Contenido

[Introducción 3](#__RefHeading__964_162444972)

[¿Qué es NetLogo? 4](#__RefHeading__966_162444972)

[Características 4](#__RefHeading__968_162444972)

[Ejemplo de Modelo: Fiesta 6](#__RefHeading__970_162444972)

[En la Fiesta 6](#__RefHeading__972_162444972)

[El Reto 8](#__RefHeading__974_162444972)

[Pensando en modelos 8](#__RefHeading__976_162444972)

[Tutorial #1: Modelos 9](#__RefHeading__978_162444972)

[Modelo de muestra: Depredación lobo-oveja 9](#__RefHeading__980_162444972)

[Controlando el modelo: botones 9](#__RefHeading__982_162444972)

[Controlando la velocidad: Barra de velocidad 9](#__RefHeading__984_162444972)

[Ajustando la configuración: Sliders y Switches 9](#__RefHeading__986_162444972)

[Recopiladores de información: Graficas y Monitores 10](#__RefHeading__988_162444972)

[Graficas 10](#__RefHeading__990_162444972)

[Monitores 10](#__RefHeading__992_162444972)

[Controlando la vista 10](#__RefHeading__994_162444972)

[Biblioteca de modelos 11](#__RefHeading__996_162444972)

[Modelos de muestra 11](#__RefHeading__998_162444972)

[Modelos curriculares 11](#__RefHeading__1000_162444972)

[Ejemplos de códigos 11](#__RefHeading__1002_162444972)

[Actividades HubNet 11](#__RefHeading__1004_162444972)

[Tutorial #2: Comandos 11](#__RefHeading__1006_162444972)

[Modelo de muestra: Tráfico básico 11](#__RefHeading__1008_162444972)

[Centro de Comandos 12](#__RefHeading__1010_162444972)

[Trabajando con colores 12](#__RefHeading__1012_162444972)

[Agente Monitores y Agente de Comandos 13](#__RefHeading__1014_162444972)

[¿Qué es NetLogo? 2](#__RefHeading__964_162444972)

[Características 3](#__RefHeading__964_162444972)

[Ejemplo de Modelo: Fiesta 4](#__RefHeading__964_162444972)

[En la Fiesta 5](#__RefHeading__964_162444972)

[El Reto 6](#__RefHeading__964_162444972)

[Pensando en modelos 7](#__RefHeading__964_162444972)

[Tutorial #1: Modelos 7](#__RefHeading__964_162444972)

[Modelo de muestra: Depredación lobo-oveja 7](#__RefHeading__964_162444972)

[Controlando el modelo: botones 7](#__RefHeading__964_162444972)

[Controlando la velocidad: Barra de velocidad 8](#__RefHeading__964_162444972)

[Ajustando la configuración: Sliders y Switches 8](#__RefHeading__964_162444972)

[Recopiladores de información: Graficas y Monitores 8](#__RefHeading__964_162444972)

[Graficas 9](#__RefHeading__964_162444972)

[Monitores 9](#__RefHeading__964_162444972)

[Controlando la vista 9](#__RefHeading__964_162444972)

[Biblioteca de modelos 9](#__RefHeading__964_162444972)

[Modelos de muestra 9](#__RefHeading__964_162444972)

[Modelos curriculares 10](#__RefHeading__964_162444972)

[Ejemplos de códigos 10](#__RefHeading__964_162444972)

[Actividades HubNet 10](#__RefHeading__964_162444972)

[Tutorial #2: Comandos 10](#__RefHeading__964_162444972)

[Modelo de muestra: Tráfico básico 10](#__RefHeading__964_162444972)

[Centro de Comandos 10](#__RefHeading__964_162444972)

[Trabajando con colores 11](#__RefHeading__964_162444972)

[Agente Monitores y Agente de Comandos 11](#__RefHeading__964_162444972)

[¿Qué es NetLogo? 1](#__RefHeading__964_162444972)

[Características 1](#__RefHeading__964_162444972)

[Ejemplo de Modelo: Fiesta 3](#__RefHeading__964_162444972)

[En la Fiesta 3](#__RefHeading__964_162444972)

[El Reto 5](#__RefHeading__964_162444972)

[Pensando en modelos 5](#__RefHeading__964_162444972)

[Tutorial #1: Modelos 6](#__RefHeading__964_162444972)

[Modelo de muestra: Depredación lobo-oveja 6](#__RefHeading__964_162444972)

[Controlando el modelo: botones 6](#__RefHeading__964_162444972)

[Controlando la velocidad: Barra de velocidad 6](#__RefHeading__964_162444972)

[Ajustando la configuración: Sliders y Switches 7](#__RefHeading__964_162444972)

[Recopiladores de información: Graficas y Monitores 7](#__RefHeading__964_162444972)

[Graficas 7](#__RefHeading__964_162444972)

[Monitores 7](#__RefHeading__964_162444972)

[Controlando la vista 8](#__RefHeading__964_162444972)

[Biblioteca de modelos 8](#__RefHeading__964_162444972)

[Modelos de muestra 8](#__RefHeading__964_162444972)

[Modelos curriculares 8](#__RefHeading__964_162444972)

[Ejemplos de códigos 8](#__RefHeading__964_162444972)

[Actividades HubNet 8](#__RefHeading__964_162444972)

[Tutorial #2: Comandos 9](#__RefHeading__964_162444972)

[Modelo de muestra: Tráfico básico 9](#__RefHeading__964_162444972)

[Centro de Comandos 9](#__RefHeading__964_162444972)

[Trabajando con colores 9](#__RefHeading__964_162444972)

[Agente Monitores y Agente de Comandos 10](#__RefHeading__964_162444972)

# Introducción

NetLogo es un ambiente de modelado multi-agente programable, es usado por decenas y cientos de estudiantes, maestros e investigadores de todo el mundo, también tiene el poder de HubNet en simulaciones participativas. Es autorizado por Uri Wilensky y desarrollado en el CCL.

## ¿Qué es NetLogo?

* Es un ambiente de modelado programable, para simular fenómenos naturales y sociales. Fue desarrollado por Uri Wilensky en 1999 y ha continuado en desarrollo desde entonces en el Centro de Aprendizaje Conectado y Modelado por Computadora.
* Es particularmente útil en el modelado de sistemas complejos desarrollado sobre tiempo. Los modeladores pueden dar instrucciones a cientos o miles de “agentes”, todos operando independientemente. Esto hace posible explorar las conexiones entre el comportamiento a micro-nivel de los individuos y macro-nivel de los patrones que surgen de sus interacciones.
* Deja a los estudiantes abrir simulaciones y ejecutarles, explorando su comportamiento en diferentes condiciones. También es un entorno de edición que permite a los estudiantes, profesores y especialistas curriculares crear sus propios modelos. NetLogo es bastante simple para los estudiantes y profesores, sin embargo también es lo suficientemente avanzado como para servir como una poderosa herramienta para los investigadores de distintas áreas.
* Tiene una extensa documentación y tutoriales. También tiene una librería de modelos, una gran colección de simulaciones pre-escritas que pueden ser usadas y modificadas. Estas simulaciones se localizan en áreas sociales y naturales incluyendo: biología y medicina, física y química, matemáticas y ciencias de la computación, y economía y psicología social. Varios programas de estudio basados en modelos de consulta que utilizan NetLogo están disponibles y otros más están en desarrollo.
* Puede también fortalecer las clases con una herramienta de simulación participativa llamada HubNet. A través del uso de redes de computadoras o de la mano de dispositivos como calculadoras graficadoras Texas Instruments, cada estudiante puede controlar un agente en una simulación.
* Es la siguiente generación de lenguajes de modelado multi-agente incluyendo StarLogo y StarLogoT. NetLogo corre sobre la JVM, por lo que trabaja sobre múltiples plataformas (Mac, Windows, Linux et al). Corre independientemente de la aplicación. Los modelos y actividades de HubNet pueden ser ejecutadas como Java applets en un explorador. La operación de comandos en línea también es soportada.

## Características

* Sistema
  + Libre, open source
  + Multi-Plataforma: corre sobre Mac, Windows, Linux et al
  + Soporte de caracteres internacionales
* Programación
  + Totalmente programable
  + Sintaxis accesible
  + Su lenguaje es Logo, dialecto extendido para soportar agentes
  + Agentes móviles (tortugas) se mueven sobre una malla de agentes estacionarios (patches)
  + Los enlaces de agentes conectan tortugas para hacer redes, grafos y agregados
  + Extenso vocabulario de conexión de primitivas de lenguaje
  + Punto flotante de doble precisión
  + *Primera-clase valores de funciones (también conocidas como task, cierre, lambda)*
  + Las ejecuciones se pueden reproducir en múltiples plataformas
* Ambiente
  + Centro de mando para interacción sobre la marcha
  + Interface de construcción, w/botones, sliders, switchers, chooser, monitors, cuadros de texto, notas, áreas de salida
  + Información tab para anotaciones de tu modelo, con formato de texto e imágenes
  + HubNet: simulación participativa utilizando dispositivos en red
  + Monitor de agentes, para inspeccionar y controlar agentes
  + Importación y exportación de funciones (exportar datos, guardar y restaurar estados del modelo, hacer una película)
  + BehaviorSpace, una herramienta open source usada para colectar datos de múltiples ejecuciones paralelas de un modelo
  + Modelador de Sistemas Dinámico
  + NetLogo 3D para modelado de mundos 3D
  + Modo Headless que permite hacer lotes de ejecutables, desde comandos en línea
* Display y visualización
  + Línea, barras y gráficos de dispersión (scatter plots)
  + El regulador de velocidad te permite avanzar rápidamente en tu modelo o verlo en cámara lenta
  + Vista de modelo en modelos ya sea 2D y 3D
  + Formas de vector rotables y escalables
  + Etiquetas de tortugas y patch
* Web
  + Los modelos y clientes HubNet pueden ser guardados como applets para ser embebidos en paginas web (nota: algunas características no están disponibles en applets, por ejemplo 3D)
* APIs
  + Controlando una API permite embeber NetLogo en un script aplicación
  + Las extensiones API permiten agregar nuevos comandos y reportes para el lenguaje NetLogo; ejemplos de extensiones open source son incluidos

# Ejemplo de Modelo: Fiesta

Esta actividad se da pensando en un modelado por computadora, de como puedes usarlo. Animamos a los usuarios principiantes a empezar aquí.

## En la Fiesta

¿Has estado en una fiesta y has notado que la gente se junta en grupos? También podrías observar que las personas no se mantienen siempre en un grupo. Ya que circulan, los grupos cambian. Si has visto estos cambios en el tiempo, puedes notar patrones.

Por ejemplo, en configuraciones sociales, las personas pueden exhibir diferentes comportamientos en el trabajo o en la casa, individuos que son confiados dentro de su ambiente de trabajo, pueden llegar a ser tímidos en una reunión social. Y otros que son reservados en su trabajo pueden llegar a ser “el alma de la fiesta” con sus amigos.

Estos patrones pueden depender del tipo de reunión. En algunas configuraciones, las personas están capacitadas para organizarse en grupos mixtos; por ejemplo, juegos de fiestas o actividades de tipo escolar. Pero en una atmosfera no estructurada, la gente tiende a agruparse en formas más aleatorias.

***¿Existen ciertos patrones para estos tipos de agrupamientos?***

Echemos un vistazo más de cerca a esta pregunta mediante el uso de la computadora para modelar el comportamiento humano en una fiesta. El modelo de fiesta en NetLogo se centra específicamente en la cuestión de agrupación por género: ¿los grupos que tienden a formarse son en su mayoría de hombre o mujeres?

* **¿Qué hacer?**

1. Iniciar NetLogo
2. Elegir “Librería de modelos”, en el menú de Archivo
3. Abrir el folder “Ciencia Social”
4. Click en el modelo “Party”
5. Presionar el botón “open”
6. Presionar el botón “setup”

En la vista de modelo, veras grupos rosas u azules con números.

Estas líneas representan mezclas de grupos en una fiesta. Los hombres se muestran en azul y las mujeres en rosa. Los números son el tamaño de los grupos.

***¿Todos los grupos tienen aproximadamente el mismo número de personas?***

***¿Todos los grupos tienen aproximadamente el mismo número de personas del mismo sexo?***

Digamos que usted está teniendo una fiesta e invitó a 150 personas. Se pregunta como se reunirán las personas. Supongamos que se forman 10 grupos en la fiesta.

***¿Cómo piensa usted que ellos se agruparán?***

En lugar de pedir a 150 de sus amigos que se agrupen al azar, vamos a simular por computadora esta situación.

**¿Qué hacer?**

1. Presione el botón “go” (presionando nuevamente “go” puede para el modelo manualmente).
2. Observe el movimiento de las personas hasta que el modelo se detenga.
3. Mira las gráficas para ver de otra manera lo que está sucediendo.
4. Usa la barra de velocidad si necesitas ver el modelo más lentamente.

***¿Ahora cuántas personas hay en cada grupo?***

En un principio, puede que hayas pensado en dividir 150 en 10 grupos, dando 15 personas por cada grupo. A partir del modelo, podemos ver que las personas eventualmente no se dividen en 10 grupos. En lugar de esto, algunos grupos llegan a ser muy pequeños, mientras que otros grupos llegan a ser muy grandes. Además, la fiesta cambia con el tiempo de los grupos mezclados de hombres y mujeres a grupos de un solo sexo.

***¿Cómo podemos explicar esto?***

Hay muchas respuestas a esta pregunta acerca de lo que se espera que suceda en una fiesta real. El diseñador de esta simulación pensó que los grupos en las fiestas no se forman aleatoriamente. Los grupos son determinados por el comportamiento de los individuos en las fiestas. El diseñador decidió enfocarse en una variable en particular, la “tolerancia”.

La tolerancia es definida aquí, como el porcentaje de personas de sexo opuesto con que un individuo se siente “cómodo”. Si el individuo está en un grupo que tiene un mayor porcentaje de personas del sexo opuesto que su tolerancia permita, entonces se sentirá incomodo y dejará su grupo para buscar otro.

Por ejemplo, si el nivel de tolerancia está al 25%, entonces los hombres sólo se sentirán “cómodos” en grupos donde menos del 25% de personas son mujeres y viceversa.

Como los individuos, llegan a sentirse “incómodos”, y dejan los grupos entonces se cambian a nuevos grupos, lo que puede causar que algunas personas de estos grupos lleguen a sentirse “incomodos” en la siguiente vuelta. Esta reacción en cadena continúa en la fiesta hasta que todos están “cómodos” en sus grupos.

Note que en el modelo la “tolerancia” no está fija. El usuario puede usar la barra de “tolerancia” para probar diferentes porcentajes y ver que salidas da al iniciar de nuevo el modelo.

**Como llegar hasta aquí:**

1. Si el botón “go” está presionado (negro), entonces el modelo está corriendo. Presione el botón de nuevo para pararlo.
2. Ajuste la barra de “tolerancia”, arrastrando la barra roja, un nuevo valor.
3. Presione el botón “setup” para reiniciar el módelo.
4. Presione el botón “go” para ejecutar de nuevo el modelo.

## El Reto

Como el anfitrión de la fiesta, quisieras ver tanto hombres como mujeres mezclándose en grupos. Ajusta la barra de tolerancia del lado de la vista para lograr como resultado final que todos los grupos se mezclen.

***Asegúrate de que los 10 grupos tengan ambos sexos, ¿en que nivel de tolerancia podrías lograr esto?***

Prueba tus predicciones en el modelo.

***¿Puedes ver otros factores o variables que pueden afectar la relación hombre-mujer dentro de un grupo?***

Has predicciones y prueba tus ideas en este modelo

Cómo estás probando tus hipótesis, puedes notar los patrones que están emergiendo de los datos. Por ejemplo, si mantienes constante el número de personas en la fiesta, pero gradualmente incrementas el nivel de tolerancia, aparecerán más grupos mezclados.

***¿Qué tan alto es el valor de tolerancia que se debe tener antes que comiences a mezclar los grupos?***

***¿Qué tanto porciento de tolerancia tiende a producir el porcentaje de mezcla***

## Pensando en modelos

Usar NetLogo para modelar una situación, como una fiesta, te permite experimentar con el sistema de una manera rápida y flexible, que sería difícil en el mundo real. El modelado también da la oportunidad de observar una situación o circunstancia con menos dificultades, puedes examinar dinámicas subyacentes de una situación. Puedes encontrar en tus modelos muchas ideas preconcebidas, acerca de varios fenómenos. Por ejemplo, un resultado sorpresivo en el modelo de la fiesta es que siempre que la tolerancia es relativamente alta, ocurre una gran separación entre sexos.

Este es un ejemplo clásico de un fenómeno “emergente”, donde un grupo de patrones resulta de una interacción de muchos individuos. Esta idea de fenómeno “emergente” se puede aplicar a casi cualquier tema.

***¿En qué otro fenómeno emergente podemos pensar?***

Para ver más ejemplos y lograr un entendimiento más profundo de este concepto y de como NetLogo ayudar a enseñar a explorar esto, puedes explorar los modelos de la librería. Esta contiene modelo que demuestran estas ideas en sistemas de todos tipos.

Para una extensa discusión de emergencia y de como NetLogo ayuda a explorar esto ver  ["Modeling Nature's Emergent Patterns with Multi-agent Languages"](http://ccl.northwestern.edu/papers/MEE/) (Wilensky, 2001).

# Tutorial #1: Modelos

Mientras exploras los modelos de la librería, en esta sección vamos a ir más profundo en las características que están disponibles.

A través de todos los tutoriales, nos preguntaremos por las predicciones que puedes hacer sobre los efectos que tienen los cambios hechos en un modelo. Teniendo en cuenta que los efectos continuamente nos sorprenden. Consideramos que estas sorpresas son excitantes y proveen excelentes oportunidades de aprendizaje.

## Modelo de muestra: Depredación lobo-oveja

Abriremos uno de los modelos de ejemplo y lo exploraremos a detalle. Probaremos con un modelo de biología: depredación lobo-oveja, un modelo de población depredador-presa.

La pestaña de interface se llenara con varios: botones, switches, sliders y monitores. Estos elementos de interface permiten que interactúes con el modelo. Los sliders y switches son verdes; y alternan la configuración del modelo. Los monitores y gráficas son beige, ellos muestran los datos.

Si quieres hacer la ventana más grande, los elementos se ven mejor, puedes usar el menú zoom.

Cuando abres por primera vez el modelo, notarán que la “vista” (la pantalla de agentes en el modelo) esta vacío (es todo negro). Para cargar el modelo hay que presionar “setup”.

## Controlando el modelo: botones

Cuando un botón es presionado el modelo responde con una acción. Un botón puede ser “de una vez” o de “para siempre”. Puedes ver la diferencia de estos dos tipos de botones con un símbolo en la cara del botón. El botón de tipo “para siempre” tiene dos flechas en la esquina inferior derecha (como el botón “go”) los botones de tipo “una vez”, no tienen esas flechas (como el botón setup).

Los botones once ejecutan una acción y después paran, cuando la acción termina, el botón bota.

Los botones forever hacen una acción una y otra vez, cuando quieres que la acción pare, presiona el botón una vez. Esto terminará la actual acción y el botón saltará.

Más modelos, incluyendo depredación lobo-oveja tienen un botón once llamado “setup” y un botón forever llamado “go”. Algunos modelos tienen un botón once llamado “go once” o “step once” que es como “go”, excepto por que avanza el modelo sólo un tick de reloj. Usando botones onces como estos puedes ver más claramente el progreso del modelo.

Parar un botón forever es la menra normal de pausar o parar un modelo, después de una pausa puedes reanudar pulsando el botón de nuevo (también puedes parar el modelo con el ítem “halt”, en el menú de herramientas, pero sólo puedes hacer esto si el modelo se trabo o ciclo por alguna razón, usando “detener/halt” puedes interrumpir el modelo a mitad de la acción y como resultado el modelo podría ser confuso)

***¿Alguna vez se obtienen resultados diferentes si se ejecuta el modelo varias veces con la misma configuración?***

## Controlando la velocidad: Barra de velocidad

La barra de velocidad te permite controlar la velocidad del modelo, esto es la velocidad a la cual se mueven las tortugas, los parches cambian de color y así sucesivamente.

Cuando mueves la barra de velocidad a la izquierda el modelo baja su velocidad, ahora las pausas entre cada tick son más largas. Esto hace más fácil ver lo que esta sucediendo. Puedes bajar la velocidad tanto como para ver lo que una sola tortuga está haciendo.

Cuando mueves la barra de velocidad a la derecha, el modelo aumenta su velocidad. NetLogo comenzará a saltar frames, esto es no actualizará la vista al final de cada tick, sólo en algunos ticks. Actualizar toma tiempo, entonces menos actualizaciones graficas hacen que el modelo avance más rápido.

Nota que si colocas la barra de velocidad muy a la derecha, la vista se puede actualizar con tan poca frecuencia que modelo parece haber disminuido, esto no tien que ver con observar el contador de carrera adelantado, sino que solo la actualización de las vistas ha disminuido.

## Ajustando la configuración: Sliders y Switches

La configuración del modelo te deja ver diferentes escenarios o hipótesis. Alternando la configuración y después ejecutando el modelo puedes ver como reacciona y tener una comprensión más profunda de los fenómenos que se están modelando.

Los switches y sliders te dan acceso a la configuración del modelo.

Veamos que pasa cambiando la configuración del modelo. Cambiando el switch grass, se afecta la salida del modelo. Con el swith en off la cantidad de pasto disponible siempre se mantiene igual, esta es una vista no realista de la relación depredador-presa; entonces por configuración y cambiando la velocidad de crecimiento del pasto, tenemos tres factores disponible en el modelo: población de ovejas, lobos y pasto.

Otro tipo de elemento de configuración el la barra/slider.

Además de los switches, el modelo también tiene sliders. Mientras un switch tiene sólo dos valores, on y off, un slide tiene un enorme rango de valores numéricos. Por ejemplo “la cantidad inicial de ovejas”, el slider tiene un minimo valor de 0 y máximo de 250. El modelo puede ejecutarse con inicialmente con 0, 250 ó alguna cantidad intermedia.

Cuando abres un modelo, todos los sliders y swtches tienen una configuración por default. Si tu abres un nuevo modelo o sales del programa, tu configuración se perderá (si no la salvaste).

Nota: Además de sliders y switches, algunos modelos tienen choosers e input boxes. El depredador lobo-oveja no tiene ninguno de estos.

## Recopiladores de información: Graficas y Monitores

El visor nos deja ver que esta sucediendo en el modelo. NetLogo también provee otras maneras de obtener información del modelo que esta corriendo, tales como graficas y monitores.

### Graficas

La grafica en el depredador lobo-oveja tiene 3 líneas ovejas, lobos y pasto/4. (el contador de pasto es dividido entre 4, ya que la grafica no tiene demasiados valores en el eje ‘y’). Las líneas muestran lo que sucede a cada momento. Las leyendas de la grafica muestran lo que cada línea indica. En este caso el crecimiento de la población.

Cuando una grafica está apunto de llenarse, el eje horizontal es comprimido con sus datos, en un espacio más pequeño. De esta manera se hace más espacio para la grafica.

Si quieres guardar una grafica usa el ítem “export plot” del menú archivo, esto guarda los datos de la grafica en un formato que puede ser leído por una hoja de cálculo y una base de datos en programas como Excel. También puedes exportar una grafica dado click derecho sobre ella y eligiendo la opción en el menú pop.

### Monitores

Son otra manera de desplegar información de un modelo

En el ejemplo, los monitores muestran la población de lobos, ovejas y la cantidad de pasto. Los números desplegados en lo monitores cambian conforme el modelo avanza, mientras las graficas muestran los datos en conjunto.

## Controlando la vista

El avance rápido del modelo y apagar la actualización de la vista resulta útil si eres impaciente y quieres que el modelo corra rápido. EL avance rápido (moviendo la barra de velocidad hacia la derecha) deja la actualización de las vista y ahora el modelo puede correr rápido. Ya que la actualización del modelo toma tiempo que podría ser usado para ejecutar el modelo

Cuando las actualización de la vista está apagada, el modelo continua ejecutándose en segundo plano, y las gráficas y los monitores se mantienen actualizados. Pero si quieres ver que pasa activar la actualización de la vista. Algunos modelos corren mucho más rápido cuando la actualización de la vista está apagada, en otros sólo se nota una pequeña diferencia.

El tamaño de la vista es determinada por cinco configuraciones diferentes: min-pxcor, maxpxcor, minpycor, max pycor y patch size.

Colocando el mouse fuera de la vista puedes ver que se convierte en una cruz, algunos elementos de configuración (además de los monitores y vistas) se pueden ahora seleccionar y cambiar de tamaño (estos cambios se verán reflejados si abres nuevamente la venta de configuración).

NetLogo es una malla de parches de dos dimensiones. Los parches son los cuadros individuales en la malla. En el depredador lobo-oveja, cuando se cambia la configuración del pasto se hace sobre parches individuales, esto se puede ver fácilmente por que algunos son verde y otros cafés.

Los parches se pueden ver como pequeños azulejos de un piso de azulejo. Por default el patch que esta exactamente a la mitad tiene la etiqueta (0,0), de manera que podemos tener un sistema coordinado que nos ayude a ubicar objetos.

El número total de azulejos de izquierda a derecha es llamado ancho del mundo y el total de arriba abajo se llama alto del mundo.

![A description...](data:None;base64,)

Cuando cambias el tamaño de patch, no cambia el número de patch, sino que los patch se hacen más grandes o pequeños.

## Biblioteca de modelos

La biblioteca de modelos tiene cuatro secciones: modelos de muestra, curricula de modelos, códigos de muestra y actividades HubNet.

### Modelos de muestra

Esta organizada por áreas y actualmente contiene más de 200 modelos. Trabajamos contantemente en añadir nuevos modelos.

Algunos de los folders tienen una etiqueta que dice “(no verificado)”. Estos modelos están completos y funcionan, pero están en proceso se de ser revisados en cuanto a contenido, precisión y calidad de código.

### Modelos curriculares

Hay modelos diseñados para ser usados en escuelas en el contexto de la curricula desarrollada en Northwestern University, algunos de ellos también ha sido usado como modelos de muestra; otros únicamente están en esta sección.

### Ejemplos de códigos

Hay demostraciones simples de características particulares de NetLogo. Son útiles cuando extiendas modelos existentes o construyas unos nuevos. Por ejemplo si quieres agregar histogramas tu modelos, puedes ver “Histrogram Example”.

### Actividades HubNet

Esta sección contiene simulaciones participativas en grupo. Para mas información consultar la guía HubNet.

# Tutorial #2: Comandos

En esta sección nos enfocaremos a comenzar a cambiar la forma de los modelos y manipularlos. Comenzarás a ver el funcionamiento interno de los modelos y estarás listo para cambiar su aspecto.

## Modelo de muestra: Tráfico básico

En este modelo poder ver un carro rojo en un flujo de carros azules. El flujo de carros se mueve en la misma dirección. De vez en cuando se apilan y dejan de moverse. Este es el modelo de como un embotellamiento puede formase sin una causa especifica, como un accidente.

Puedes alterar las configuraciones y observar algunas ejecuciones para tener una mejor compresión del modelo.

Puedes observar que el modelo es bastante simple; un fondo negro con una calle blanca y algunos carros azules y uno rojo. Los cambios que se podrían hacer par a el modelo incluyen: cambio de color y forma de los carros, agregar una casa o farol, crear una luz de alto o crear otra línea de tráfico. Algunas de estas sugerencias son cambios visuales, para mejorar el aspecto del modelo, mientras otras se refieren al comportamiento. En este tutorial nos enfocaremos más a cambios simples o estéticos. (El tutorial #3 presentara grandes detales acerca de cambios en el comportamiento, que requieren cambios la lengüeta de código).

Para hacer estos simples cambios usaremos el Centro de Comandos.

## Centro de Comandos

El centro de comandos se encuentra en la lengüeta de interfaz. Permite introducir comandos o direccione sal modelo. Los comandos son instrucciones que puedes dar a agentes de NetLogo: tortugas, parches, enlaces y observadores.

Puedes escribir en el centro de comandos *observer> ask patches [set pcolor yellow]* y verás que el fondo del visor cambia a amarillo y la calle desaparece. Pedimos que solamente los parches cambiaran de color. En este modelo, los carros se representan por otro tipo de agentes llamados tortugas, para cambiar su color ponemos *observer> ask turtles [set color brown]*.

El mundo NetLogo es un mundo de dos dimensiones que se compone de tortugas, parches, enlaces y observadores. Los parches están sobre le fondo en el cual se mueven las tortugas. Los enlaces son conexiones entre tortugas. Y los observadores supervisan lo que está pasando. (para más detalles ver Guía de Programación de NetLogo).

En el centro de comandos puedes dar comandos para estos tipos de agentes. Puedes elegir el tipo de agente usando el pop menú localizado el botón de la esquina inferior izquierda o presionando el tabulador.

El observador supervisa el mundo y por lo tanto puede dar un comando a los parches o tortugas usando ask.

*observador> ask patches [set pcolor yellow]*

*observador> ask turtles [set color brown]*

*tortugas> set color pink*

*parcelas> set pcolor White*

Si presionamos el botón setup el modelo se reconfigurará a la configuración que está en la pestaña de Código. El centro de comandos no hace cambios permanentes en el modelo. Esto permite que tu manipules el mundo de NetLogo directamente (La lengüeta de código es explicada en el siguiente tutorial y el la guía de programación).

## Trabajando con colores

En las secciones anteriores usamos dos palabras diferentes para cambiar de color: *color* y *pcolor*.

Llamamos “variables” a color y pcolor. Algunos comandos y variables son específicos de tortugas y algunos de parches. Por ejemplo la variable color es una variable tortuga, mientras que la variable pcolor es una variable parche.

Usando el comando set y con las variables color y pcolor se pueden alterar los olores de las tortugas y parches respectivamente.

En NetLogo, los colores tienen un valor numérico. Anteriormente hemos usado el nombre del color, ya que NetLogo reconoce el nombre de 16 colores. Hay muchos matices que se pueden utilizar. A continuación se muestra una tabla que muestra todo el espacio de color de NetLogo:

![A description...](data:None;base64,)

Para elegir un color intermedio, te puedes referir a su número o sumar o restar un número al nombre. Por ejemplo: set color red, sería lo mismo que poner set color 15, para elegir un color más claro o más obscuro puedes poner ser pcolor red - 2 ó set pcolor red + 2.

## Agente Monitores y Agente de Comandos

Para cambiar el color de un solo carro:

1. Click derecho sobre el carro.
2. Seleccionar “inspect turtle”

Ahora se pueden todas las variables que pertenecen a un auto, otra manera de abrir es monitor de tortugas desde el menú de herramientas u otra es desde el centro de comandos teclear inspect turtle 15. Desde el centro de comando se pueden cambiar los parámetros de una tortuga.

Para pedir (ask) a una tortuga que haga algo usamos su número pero para dirigirnos a un parche necesitamos dos números que corresponden a sus coordenadas. Para cambiar el color del parche (0,0) desde el centro de comandos observador> ask patch 0 0 [set pcolor green], también podemos dar click derecho sobre el parche (0,0) y abrir el inspector de parches, desde el menú de herramientas o desde el centro de comandos con inspect patch 0 0.

Tutorial #4 Procedimientos

Exiten 4 tipos de agentes, los observadores supervisan todas las cosa que van a hacer las tortugas, parches y links; y lo que no pueden hacer por si mismos.

Los agentes pueden ejecutar comandos y procedimientos. Un procedimiento es la combinación de una serie de comandos de NetLogo que están dentro de un nuevo comando que tu defines. Los procedimientos hacen que las tortugas coman, se reproduscan, se muevan y mueran.

Cambios en la actualización de la vista basados en tick

Ahora que usamos el contador de ticks (con reset-ticks). Podemos decirle a NetLogo que solo necesitamos que actualice la vista uan vez por cada tick, en lugar de hacerlo continuamente. En el menú de opciones solo debemos selecionar “manualmente (por ticks)”. Esto hace que el modelo funcione más rápidamente y garantiza una apariencia consistente (ya que los cambios se deránen tiempos constantes). Vea la guia de progrmación para una discución más completa de actualización de vistas.

Haciendo un botón go

En la ventana de configuración de botón hay que activar la opción “desabilitar hasta que los ticks comiencen”, impide que se pueda presionar go antes de que se haya cargado el modelo “setup”.

También hay que seleccionar la opción “continuamente/forever” que hace que el botón una vez presionado permanezca así, para que su procedimiento se ejecute una y otra vez.

Nota: si una tortuga se mueve afuera del borde del mundo “se enrrolla”, es decir que aparece en otro lado, etse comportamiento es predeterminado, pero se puede cambiar. Consulte la sección de topología en la guia de progrmación para obtener más información.

Puedes hacer que las tortugas pinten su trayectoria con turtle>pen-down en el centro de comandos.