Particle Inspector:

The *Particle Inspector* contains the following options for use with particles:

Creating/Removing and Browsing the current Particles

paste_11.tiff ¬

The *New Particle* button creates and adds another particle to the current list of particles and displays its default name and default parameter settings. The user can create as many particles as possible within the memory limits of the machine.

The *Delete Particle* button deletes the current particle whose attributes are currently being displayed.
If no more particles exist because of this action, certain buttons will become disabled until the *New Particle* button again creates a particle.

The arrow buttons allow the user to browse the particles and to change the currently displayed

particle. When the arrows try to move before the first particle or after the last particle, the inspector wraps around and displays the beginning or end of the list of particles.

Whenever the *Particle Inspector* panel is displaying a particle, the user may change the name of the current particle at any time by editing the particle name's text field:

paste_28.tiff ¬

Particle Attributes:

paste_1.tiff ¬ **paste_2.tiff ¬**

A particle has basic and advanced parameters which can be viewed/changed using the *Parameters* pop-up button.

Basic Parameters:

paste_30.tiff ¬

The initial position and initial velocity vectors of the particle can be set by the two sets of four initial position and velocity text fields. Each set of four fields represents the x-component of

the vector, the y-component of the vector, the magnitude of the vector, and the angle (theta) that the vector makes with the x-axis of the coordinate system. All values in *PhysicsWorld* are taken to be in the International System of Units (SI). In this case, meters and meters/second are the units to use for entering the initial position and initial velocity vectors respectively. When the user enters values and **presses the *Return* key** the other fields are updated as well. For example, if the user entered "10" for the x component of the initial position vector, the magnitude field of the initial position vector would immediately change to "10" as well.

If the user did not press the return key after entering new values for any of the fields, there would exist an inconsistent set of values between the four fields. However, *PhysicsWorld* always gets the initial vectors from the x and y component fields, so that if the magnitude or theta fields were changed without the x and y components being updated, the changes would not be recorded.

The default value of the initial position and the initial velocity is (0 m, 0 m) and (0 m/s, 0 m/s) respectively.

The charge of the particle is in units of Coulomb (C). The default value of charge for a particle is 0 Coulombs.

The mass of the particle is in units of kilograms (kg). The default value of mass for a

particle is 1 kg. If the user tries to enter a negative value of mass, for example "-10 kg", then *PhysicsWorld* replaces this choice with its absolute value ("10 kg"). If the user tries to enter 0 kg (zero kilograms) for the mass of the particle, *PhysicsWorld* will replace this with the default value of 1 kg.

A particle can be given a finite size with which to see the particle. The default value of a particle's size is 0 meters. Even with a value of zero meters for its size, a particle can still be seen as a single dot when viewing the simulation.

paste_35.tiff ¬

The color of a particle may be set by dragging in a new color using the *Color Panel* found in the *Tools* menu. The default color is dark gray. Every particle is filled with this color, and an outline of black is drawn on the border of every particle.

paste_36.tiff ¬ paste_37.tiff ¬

The capabilities of the particle to move in space is set by the *movement* pop-up list. The default value of particle movement is *Free* and this simply allows the particle to move wherever the simulation and its forces causes the particle to go. The user may choose to constrain the movement of a particle in either the x, y, or both x and y directions. This allows for fixed particles to act as stationary sources for specific simulations or, in the case of

simulating wave phenomena, to allow the particle to move in only one direction based on where it has been initially placed. The user should note that constrained motion is a feature which may create very unrealistic and energy losing simulations.

Advanced Parameters:

paste_0.tiff ¬

The user has the option to add resistive forces that are proportional to some power of velocity to simulation for example, air resistance or viscosity. The default values are zero, so that particles do not initially contain any built in effects of air resistance. The parameters are entered with a radius term already factored in. This allows for the same coefficients to be used for all particles, and thus for different sized particles to create a different individual resistive effect. Typical values to use in modeling resistance due to air are: A1=1, B1 = 0.00031 kg/ m·s (a linear velocity term) and A2=2, B2 = 0.87 kg/m^3 (a squared velocity term) together.

paste_39.tiff ¬

A particle may also be given a constant acceleration term throughout the simulation independent of the other particles, external fields, etc. This allows for the setup of simple

kinematics problems such as: "Car A has an acceleration of 3 m/s^2 and Car B has an acceleration of 10 m/s^2... ". The default value of the constant acceleration term is 0 m/s^2.

paste_3.tiff ¬

The user can specify a forcing function on an object. The function can be entered in terms of components or in terms of magnitude and direction. The Expression object used was written by David Holscher at Rose-Hulman Institute of Technology for the Freshman Integrated Curriculum.

The function can be any valid mathematical function according to the rules that follow. When the function is typed in, hit the return key. The parser then checks the expression for errors. If there are any errors the function will disappear.

PhysicsWorld defined Variables

The following variables will be accepted as part of the function:

Variable	**Description**
time	the current time of the simulation
t	"

mass	the mass of the particle
m	"
inittime	the initialization time of the particle
charge	the charge of the particle
q	"
rx	x-component of position of the particle
ry	y-component of position of the particle
vx	x-component of velocity of the particle
vy	y-component of velocity of the particle
radius	the radius of the particle

Built-in-operators

Operator	**Description**
+	binary addition
-	binary subtraction, unary negation
/	binary division
*	binary multiplication
%	binary modulus
^	binary exponential

Conditionals

Conditionals evaluate an expression for TRUE or FALSE (1 or 0). If the expression is TRUE the expression to the left of the ELSE is returned, otherwise the conditional to the right of the ELSE is returned.

Conditional Format

(expression) IF expression THEN expression ELSE expression ENDIF

Conditional Operators	Description
<	less than
>	greater than
<=	less than or equal
>=	greater than or equal
==	equal
!=	not equal
!	(unary) negation
||	logical or
&&	logical and

Associativity and Order of Operations

Operators are listed in order of precendence with operators of equal precedence on each line.

Operator	Description
^	right to left
- !	right to left
* / %	left ro right
+ -	left to right
< > <= >=	left to right
== !=	left to right
&&	left to right
||	left to right

Constants

Constant	Value
E	2.7182818284590452354
LOG2E	1.4426950408889634074
LOG10E	0.43429448190325182765
LN2	0.69314718055994530942
LN10	2.30258509299404568402
PI	3.14159265358979323846

```
PI_2        1.57079632679489661923
PI_4        0.78539816339744830962
1_PI        0.31830988618379067154
2_PI        0.63661977236758134308
2_SQRTPI        1.12837916709551257390
SQRT2       1.41421356237309504880
SQRT1_2     0.70710678118654752440
MAXDOUBLE       1.79769313486231570e308
MINDOUBLE       4.94065645841246544e-324
```

Built-in-functions

Function	No of Parameters	C-Function Called
abs	1	abs
acos	1	acos
acosh	1	acosh
asin	1	asin
asinh	1	asinh
atan	1	atan
atanh	1	atanh
atan2	2	atan2
cbrt	1	cbrt
ceil	1	ceil
copysign	2	copysign

```
cos     1       cos
cosh        1       cosh
drem        2       drem
erf     1       erf
erfc        1       erfc
exp     1       exp
expm1       1       expm1
fabs        1       fabs
finite      1       finite
floor       1       floor
fmod        2       fmod
hypot       2       hypot
jn      2       jn
j0      1       j0
j1      1       j1
lgamma      1       lgamma
ln      1       log
logb        1       logb
log1p       1       log1p
log10       1       log10
pow     2       pow
rand        0       random (is called then divided by MAXINT)
rint        1       rint
scalb       2       scalb
sin     1       sin
sinh        1       sinh
```

```
sqrt        1       sqrt
tan     1       tan
tanh        1       tanh
yn      2       yn
y0      1       y0
y1      1       y1
```

Do not leave out math operators. i.e. If you are multiplying two parameters, put a * between them.

paste_40.tiff ¬

An extra, specialized feature of a particle is an initialization time. This allows for particles to "kick in" at some later time than the default of immediately at time zero. The radio buttons determine how the particle will act before the particle "kicks in" if in fact, it has a non-zero initialization time.

For example, if the particle were told to have an initialization time of 10 seconds, and during the time until initialization it was "invisible and inactive" then we would essentially have a simulation up until time 10 seconds which in no way knows about this particle. At time equals 10 seconds, the particle appears given its initial parameters and moves through the simulation acting on and being acted on upon by the other objects in the simulation. This mode could be used to create a firing source. The user could create a number of particles all

starting from the same place, but with different initialization times. As the simulation is running the various particles appear at their respective times.

If during the time until initialization, the particle was "visible and inactive" then we would simply "see" the particle, but it completely ignores and is completely ignored by the rest of the simulation until its initialization time. This mode is useful for certain simple kinematics problems where, for example: "Car A takes off at 10 meters/second, and then 10 seconds later Car B takes off after Car A with 20 meters/second...".

The third mode allows a particle to exist before its initialization mode and interact with the simulation, by either falling in a gravitational field, moving from the repulsion of another particle, etc. However, at its time of initialization, its velocity is set to its initial velocity so as to simulate an impulse at some time later than time zero.

The *OK* button accepts all of the recent changes made to the current particle, reinitializes the simulation and closes the *Particle Inspector* panel.