C), n_e is the electron density (1/m³), and $\tilde{\mathbf{v}}_e$ is the mean electron velocity under the following two assumptions:

- 1 Ion motion is neglected with respect to the electron motion on the microwave time scale.
- 2 The electron density is assumed constant in space on the microwave time scale.

The mean electron velocity on the microwave time scale, $\tilde{\mathbf{v}}_e$, is obtained by assuming a Maxwellian distribution function and taking a first moment of the Boltzmann equation [Ref. 2](#):

$$\frac{\partial \tilde{\mathbf{v}}_e}{\partial t} = -\frac{q}{m_e} \tilde{\mathbf{E}} + \nu_m \tilde{\mathbf{v}}_e \quad (8-7)$$

where m_e is the electron mass (kg) and ν_m is the momentum transfer frequency between the electrons and background gas (1/s). As pointed out in [Ref. 1](#), the equations are linear, so we can take a Fourier transform of the equation system. Taking a Fourier transform of equation [Equation 8-7](#) gives:

$$j\omega \bar{\mathbf{v}}_e + \nu_m \bar{\mathbf{v}}_e = \frac{q}{m_e} \bar{\mathbf{E}}$$

where the tilde has been replaced by a bar to reflect the fact that we are now referring to the amplitude of the fields. Multiplying both sides by $-en_e$ and re-arranging gives:

$$-qn_e \bar{\mathbf{v}}_e = \frac{n_e q^2}{m_e(\nu_m + j\omega)} \bar{\mathbf{E}}$$

or, in simpler form:

$$\bar{\mathbf{J}}_p = \sigma \bar{\mathbf{E}}$$

where:

$$\sigma = \frac{n_e q^2}{m_e (\nu_m + j\omega)} \quad (8-8)$$

Equation 8-4 and Equation 8-5 can be re-arranged by taking the time derivative of Equation 8-5 and substituting in Equation 8-4:

$$\nabla \times \mu^{-1} \nabla \times \bar{\mathbf{E}} = (\omega^2 \epsilon_0 \epsilon_r - j\omega\sigma) \bar{\mathbf{E}}$$

where μ is the permeability, σ is given in equation Equation 8-8 above, and the plasma relative permittivity set to one. The equation could also be recast where the relative permittivity is complex-valued and the plasma conductivity is zero Ref. 3. The convention employed throughout the Plasma Module is that the plasma conductivity is given by equation Equation 8-8 and the plasma relative permittivity is set to 1.

Solving the above equation with appropriate boundary conditions allows for the power transferred from the electromagnetic fields to the electrons to be calculated:

$$Q_{rh} = \frac{1}{2} \text{real}(\bar{\mathbf{J}} \cdot \bar{\mathbf{E}}^\circ)$$

where \mathbf{J} is the total current density (the plasma current plus the displacement current density) and $^\circ$ denotes the complex conjugate.

AMBIPOLAR FIELDS

In addition to the equation above, a set of equations are solved in the time domain for the electron density n_e , electron energy density n_e , plasma potential V , and all ionic and neutral species. These are the same equations solved in the Plasma interface and are given in the Plasma Reactors Theory section.

TE AND TM MODE PROPAGATION

In 2D or 2D axisymmetric models, the electromagnetic waves propagate in either the transverse electric (TE) mode or the transverse magnetic (TM) mode. In the TE mode, the electric field is only in the transverse direction and the magnetic field in the direction of propagation. Therefore, COMSOL solves only for the out-of-plane component of the high-frequency electric field. In the TM mode, the magnetic field is in the transverse direction and the electric field only in the direction of propagation, so COMSOL solves only for the in-plane components of the high-frequency electric field.

TE MODE

In the TE mode, electrons do not experience any change in the high-frequency electric field during the microwave time scale. This means that the phase coherence between the electrons and electromagnetic waves is only destroyed through collisions with the background gas. The loss of phase coherence between the electrons and high-frequency fields is what results in energy gain for the electrons. Therefore, the momentum collision frequency is simply given by:

$$\nu_m = \nu_e$$

where ν_e is the collision frequency between the electrons and neutrals.

TM MODE

The TM mode causes in-plane motion of the electrons on the microwave time scale, so in regions where the high-frequency electric field is significant (the contour where the electron density is equal to the critical density), the time-averaged electric field experienced by the electrons may be non-zero. This destroys the phase coherence between the electrons and the fields, causing the electrons to gain energy. This is an example of a non-local kinetic effect, which is difficult to approximate with a fluid model. However, since this effect is similar to collisions with a background gas, the non-local effects can be approximated by adding an effective collision frequency to the momentum collision frequency:

$$\nu_m = \nu_e + \nu_{\text{eff}}$$

where ν_{eff} is the effective collision frequency to account for non-local effects. This is discussed in more detail in Ref 1, where an effective collision frequency of no more than $\omega/20$ is suggested.

ECR REACTORS

When modeling ECR (*electron cyclotron resonance*) reactors, another layer of complication is added to the problem. The electron transport properties become tensors and functions of a static magnetic flux density, which can be created using permanent magnets. The plasma conductivity also becomes a full tensor, and a highly nonlinear function of the static magnetic flux density. In addition, it is necessary to consider all three components of the electromagnetic field. Comprehensive details on how to set up and solve a model of an ECR reactor can be found in the Dipolar Microwave Plasma Source model documentation.

RESONANCE ZONE

The above equations are quite straightforward to solve, provided that the plasma frequency is below the angular frequency everywhere in the modeling domain. At a frequency of 2.45 GHz, this corresponds to an electron density of $7.6 \times 10^{16} \text{ 1/m}^3$, which is lower than most industrial applications. When the plasma density is equal to this value, the electromagnetic wave transitions from propagating waves to evanescent waves. Applications of microwave plasmas where the electron density is greater than the critical density include:

- Atmospheric pressure discharges, where the electron density can be several orders of magnitude higher than the critical density.
- Traveling-wave-sustained and surface-wave discharges. For the surface wave to propagate, the electron density must be higher than the critical density.

The resonance zone can be smoothed by activating the **Compute tensor plasma conductivity** check box in the **Plasma Conductivity Coupling** multiphysics coupling:

The screenshot shows the 'Plasma Conductivity Coupling' settings. It includes a dropdown for 'Equation' set to 'Study 1, Frequency-Transient'. The formula for plasma conductivity is displayed: $\sigma_{\text{plasma}} = \frac{q^2 n_e}{m_e(\nu_e + j\omega)} f(\mathbf{B})$. Below this, the 'Compute Tensor Plasma Conductivity' section has a checked box for 'Compute tensor plasma conductivity' and a 'Doppler broadening parameter' δ set to 20. The 'Coupled Interfaces' section shows 'Source' as 'Plasma (plas)' and 'Destination' as 'Magnetic Fields (mf)'.

The **Doppler broadening parameter**, δ , corresponds to the value used for the effective collision frequency via the formula:

$$\nu_m = \nu_e + \frac{\omega}{\delta}$$

Therefore, a value of 20 is a compromise between accuracy and numerical stability as detailed above.

DEPOSITED AND REFLECTED POWER

When using the Port boundary condition, the sum of the deposited and reflected power is supplied by default. It is also possible to specify only the deposited power, as shown in the **Settings** window below:

The screenshot shows the 'Settings' window for a 'Port' boundary condition. The 'Label' is 'Port 1'. Under 'Boundary Selection', the 'Selection' is 'Manual'. There is a '9' in a box and an 'Active' button. Below this are sections for 'Override and Contribution', 'Equation', and 'Port Properties'. In 'Port Properties', 'Port name' is '1', 'Type of port' is 'Rectangular', and 'Wave excitation at this port' is 'On'. The 'Specify deposited power' checkbox is checked. 'Deposited power' is set to P_{dep} 100[W] W. 'Port phase' is θ_{in} 0 rad. There is an option to 'Activate slit condition on interior port'. Under 'Port Mode Settings', 'Mode type' is 'Transverse magnetic (TM)' and 'Mode number' is '1'.

Using this option results in a more stable equation system because the total power transferred to the electrons remains constant. When you use the **Port input power** option, some of the power is deposited and some is reflected back out of the port, depending on the plasma's current state. The plasma can go from absorbing a very small amount to a very large amount of power in a very short time period, which can make the problem numerically unstable or lead to the solver taking extremely small time steps.

MODELING SUGGESTIONS

The following is a collection of tips and tricks to try to help with convergence and decrease computation time:

- Fix the total power into the discharge by using the Specify deposited power option in the Port boundary condition, or by using the approach suggested in [Ref. 1](#), which the Applications Libraries example “Dipolar Microwave Plasma Source” demonstrates.
- Start the Doppler broadening parameter to a number below 1, then ramp this number up to 20 over the course of the simulation. This smears out the resonance zone initially, then gradually make the region smaller and smaller. You can implement it by defining a Ramp function with a cutoff value of 20.
- If the number density in the discharge is very high, it is probably operating in full surface-wave mode. In this case, it may be necessary to use a Doppler broadening parameter of 10. This makes the model more stable when solving, but the accuracy of such a model should be carefully considered.
- The initial electron density should be below the critical plasma density in the TM mode case. This recommendation is not strictly necessary for the TE mode case.
- Negatively biasing an electrode may have a significant effect on the discharge because it may shift the contour of critical plasma density away from its unbiased location. The best thing to do is to solve the model with a value of around 5 for the Doppler broadening parameter, then slowly increase its value.

SOLVER SETTINGS

Solver settings play an important role and COMSOL automatically generates the best solver settings depending on how the model is set up. By default, when the **Port input power** option is used, the solver settings mentioned below are implemented. The segregated solver is used with two groups:

- All the plasma variables (electron density, electron energy, ion density, plasma potential, and so on).
- All the variables associated with the electromagnetic waves (high frequency electric field, S-parameters).

When the **Specify deposited power** option in the **Port** boundary condition is used, the solver suggestion is modified so that there are three groups:

- All the plasma variables (electron density, electron energy, ion density, plasma potential, etc.).

- All the variables associated with the electromagnetic waves (high-frequency electric field, S-parameters).
- A dependent variable called **Pdeposited**, which is a differential algebraic equation used to fix the deposited, rather than total power.



In COMSOL Multiphysics the absorption of electromagnetic waves must be realized through a complex-valued electrical conductivity. It is not possible to specify a complex-valued permittivity. The Plasma Conductivity Coupling multiphysics coupling automatically computes the plasma conductivity based on the electron density, collision frequency, and any external DC magnetic flux density that is present.

Plasma Conductivity in the Presence of a Static Magnetic Field

The relationship between the plasma current density and the electric field becomes more complicated in the presence of a DC magnetic field. The following equation defines this relationship:

$$\boldsymbol{\sigma}^{-1} \cdot \mathbf{J} = \mathbf{E}$$

where $\boldsymbol{\sigma}$ is the plasma conductivity tensor which is a function of the electron density, collision frequency and magnetic flux density. Using the definitions:

$$\alpha = \frac{q}{m_e(\nu_e + j\omega)}, \beta = n_e q \alpha$$

where q is the electron charge, m_e is the electron mass, n_e is the collision frequency, and ω is the angular frequency of the electromagnetic field, the inverse of the plasma conductivity can be written as:

$$\boldsymbol{\sigma}^{-1} = \begin{bmatrix} \frac{1}{\beta} & -\frac{\alpha}{\beta} B_z & \frac{\alpha}{\beta} B_y \\ \frac{\alpha}{\beta} B_z & \frac{1}{\beta} & -\frac{\alpha}{\beta} B_x \\ -\frac{\alpha}{\beta} B_y & \frac{\alpha}{\beta} B_x & \frac{1}{\beta} \end{bmatrix}$$

where \mathbf{B} is the magnetic flux density which has three components B_x , B_y , and B_z . The inverse of the plasma conductivity is used because it can be written in a compact, convenient form. The plasma conductivity is a highly nonlinear function of the

magnetic flux density. The conductivity exhibits resonant behavior at a critical magnetic flux density given by the simple formula:


$$B_{\text{res}} = \frac{m_e \omega}{q}$$

At this critical magnetic flux density the electrons continually gain energy from the electric field at all phases of the RF cycle. This principle, called electron cyclotron resonance is exploited in ECR reactors.

References

1. G.J.M. Hagelaar, K. Makasheva, L. Garrigues, and J.-P. Boeuf, “Modelling of a dipolar microwave plasma sustained by electron cyclotron resonance,” J. Phys. D: Appl. Phys., vol. 42, p. 194019 (12pp), 2009.
2. R.L. Kinder and M.J. Kushner, “Consequences of mode structure on plasma properties in electron cyclotron resonance sources,” J. Vac. Sci. Technol. A, vol. 17, Sep/Oct 1999.
3. Michael A. Lieberman and Allan J. Lichtenberg, “Principles of Plasma Discharges and Materials Processing”, Wiley (2005).


Equilibrium Discharges Interfaces


This chapter describes the physics interfaces found under the **Plasma>Equilibrium Discharges** branch (). The Equilibrium Discharges interfaces can be used to model plasma in or in close local thermodynamic equilibrium (LTE), that is, where the electrons and heavy particles temperature are approximately equal. Close to LTE conditions are usually observed at high pressure and/or under small values of electrical fields.

In this chapter:

- [The Equilibrium DC Discharge Interface](#)
- [The Combined Inductive/DC Discharge Interface](#)
- [The Equilibrium Inductively Coupled Plasma Interface](#)
- [Equilibrium Discharges Theory](#)
- [Summary of the Equilibrium Discharge Interfaces Features](#)

The Equilibrium DC Discharge Interface

The **Equilibrium DC Discharge** () interface is used to study equilibrium discharges that are sustained by a static (or slowly varying) electric field where induction currents and fluid flow effects are negligible. This multiphysics interface adds an **Electric Currents** interface and a **Heat Transfer in Fluids** interface. The multiphysics couplings add special boundary conditions to model the ion and electron heating at the plasma boundaries as well as heating and cooling of the equilibrium plasma by enthalpy transport, Joule heating and radiation loss.

When a predefined **Equilibrium DC Discharge** interface is added from the **Plasma>Equilibrium Discharges** branch () of the **Model Wizard** or **Add Physics** windows, **Electric Currents** and **Heat Transfer in Fluids** interfaces are added to the Model Builder. In addition, the **Multiphysics** node is added, which automatically includes the multiphysics coupling features **Equilibrium Discharge Heat Source**, one **Temperature Coupling** node, and a **Equilibrium Discharge Boundary Heat Source** feature.

On the Constituent Physics Interfaces

The **Electric Currents** interface computes electric field, current, and potential distributions in conducting media under conditions where inductive effects are negligible; that is, when the skin depth is much larger than the studied device. The **Electric Currents** interface solves a current conservation equation based on Ohm's law using the scalar electric potential as the dependent variable.

The **Heat Transfer in Fluids** interface provides features for modeling heat transfer by conduction, convection, and radiation. A **Heat Transfer in fluids** model is active by default on all domains. All functionality for including other domain types, such as a solid domain, is also available.



For details about predefined multiphysics couplings, see [Multiphysics Modeling Approaches](#) in the *COMSOL Multiphysics Reference Manual*.

SETTINGS FOR PHYSICS INTERFACES AND COUPLING FEATURES

When physics interfaces are added using the predefined couplings, for example **Equilibrium DC Discharge**, specific settings are included with the physics interfaces and the coupling features.

However, if physics interfaces are added one at a time, followed by the coupling features, these modified settings are not automatically included.

For example, if single **Electric Currents** and **Heat Transfer in Fluids** interfaces are added, COMSOL adds an empty **Multiphysics** node. You can then choose from the available coupling features, **Equilibrium Discharge Heat Source**, **Temperature Coupling**, and **Equilibrium Discharge Boundary Heat Source** but the modified settings are not included.


	Coupling features are available from the context menu (right-click the Multiphysics node) or from the Physics toolbar, Multiphysics menu.
---	--

TABLE 9-1: MODIFIED SETTINGS FOR AN EQUILIBRIUM DC DISCHARGE INTERFACE

PHYSICS INTERFACE OR COUPLING FEATURE	MODIFIED SETTINGS
Electric Currents	No changes
Heat Transfer in Fluids	No changes
Equilibrium Discharge Heat Source	The Domain Selection is the same as that of the participating physics interfaces. The corresponding Electric Currents and Heat Transfer in Fluids interfaces are preselected in the Equilibrium Discharge Heat Source section.
Temperature Coupling	The corresponding Electric Currents and Heat Transfer in Fluids interfaces are preselected in the Temperature Coupling section (described in the <i>COMSOL Multiphysics Reference Manual</i>).
Equilibrium Discharge Boundary Heat Source	The Boundary Selection contains all boundaries of the participating physics interfaces. The corresponding Electric Currents and Heat Transfer in Fluids interfaces are preselected in the Equilibrium Discharge Boundary Heat Source section.

PHYSICS INTERFACE AND COUPLING FEATURES



Use the online help in COMSOL Multiphysics to locate and search all the documentation. All these links also work directly in COMSOL Multiphysics when using the Help system.

Coupling Features

- The [Equilibrium Discharge Heat Source](#) and [Equilibrium Discharge Boundary Heat Source](#) multiphysics coupling nodes are described in this section.
- The [Boundary Electromagnetic Heat Source](#) and [Temperature Coupling](#) feature nodes are described for [The Joule Heating Interface](#) in the *COMSOL Multiphysics Reference Manual*.

Physics Interface Features

Physics feature nodes are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).




In general, to add a node, go to the **Physics** toolbar, no matter what operating system you are using.

- The available physics features for [The Electric Currents Interface](#) are listed in the section [Domain, Boundary, Edge, Point, and Pair Nodes for the Electric Currents Interface](#).
- The available physics features for the heat transfer interfaces are listed in the sections [Domain Features](#), [Boundary Features](#), and [Edge and Point Features](#) in the *COMSOL Multiphysics Reference Manual*.



If you also have the AC/DC Module, see *ICP Torch*: Application Library path **Plasma_Module/Equilibrium_Discharges/icp_torch**

Equilibrium Discharge Heat Source

The **Equilibrium Discharge Heat Source** multiphysics coupling () represents the source term Q (SI units: W/m^3) in the heat equation implemented by

$$\rho C_p \left(\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T \right) - \nabla \cdot (k \nabla T) = Q \quad (9-1)$$

The source term Q includes three selectable source/sink components:

- Resistive heating (ohmic heating).

$$Q = \mathbf{J} \cdot \mathbf{E} \quad (9-2)$$

- Volumetric net radiation loss Q_{rad} defined by the total volumetric emission coefficient, which comes from a material property.
- Enthalpy transport (energy carried by the electric current)

$$\frac{\partial}{\partial T} \left(\frac{5k_B T}{2q} \right) (\nabla T \cdot \mathbf{J}) \quad (9-3)$$

Note that the enthalpy transport term prevails, for instance, in the boundary layers close to electrodes in a fully ionized electric discharge, [Ref. 1](#).

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern <name>.<variable_name>. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the **name** string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is phs1.

HEAT SOURCE COMPONENTS

By default, the **Include enthalpy transport**, **Include Joule heating**, and **Include volumetric net radiation loss** check boxes are selected. From the list, select a **Total volumetric emission coefficient** Q_{rad} (SI unit: W/m³)—**From material** (the default) or **User defined**. For **User defined**, enter a value or expression in the field. The heat source component are selectable in order to be able to see the effect of each of the component on the plasma dynamics.

EQUILIBRIUM DISCHARGE HEAT SOURCE

This section defines the physics involved in the Equilibrium Discharge Heat Source multiphysics coupling. By default, the applicable physics interface is selected in the


Electromagnetic list to apply the **Heat transfer** to its physics interface to establish the coupling.

You can also select **None** from either list to uncouple the **Equilibrium Discharge Heat Source** node from a physics interface. If the physics interface is removed from the **Model Builder**, for example **Heat Transfer in Fluids** is deleted, then the **Heat transfer** list defaults to **None** as there is nothing to couple to.



If a physics interface is deleted and then added to the model again, and in order to re-establish the coupling, you need to choose the physics interface again from the lists. This is applicable to all multiphysics coupling nodes that would normally default to the once present physics interface. See [Multiphysics Modeling Approaches](#) in the *COMSOL Multiphysics Reference Manual*.

Flow Coupling

The **Flow Coupling** multiphysics coupling () defines **u** and **p** variables in order to set the model inputs in the physics interface. In addition it provides all the fluids quantities that may be needed by the physics interface (for example, viscosity, turbulence parameters, and so on).

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the **name** string must be unique. Only letters, numbers, and underscores (`_`) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is `fc1`.

FLOW COUPLING

This section defines the physics involved in the Flow Coupling multiphysics coupling. By default, the applicable physics interface is selected in the **Source** list to apply the **Destination** to its physics interface to establish the coupling.


You can also select **None** from either list to uncouple the **Flow Coupling** node from a physics interface. If the physics interface is removed from the **Model Builder**, for

example **Heat Transfer in Fluids** is deleted, then the **Destination** list defaults to **None** as there is nothing to couple to.



If a physics interface is deleted and then added to the model again, and in order to re-establish the coupling, you need to choose the physics interface again from the lists. This is applicable to all multiphysics coupling nodes that would normally default to the once present physics interface. See [Multiphysics Modeling Approaches](#) in the *COMSOL Multiphysics Reference Manual*.

Equilibrium Discharge Boundary Heat Source

The **Equilibrium Discharge Boundary Heat Source** multiphysics coupling () maps the electromagnetic surface losses as a heat source on the boundary (SI unit: W/m^2) in the heat transfer part of the model.

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (`_`) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is `bphs1`.

ELECTRODE PROPERTIES

Select an **Electrode polarity**—**Anode** (the default) or **Cathode**. Enter a **Surface work function** Φ (SI unit: V). The default is 4.15 V.

ELECTRON CURRENT

This section is available when **Cathode** is selected as the **Electrode polarity**.

Enter an **Effective Richardson's constant** A_R^* (SI unit: $\text{A}/(\text{m}^2 \cdot \text{K}^2)$). The default is $120 \text{ A}/(\text{m}^2 \cdot \text{K}^2)$. Enter an **Effective work function** Φ_{eff} (SI unit: V). The default is 2.6 V.

ION CURRENT

This section is available when **Cathode** is selected as the **Electrode polarity**. Enter a value or expression for the **Plasma's ionization potential** V_{ion} (SI unit: V). The default is 15.7 V.

SOURCE POSITION

To display this section, click the **Show** button () and select **Advanced Physics Options**.

Select a **Source position**—**Layer** (the default), **Upside**, or **Downside**.


EQUILIBRIUM DISCHARGE BOUNDARY HEAT SOURCE


This section has the same setting information as defined under [Equilibrium Discharge Heat Source](#).

The Combined Inductive/DC Discharge Interface



The 3D version of this physics interface requires an AC/DC Module license. For theory and physics interface feature descriptions, see the *AC/DC Module User's Guide*.

The **Combined Inductive/DC Discharge** () interface is used to study equilibrium discharges that are sustained by induction currents and/or static (slowly varying) electric field, for instance, in arc welding simulations. This multiphysics interface adds these single physics interfaces: Electric Currents, Magnetic Fields, Heat Transfer in Fluids, and Laminar Flow. The multiphysics couplings add special boundary conditions to model the ion and electron heating at the plasma boundaries as well as heating and cooling of the equilibrium plasma by enthalpy transport, Joule heating and radiation loss. The multiphysics couplings also combine the induction and electrostatic currents as well as the Lorentz forces to the hydrodynamics model.

When a predefined **Combined Inductive/DC Discharge** interface is added from the **Plasma>Equilibrium Discharges** branch () of the **Model Wizard** or **Add Physics** windows, **Electric Currents**, **Magnetic Fields**, **Heat Transfer in Fluids**, and **Laminar Flow** interfaces are added to the Model Builder. In addition, the **Multiphysics** node is added, which automatically includes the multiphysics coupling features **Equilibrium Discharge Heat Source**, **Static Current Density Component**, **Induction Current Density Component**, **Lorentz Force**, **Flow Coupling**, three **Temperature Coupling** nodes, and a **Equilibrium Discharge Boundary Heat Source** feature.


On the Constituent Physics Interfaces

The Electric Currents interface computes electric field, current, and potential distributions in conducting media under conditions where inductive effects are negligible; that is, when the skin depth is much larger than the studied device. The Electric Currents interface solves a current conservation equation based on Ohm's law using the scalar electric potential as the dependent variable.

The Magnetic Fields interface is used to compute magnetic field and induced current distributions in and around coils, conductors, and magnets. The Magnetic Fields interface solves Maxwell's equations formulated using the magnetic vector potential and, optionally for coils, the scalar electric potential as the dependent variables.

The Heat Transfer in Fluids interface provides features for modeling heat transfer by conduction, convection, and radiation. A Heat Transfer in fluids model is active by default on all domains. All functionality for including other domain types, such as a solid domain, is also available.

The Laminar Flow interface solves for conservation of energy, mass and momentum in fluids.




For details about predefined multiphysics couplings, see [Multiphysics Modeling Approaches](#) in the *COMSOL Multiphysics Reference Manual*.

SETTINGS FOR PHYSICS INTERFACES AND COUPLING FEATURES

When physics interfaces are added using the predefined couplings, for example **Combined Inductive/DC Discharge**, specific settings are included with the physics interfaces and the coupling features.

However, if physics interfaces are added one at a time, followed by the coupling features, these modified settings are not automatically included.

For example, if single **Electric Currents**, **Magnetic Fields**, **Heat Transfer in Fluids**, and **Laminar Flow** interfaces are added, COMSOL adds an empty **Multiphysics** node. You can then choose from the available coupling features, **Equilibrium Discharge Heat Source**, **Static Current Density Component**, **Induction Current Density Component**, **Lorentz Force**, **Flow Coupling**, **Temperature Coupling**, and **Equilibrium Discharge Boundary Heat Source** but the modified settings are not included.



Coupling features are available from the context menu (right-click the **Multiphysics** node) or from the **Physics** toolbar, **Multiphysics** menu.

TABLE 9-2: MODIFIED SETTINGS FOR A COMBINED INDUCTIVE/DC DISCHARGE INTERFACE

PHYSICS INTERFACE OR COUPLING FEATURE	MODIFIED SETTINGS
Electric Currents	No changes
Magnetic Fields	No changes
Heat Transfer in Fluids	No changes
Laminar Flow	No changes

TABLE 9-2: MODIFIED SETTINGS FOR A COMBINED INDUCTIVE/DC DISCHARGE INTERFACE

PHYSICS INTERFACE OR COUPLING FEATURE	MODIFIED SETTINGS
Equilibrium Discharge Heat Source	The Domain Selection is the same as that of the participating physics interfaces. The corresponding Electric Currents and Heat Transfer in Fluids interfaces are preselected in the Equilibrium Discharge Heat Source section.
Static Current Density Component	The Domain Selection is the same as that of the participating physics interfaces. The corresponding Electric Currents and Magnetic Fields interfaces are preselected in the Static Current Density Component section.
Induction Current Density Component	The Domain Selection is the same as that of the participating physics interfaces. The corresponding Electric Currents and Magnetic Fields interfaces are preselected in the Induction Current Density Component section.
Lorentz Force	The Domain Selection is the same as that of the participating physics interfaces. The corresponding Magnetic Fields and Laminar Flow interfaces are preselected in the Lorentz Force section.
Flow Coupling	The corresponding Laminar Flow and Heat Transfer in Fluids interfaces are preselected in the Flow Coupling section (described in the <i>COMSOL Multiphysics Reference Manual</i>).
Temperature Coupling	Three Temperature Coupling nodes (described in the <i>COMSOL Multiphysics Reference Manual</i>) are generated. The corresponding Electric Currents and Heat Transfer in Fluids interfaces, the Magnetic Fields and Heat Transfer in Fluids interfaces and the Laminar Flow and Heat transfer in Fluids are preselected for these nodes.
Equilibrium Discharge Boundary Heat Source	The Boundary Selection contains all boundaries of the participating physics interfaces. The corresponding Electric Currents and Heat Transfer in Fluids interfaces are preselected in the Equilibrium Discharge Boundary Heat Source section.

PHYSICS INTERFACE AND COUPLING FEATURES



Use the online help in COMSOL Multiphysics to locate and search all the documentation. All these links also work directly in COMSOL Multiphysics when using the Help system.

Coupling Features

- [Static Current Density Component](#), [Induction Current Density Component](#), and [Lorentz Force](#) are described in this section.
- The [Equilibrium Discharge Heat Source](#), [Flow Coupling](#), and [Equilibrium Discharge Boundary Heat Source](#) multiphysics coupling nodes are described for [The Equilibrium DC Discharge Interface](#).
- The [Boundary Electromagnetic Heat Source](#) and [Temperature Coupling](#) feature nodes are described for [The Joule Heating Interface](#) in the *COMSOL Multiphysics Reference Manual*.

Physics Interface Features


Physics nodes are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).



In general, to add a node, go to the **Physics** toolbar, no matter what operating system you are using.

- The available physics features for [The Laminar Flow Interface](#) are listed in the section [Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow](#).
- The available physics features for [The Electric Currents Interface](#) are listed in the section [Domain, Boundary, Edge, Point, and Pair Nodes for the Electric Currents Interface](#).
- The available physics features for the heat transfer interfaces are listed in the sections [Domain Features](#), [Boundary Features](#), and [Edge and Point Features](#) in the *COMSOL Multiphysics Reference Manual*.
- The available physics features for [The Magnetic Fields Interface](#) are listed in the section [Domain, Boundary, Point, and Pair Nodes for the Magnetic Fields Interface](#) in the *AC/DC Module User's Guide*.

Static Current Density Component

The **Static Current Density Component** multiphysics coupling () takes the induced current density defined in the Electric Currents interface

$$\mathbf{J} = \sigma \mathbf{E}$$

and maps it to the external current density defined in the Magnetic Fields interface.

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern <name>.<variable_name>. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the name string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is `sdc1`.

STATIC CURRENT DENSITY COMPONENT


This section defines the physics involved in the Static Current Density Component multiphysics coupling. By default, the applicable physics interface is selected in the **From electric currents** list and is applied to the **To magnetic fields** interface to establish the coupling.

You can also select **None** from either list to uncouple the Static Current Density Component node from a physics interface. If the physics interface is removed from the **Model Builder**, for example **Electric Currents** is deleted, then the **From electric currents** list defaults to **None** as there is nothing to couple to.



If a physics interface is deleted and then added to the model again, and in order to re-establish the coupling, you need to choose the physics interface again from the lists. This is applicable to all multiphysics coupling nodes that would normally default to the once present physics interface. See [Multiphysics Modeling Approaches](#) in the *COMSOL Multiphysics Reference Manual*.

Induction Current Density Component

The **Induction Current Density Component** multiphysics coupling () takes the induced current density defined in the Magnetic Fields interface

$$\mathbf{J} = -\sigma \frac{\partial \mathbf{A}}{\partial t} \quad (9-4)$$

and maps it to the external current density defined in the Electric Currents interface.

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern <name>.<variable_name>. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the **name** string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is `icdc1`.

INDUCTION CURRENT DENSITY COMPONENT


This section defines the physics involved in the Induction Current Density Component multiphysics coupling. By default, the applicable physics interface is selected in the **From magnetic fields** list and is applied to the **To electric currents** interface to establish the coupling.

You can also select **None** from either list to uncouple the Induction Current Density Component node from a physics interface. If the physics interface is removed from the **Model Builder**, for example **Magnetic Fields** is deleted, then the **From magnetic fields** list defaults to **None** as there is nothing to couple to.



If a physics interface is deleted and then added to the model again, and in order to re-establish the coupling, you need to choose the physics interface again from the lists. This is applicable to all multiphysics coupling nodes that would normally default to the once present physics interface. See [Multiphysics Modeling Approaches](#) in the *COMSOL Multiphysics Reference Manual*.

Lorentz Force

The **Lorentz Force** multiphysics coupling () maps the Lorentz Force contribution defined in the Magnetic Fields interface to the Volume force defined in the Laminar Flow interface.

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern <name>.<variable_name>. In

order to distinguish between variables belonging to different coupling nodes or physics interfaces, the **name** string must be unique. Only letters, numbers, and underscores (**_**) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is **1f1**.

LORENTZ FORCE

This section defines the physics involved in the Lorentz Force multiphysics coupling. By default, the applicable physics interface is selected in the **Electromagnetic** list to apply the **Fluid flow** to its physics interface to establish the coupling.

You can also select **None** from either list to uncouple the Lorentz Force node from a physics interface. If the physics interface is removed from the **Model Builder**, for example **Magnetic Fields** is deleted, then the **Electromagnetic** list defaults to **None** as there is nothing to couple to.





If a physics interface is deleted and then added to the model again, and in order to re-establish the coupling, you need to choose the physics interface again from the lists. This is applicable to all multiphysics coupling nodes that would normally default to the once present physics interface. See [Multiphysics Modeling Approaches](#) in the *COMSOL Multiphysics Reference Manual*.

The Equilibrium Inductively Coupled Plasma Interface



The 3D version of this physics interface requires an AC/DC Module license. For theory and physics interface feature descriptions, see the *AC/DC Module User's Guide*.

The **Equilibrium Inductively Coupled Plasma** () interface is used to study equilibrium discharges that are sustained by induction currents, for example in inductively coupled plasma torches. This multiphysics interface adds these single physics interfaces: Magnetic Fields, Heat Transfer in Fluids, and Laminar Flow. The multiphysics couplings add special boundary conditions to model the ion and electron heating at the plasma boundaries as well as heating and cooling of the equilibrium plasma by enthalpy transport, Joule heating and radiation loss. The multiphysics couplings also add the Lorentz forces to the hydrodynamics model.

When a predefined **Equilibrium Inductively Coupled Plasma** interface is added from the **Plasma>Equilibrium Discharges** branch () of the **Model Wizard** or **Add Physics** windows, **Magnetic Fields**, **Heat Transfer in Fluids**, and **Laminar Flow** interfaces are added to the Model Builder. In addition, the **Multiphysics** node is added, which automatically includes the multiphysics coupling features **Equilibrium Discharge Heat Source**, **Lorentz Force**, **Flow Coupling**, and two **Temperature Coupling** nodes.

On the Constituent Physics Interfaces

The Magnetic Fields interface is used to compute magnetic field and induced current distributions in and around coils, conductors, and magnets. The Magnetic Fields interface solves Maxwell's equations formulated using the magnetic vector potential and, optionally for coils, the scalar electric potential as the dependent variables.

The Heat Transfer in Fluids interface provides features for modeling heat transfer by conduction, convection, and radiation. A Heat Transfer in fluids model is active by default on all domains. All functionality for including other domain types, such as a solid domain, is also available. The temperature equation defined in solid domains corresponds to the differential form of the Fourier's law that may contain additional contributions like heat sources.

The Laminar Flow interface solves for conservation of energy, mass and momentum in fluids.



For details about predefined multiphysics couplings, see [Multiphysics Modeling Approaches](#) in the *COMSOL Multiphysics Reference Manual*.

SETTINGS FOR PHYSICS INTERFACES AND COUPLING FEATURES

When physics interfaces are added using the predefined couplings, for example **Equilibrium Inductively Coupled Plasma**, specific settings are included with the physics interfaces and the coupling features.

However, if physics interfaces are added one at a time, followed by the coupling features, these modified settings are not automatically included.

For example, if single **Magnetic Fields**, **Heat Transfer in Fluids**, and **Laminar Flow** interfaces are added, COMSOL adds an empty **Multiphysics** node. You can then choose from the available coupling features, **Equilibrium Discharge Heat Source**, **Lorentz Force**, **Flow Coupling**, and **Temperature Coupling** but the modified settings are not included.



Coupling features are available from the context menu (right-click the **Multiphysics** node) or from the **Physics** toolbar, **Multiphysics** menu.

TABLE 9-3: MODIFIED SETTINGS FOR AN EQUILIBRIUM INDUCTIVELY COUPLED PLASMA INTERFACE

PHYSICS INTERFACE OR COUPLING FEATURE	MODIFIED SETTINGS
Magnetic Fields	No changes
Heat Transfer in Fluids	No changes
Laminar Flow	No changes
Equilibrium Discharge Heat Source	The Domain Selection is the same as that of the participating physics interfaces. The corresponding Electric Currents and Heat Transfer in Fluids interfaces are preselected in the Equilibrium Discharge Heat Source section.
Lorentz Force	The Domain Selection is the same as that of the participating physics interfaces. The corresponding Magnetic Fields and Laminar Flow interfaces are preselected in the Lorentz Force section.

TABLE 9-3: MODIFIED SETTINGS FOR AN EQUILIBRIUM INDUCTIVELY COUPLED PLASMA INTERFACE

PHYSICS INTERFACE OR COUPLING FEATURE	MODIFIED SETTINGS
Flow Coupling	The corresponding Laminar Flow and Heat Transfer in Fluids interfaces are preselected in the Flow Coupling section (described in the <i>COMSOL Multiphysics Reference Manual</i>).
Temperature Coupling	Two Temperature Coupling nodes (described in the <i>COMSOL Multiphysics Reference Manual</i>) are generated. The corresponding Magnetic Fields and Heat Transfer in Fluids interfaces and the Laminar Flow and Heat transfer in Fluids are preselected for these nodes.

PHYSICS INTERFACE AND COUPLING FEATURES



Use the online help in COMSOL Multiphysics to locate and search all the documentation. All these links also work directly in COMSOL Multiphysics when using the Help system.

Coupling Features

- The [Equilibrium Discharge Heat Source](#) and [Flow Coupling](#) multiphysics coupling nodes are described for [The Equilibrium DC Discharge Interface](#).
- The [Lorentz Force](#) multiphysics coupling node is described for [The Combined Inductive/DC Discharge Interface](#).
- The [Boundary Electromagnetic Heat Source](#) and [Temperature Coupling](#) feature nodes are described for [The Joule Heating Interface](#) in the *COMSOL Multiphysics Reference Manual*.

Physics Interface Features

Physics nodes are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).



In general, to add a node, go to the **Physics** toolbar, no matter what operating system you are using.

- The available physics features for [The Laminar Flow Interface](#) are listed in the section [Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow](#).

- The available physics features for the heat transfer interfaces are listed in the sections [Domain Features](#), [Boundary Features](#), and [Edge and Point Features](#) in the *COMSOL Multiphysics Reference Manual*.
- The available physics features for [The Magnetic Fields Interface](#) are listed in the section [Domain, Boundary, Point, and Pair Nodes for the Magnetic Fields Interface](#) in the *AC/DC Module User's Guide*.

Equilibrium Discharges Theory



This theory applies to the Equilibrium DC Discharge, Equilibrium Inductively Coupled Plasma, and Combined Inductive/DC Discharge interfaces.

Equilibrium discharges (thermal plasmas) have nowadays a large range of industrial applications including: cutting, welding, spraying, waste destruction, and surface treatment ([Ref. 1](#)). The current methods used for producing thermal plasmas generally employ high-intensity arcs, inductively coupled high-frequency discharges, or microwave discharges. The purpose of the Equilibrium Discharge interface is to help users model thermal plasmas generated by the first two methods (that is, arcs or inductively coupled discharges). Accordingly, the targeted applications are primarily DC and inductively coupled plasma torches as well as arc welding devices and circuit breakers.

Thermal plasmas are assumed to be under partial to complete local thermodynamic equilibrium (LTE) conditions ([Ref. 1](#)). At a macroscopic level, these kind of plasma can be considered as conductive fluid mixtures, which leads to the magnetohydrodynamics (MHD) equations. The latter combine the Navier-Stokes, heat and Maxwell equations to describe the motion of the conducting fluid in an electromagnetic field.

The Equilibrium Discharge interfaces are based on a set of assumptions that lead to simplifications of the MHD equations:

- The plasma is fully ionized (two-component mixture, that is, electrons and ions).
- The plasma is under local thermodynamic equilibrium (LTE) conditions (the electron temperature T_e is approximately equal to the heavy particle temperature, T_h , under low electric field at atmospheric pressure).
- The plasma is considered to be a locally neutral Newtonian fluid mixture.
- The plasma flow is laminar and quasi-incompressible (low Mach number).
- Viscous dissipation and pressure work in the energy equation are neglected.
- The displacement current is negligible (quasi-static approximation, that is, low frequencies).
- Magnetic diffusion dominates induction by fluid motion ($R_{em} \ll 1$).
- The plasma is optically thin.

The magnetic Reynolds number (R_{em}) gives the ratio of induction by fluid motion over the magnetic diffusion

$$R_{em} = \frac{|\nabla \times (\mathbf{u} \times \mathbf{B})|}{|\eta \nabla^2 \mathbf{B}|} \cong \frac{u_0 l_0}{\eta} \quad (9-5)$$

Where the magnetic diffusivity coefficient

$$\eta = (\mu_0 \sigma)^{-1}$$

is the inverse of the product of the plasma conductivity (σ) and vacuum permeability (μ_0). The values u_0 and l_0 are respectively the velocity and length scale of the system under study.

DOMAIN EQUATIONS

Considering the assumptions stipulated above, the Equilibrium discharges can be described by a set of equations defined in the physics interfaces constituting the different multiphysics interfaces. Note that all the physics features available in the individual physics interfaces are also available to the multiphysics interface in use.

Energy conservation (Heat transfer in Fluids interface and Equilibrium Discharge Heat Source multiphysics coupling feature)

The energy conservation equation is used for all the Equilibrium Discharge interfaces.

$$\rho C_p \left(\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T \right) - \nabla \cdot (k \nabla T) = Q \quad (9-6)$$

The source Q (W/m^3) defined in the Equilibrium Discharge Heat Source multiphysics coupling feature includes three source/sink components:

- Resistive heating (ohmic heating)

$$Q = \mathbf{J} \cdot \mathbf{E} \quad (9-7)$$

The definition of the electric field differs following the physics interface in use.

- Equilibrium DC Discharge interface:

$$\mathbf{E} = -\nabla V \quad (9-8)$$

- Equilibrium Inductively Coupled Plasma interface

$$\mathbf{E} = -\frac{\partial \mathbf{A}}{\partial t} \quad (9-9)$$

- Combined Inductive/DC Discharge

$$\mathbf{E} = -\left(\nabla V + \frac{\partial \mathbf{A}}{\partial t}\right) \quad (9-10)$$

- Volumetric net radiation loss Q_{rad} defined by the total volumetric emission coefficient, which is a material property from [Ref. 1](#).
- Enthalpy transport (energy carried by the electric current)

$$\frac{\partial}{\partial T} \left(\frac{5k_B T}{2q} \right) (\nabla T \cdot \mathbf{J}) \quad (9-11)$$

Note that the enthalpy transport term prevails, for example, in the boundary layers close to electrodes in a fully ionized electric discharge, [Ref. 1](#).

Momentum Conservation (Laminar Flow Interface)

The momentum conservation equation is used for the Equilibrium Inductively Coupled Plasma and Combined Inductive/DC Discharge interfaces.

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = \nabla \cdot \left[-p \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} \right] + \mathbf{F} \quad (9-12)$$

Where μ is the dynamic viscosity of the fluid, \mathbf{I} is the identity matrix and

$$\mathbf{F} = \mathbf{J} \times \mathbf{B}$$

the Lorentz force acting on the fluid defined in the Magnetic Fields interface.

Current conservation (Electric Currents interface)

The Current conservation equation is used for the Equilibrium DC Discharge and Combined Inductive/DC Discharge interfaces.

$$\nabla \cdot (-\sigma \nabla V + \mathbf{J}_e) = 0 \quad (9-13)$$

The definition of the external current density differs following the physics interface in use.

- Equilibrium DC Discharge

$$\mathbf{J}_e = 0 \quad (9-14)$$

- Combined Inductive/DC Discharge interface (Induction Current Density Component multiphysics coupling feature)

$$\mathbf{J}_e = -\sigma \frac{\partial \mathbf{A}}{\partial t} \quad (9-15)$$

Ampere's law (Magnetic Fields interface)

The Ampere's law equation is used for the Equilibrium Inductively Coupled Plasma and Combined Inductive/DC Discharge interfaces.

$$\nabla \times \left(\frac{1}{\mu_r \mu_0} \nabla \times \mathbf{A} \right) = \mathbf{J}_e \quad (9-16)$$

The definition of the external current density differs following the physics interface in use.

- Equilibrium Inductively Coupled Plasma interface

$$\mathbf{J}_e = 0 \quad (9-17)$$

- Combined Inductive/DC Discharge interface (Static Current Density Component multiphysics coupling feature)

$$\mathbf{J}_e = -\sigma \nabla V \quad (9-18)$$

TRANSPORT PROPERTIES

The above equations require specification of material properties, which are functions of temperature. The Equilibrium Discharge folder in the Material Browser contains properties for density, specific heat, viscosity, thermal conductivity and electrical conductivity as a function of temperature up to 24,000 K. Available gases include air, argon, helium, hydrogen, nitrogen, and oxygen. The data is taken from the tables in the Appendix of [Ref. 1](#).

BOUNDARY EQUATIONS

The Equilibrium DC Discharge and Combined Inductive/DC Discharge interfaces include by default cooling/heating of electrodes in contact with the equilibrium

discharge [Ref. 2](#).

Ion bombardment and thermionic emission at the cathode

Positive ions from the plasma are accelerated towards the cathode and generates heat at the electrode surface. As the electrode heats up more electrons are emitted by thermionic emission which cause cooling of the cathode. The associated cathode heat flux is defined as:

$$-\mathbf{n} \cdot (-k\nabla T) = -|\mathbf{J}_{elec}|\phi_s + |\mathbf{J}_{ion}|V_{ion} \quad (9-19)$$

Where k is the thermal conductivity (W/(m-K)), ϕ_s the surface work function of the electrode (V) and V_{ion} is the ionization potential of the plasma (V). The ion current density norm is defined by:

$$|\mathbf{J}_{ion}| = |\mathbf{J} \cdot \mathbf{n}| - |\mathbf{J}_{elec}| \quad (9-20)$$

where

$$|\mathbf{J} \cdot \mathbf{n}|$$

is the normal current density at the interface and where

$$|\mathbf{J}_{elec}| = \begin{cases} J_R(T), & |\mathbf{J} \cdot \mathbf{n}| > J_R(T) \\ |\mathbf{J} \cdot \mathbf{n}|, & |\mathbf{J} \cdot \mathbf{n}| \leq J_R(T) \end{cases} \quad (9-21)$$

is the electron current density norm. The latter is defined by Richardson-Dushman current density if the total normal current density is larger than

$$J_R(T) = A_R T^2 \exp\left(-\frac{q\phi_{eff}}{k_B T}\right) \quad (9-22)$$

Where A_R is the Richardson's constant (A/(m²K²)), q is the electronic charge (C), k_B the Boltzmann's constant (J/K) and ϕ_{eff} the effective work function of the surface (V). Note that the ion current density norm $\mathbf{J}_{ion}=0$ if the Richardson-Dushman current density is larger than the total normal current at the interface.

Resistive heating at the anode

Electron entering the anode generates heat. Following the approach presented in [Ref. 2](#), we assume that there is no ion current and hence no ion heating at the anode. Accordingly the anode heat flux is defined as:

$$-\mathbf{n} \cdot (-k \nabla T) = |\mathbf{J} \cdot \mathbf{n}| \phi_s \quad (9-23)$$

Where

$$|\mathbf{J} \cdot \mathbf{n}|$$

is the normal current density at the interface, k is the thermal conductivity and ϕ_s the surface work function of the anode. Note that all the physics features available in the individual interfaces are also available to the multiphysics interface in use. This include, for instance, radiation heat losses, and wall boundary conditions.

Summary of the Equilibrium Discharge Interfaces Features

The Combined Inductive/DC Discharge contains all the equations necessary to model a thermal plasma under the assumptions stipulated at the beginning of the section. The Equilibrium DC Discharge and Equilibrium Inductively Coupled Plasma interfaces are simplified version of the Combined Inductive/DC Discharge. The following tables show the physics interfaces as well as the available multiphysics coupling features involved in each multiphysics interface.

Interface	Electric Currents	Magnetic Fields	Single phase flow (Laminar flow)	Heat transfer in Fluids
Equilibrium DC Discharge	✔			✔
Equilibrium Inductively Coupled Plasma		✔	✔	✔
Combined Inductive/DC Discharge	✔	✔	✔	✔

Figure 9-1: Physics interfaces included by default in the multiphysics interfaces.

Interface	Static Current Density Component	Induction Current density Component	Lorentz Force	Equilibrium Discharge Heat Source	Temperature Coupling	Flow Coupling	Equilibrium Discharge Boundary Heat Source
Equilibrium DC Discharge				✔	✔		✔
Equilibrium Inductively Coupled Plasma			✔	✔	✔	✔	
Combine Inductive/DC Discharge	✔	✔	✔	✔	✔	✔	✔

Figure 9-2: Multiphysics coupling features available by default in the multiphysics interfaces.

References

1. M. I. Boulos, P. Fauchais, and E. Pfender, “Thermal Plasmas: Fundamentals and Applications,” Plenum Press, Springer (1994).
2. J. J. Lowke, “A Unified Theory of Arcs and their Electrodes,” J. Phys. IV France Vol. 7 (1997) C4-283-C4-294.

I n d e x

- A**
 - absolute pressure 126
 - activation energy 222
 - Application Libraries window 22
 - application library examples
 - Boltzmann models 175
 - CCP models 255
 - contact impedance 76
 - DC models 255
 - dielectric barrier discharge 49
 - drift diffusion 192
 - electric currents 63
 - electrical circuits 114
 - electrostatics 40
 - equilibrium discharges 290
 - flux matching 244
 - heavy species transport 215
 - laminar flow 125
 - Arrhenius coefficients
 - data requirements 29
 - Arrhenius expressions 222, 243
 - Arrhenius parameters 221
- B**
 - base node 90
 - Boltzmann equation, two-term approximation interface 172
 - theory 180
 - Boltzmann model (node) 176
 - boundary current source (node) 70
 - boundary nodes
 - Boltzmann equation, two-term approximation 175, 179
 - drift diffusion 192
 - electric currents 63
 - electrostatics 40
 - heavy species transport 216
 - plasma interface 256
 - boundary stress (node) 136
 - bulk node 91
- C**
 - capacitively coupled plasma interface
 - theory 268
 - capacitor (node) 84
 - CFL number
 - settings 125
 - change cross-section (node) 57
 - change thickness (out-of-plane) (node) 57
 - charge conservation (node) 43
 - charge conservation, piezoelectric (node) 58
 - charge relaxation theory 98
 - circuit import, SPICE 96
 - closure law 236
 - collector node 90
 - collision (node) 178
 - collisionless heating (node) 262
 - combined inductive/DC discharge interface 295, 302
 - common settings 19
 - contact impedance (node) 75
 - coupling, to the electrical circuits interface 115
 - cross section import (node) 217
 - cross-section data
 - electron impact reactions, and 28
 - importing 33
 - current source (node) 86
 - current sources, theory 100
 - current-controlled current source (node) 88
 - current-controlled voltage source (node) 88
- D**
 - data requirements 28

- DC discharge interface
 - theory 271
- DC magnetic field 251
- device models, electrical circuits 105
- dielectric contact (node) 59
- dielectric shielding (node) 50
- diffusion and migration (node) 217
- diffusion models 212, 251
- diffusive flux vector 233
- diode (node) 93
- diode transistor model 111
- discharges, non-equilibrium 14
- distributed capacitance (node) 53
- distributed impedance (node) 71
- distribution functions 181
- documentation 20
- domain nodes
 - Boltzmann equation, two-term approximation 175
 - drift diffusion 192
 - electric currents 63
 - electrostatics 40
 - heavy species transport 216
 - plasma interface 256
- drain node 91
- drift diffusion interface 190
 - theory 201
- drift diffusion model (node) 193
- drift velocity 185
- drift velocity, calculating 180
- Druyvesteyn distribution function 181
- Druyvesteyn EEDF 214, 253
- E** edge nodes
 - electric currents 63
 - electrostatics 40
 - plasma interface 256
- EEDF 180
- Einstein's relation 226, 234
- elastic collisions, data requirements 29
- electric currents interface 61
 - theory 102
- electric displacement field (node) 47
- electric fields theory 98
- electric potential (node) 45
- electrical circuit interface 81
 - theory 104
- electrical circuits
 - modeling techniques 114
- electron cyclotron resonance (ECR) 281
- electron density and energy (node) 197
- electron energy distribution function 180
- electron impact reaction (node) 218
- electron impact reactions, data requirements and 28
- electron outlet (node) 200
- electron production rate (node) 196
- electron-electron collisions 182
- electrostatic point dipole (node) 60
- electrostatics interface 38
 - theory 98
- emailing COMSOL 22
- emission energy and fluxes 272
- emitter node 90
- enthalpy and entropy 237
- entrance length 131
- equation of continuity 98
- equilibrium DC discharge interface 288
- equilibrium discharge boundary heat
 - source (node) 293
- equilibrium discharge heat source (node) 290
- error message, electrical circuits 114
- exit length 134
- exporting
 - SPICE netlists 119
- external I vs. U (node) 93

- external I-terminal (node) 95
- external surface charge accumulation (node) 47
- external U vs. I (node) 94
- F**
 - floating potential (node)
 - electric currents 69
 - electrostatics 52
 - flow continuity (node) 139
 - flow coupling (node) 292
 - flow rate in SCCMs 132
 - flux (node)
 - drift diffusion 199
 - heavy species transport 229
 - forward rate constants 30
 - forward sticking coefficients 30
 - frequency domain study 103
 - frequency factor 222
 - frequency factor for reaction 237
- G**
 - gas constant 222
 - gas phase reactions 30
 - gas phase solvent 240
 - gate node 91
 - Gauss' law and charge relaxation theory 98
 - Gauss' law equation 100
 - general power deposition (node) 196
 - general stress (boundary stress condition) 136
 - generalized distribution function 181
 - gravity 142
 - ground (node) 44
 - ground node (node) 83
- H**
 - heavy species transport interface 212
 - theory 232
 - heavy species, required information 30
- I**
 - ICP reactors 274
 - importing
 - cross-section data 33
 - SPICE netlists 96, 118
 - induction current density component (node) 299
 - inductive power deposition (node) 197
 - inductively coupled plasma interface 261
 - theory 274
 - inductor (node) 84
 - inelastic collisions
 - data requirements 29
 - source coefficients, and 206
 - initial values (node)
 - Boltzmann equation, two-term approximation 177
 - drift diffusion 195
 - electric currents 67
 - electrostatics 44
 - Plasma interface 259
 - single-phase, laminar flow 127
 - inlet (node)
 - heavy species transport 229
 - single-phase flow 130
 - insulation (node) 197
 - integrodifferential 201
 - internet resources 20
 - ion flux 268
- K**
 - kinetic gas theory 238
 - Kirchhoff's circuit laws 104
 - knowledge base, COMSOL 23
- L**
 - laminar flow interface 122
 - laminar inflow (inlet boundary condition) 131
 - laminar outflow (outlet boundary condition) 134
 - law of mass action 242
 - leaking wall, wall boundary condition 129
 - line charge (node) 54
 - line charge (on axis) (node) 54

- line charge (out-of-plane) (node) 55
- line current source (node) 77
- line current source (on axis) (node) 78
- line mass source (node)
 - fluid flow 141
- local
 - CFL number 125
- Lorentz force (node) 300
- low-temperature plasmas 14
- M** magnetohydrodynamics equations 306
 - Maxwell's equations
 - electrical circuits and 104
 - Maxwellian distribution function 181
 - Maxwellian EEDF 206, 213, 252
 - Maxwell-Stefan equations 232
 - metal contact (node) 58
 - MHD equations 306
 - microwave plasma interface 264
 - theory 274, 278
 - microwave power deposition (node) 197
 - modeling plasmas 14
 - molar enthalpy and entropy 237
 - Motz-Wise correction 227
 - MPH-files 22
 - multicomponent diffusion 233
 - multicomponent mixtures 240
 - multiphysics coupling
 - combined inductive/DC discharge 295, 302
 - equilibrium DC discharge 288
 - equilibrium discharge boundary heat source 293
 - equilibrium discharge heat source 290
 - induction current density component 299
 - Lorentz force 300
 - static current density component 298
 - mutual inductance (node) 92
- N** NASA polynomials 30, 237
 - n-Channel MOS transistor 90, 108
 - n-Channel MOSFET (node) 90
 - netlists, SPICE 96, 118
 - neutral gas temperature 30
 - no slip, wall boundary condition 128
 - no viscous stress (open boundary) 135
 - nodes, common settings 19
 - nomenclature, in the Plasma Module 16
 - non-electron species 267
 - non-equilibrium discharges 14
 - normal stress, normal flow (boundary stress condition) 136
 - NPN bipolar junction transistor 89, 105
 - NPN BJT (node) 89
- O** Ohm's law and charge relaxation theory 98
 - open boundary (node)
 - single-phase flow 135
 - outflow (node) 230
 - outlet (node) 133
- P** pair contact impedance (node) 75
 - pair nodes
 - Boltzmann equation, two-term approximation 175
 - drift diffusion 192
 - electric currents 63
 - electrostatics 40
 - heavy species transport 216
 - plasma interface 256
 - p-Channel MOS transistor 91
 - p-Channel MOSFET (node) 91
 - Peclet numbers 268
 - periodic condition (node) 48
 - periodic flow condition (node) 138
 - physics interfaces, common settings 19
 - plasma 250
 - plasma model (node)

- plasma interface 257
 - plasma reactor theory 265
 - PNP bipolar junction transistor 90
 - PNP BJT (node) 90
 - point charge (node) 55
 - point charge (on axis) (node) 56
 - point current source (node) 78
 - point current source (on axis) (node) 79
 - point mass source (node)
 - fluid flow 140
 - point nodes
 - electric currents 63
 - electrostatics 40
 - plasma interface 256
 - Poisson's equation 16
 - predefined couplings, electrical circuits 115
 - pressure point constraint (node) 140
 - pseudo time stepping
 - settings 125
- R**
 - rate expression for species 233
 - reaction (node) 222
 - reaction rate 237
 - reflection coefficient 208
 - resistor (node) 83
 - RF cycle 268, 274
- S**
 - SCCM, flow rate in 132
 - sector symmetry (node) 76
 - SEMI standard E12-0303 132
 - semiconductor device models 105
 - shear rate magnitude variable 127
 - single-phase flow interface
 - laminar flow 122
 - sliding wall, wall boundary condition 128
 - slip velocity, wall boundary condition 129
 - slip, wall boundary condition 128
 - source coefficients 205, 266
 - source node 91
 - space charge density (node) 44
 - species (node) 223
 - species dipole moment 240
 - spf.sr variable 127
 - SPICE
 - exporting 119
 - SPICE netlists 14, 96, 118
 - standard cubic centimeters per minute 132
 - standard flow rate 132
 - standard settings 19
 - static current density component (node) 298
 - static pressure curves 137
 - steady-state condition 273
 - sticking coefficients 243
 - Stiel-Thodos equation 240
 - stoichiometric coefficients 242
 - stoichiometric matrix 237
 - study types
 - electric currents 103
 - subcircuit definition (node) 89
 - subcircuit instance (node) 89
 - surface charge accumulation (node) 49
 - surface charge density (node) 46
 - surface reaction (node) 227
 - surface reactions 30
 - surface reactions, theory 242
 - surface species (node) 228
 - symmetry (node) 134
- T**
 - technical support, COMSOL 22
 - temperature exponent 222
 - temperatures, ranges for 225
 - terminal (node)
 - electric currents 72
 - electrostatics 50
 - theory
 - Boltzmann equation, two-term ap-

- proximation 180
 - capacitively coupled plasma interface 268
 - DC discharges interface 271
 - drift diffusion 201
 - electric currents interface 102
 - electric fields 98
 - electrical circuit interface 104
 - electrostatics interface 98
 - heavy species transport 232
 - inductively coupled plasma interface 274
 - microwave plasma interface 274, 278
 - plasma reactors 265
 - source coefficients 205, 266
 - thermal conductivity 240
 - thermal creep, wall boundary condition 129
 - thermionic emission 193
 - thermodynamic polynomials 225
 - thermophoretic effects 225
 - thin low permittivity gap (node) 48
 - time dependent study 103
 - Townsend coefficients 185
 - Townsend coefficients theory 205, 266
 - traction boundary conditions 136
 - transformer (node) 92
- V**
 - vacuum pump (node) 137
 - velocity (node) 196
 - viscous slip, wall boundary condition 129
 - voltage source (node) 85
 - voltage-controlled current source (node) 87
 - voltage-controlled voltage source (node) 87
 - volume force (node) 127
- W**
 - wall (node)
 - drift diffusion 198
 - single-phase flow 128
 - weakly ionized 203
 - websites, COMSOL 23
- Z**
 - zero charge (node) 44
 - zero energy flux (node) 179
 - zero probability (node) 179