En el presente documento se explican los elementos teóricos de la simulación basada en agentes, los agentes inteligentes, los modelos matemáticos de propagación de pandemias, entre los que desataca el SEIR, la arquitectura otros aspectos que posibilitan el entendimiento del trabajo. Se exponen por separado los distintos elementos que conforman el objeto de estudio del trabajo.

# Agentes y la simulación

## Agentes inteligentes

Un agente es un ente capaz de percibir su medio ambiente con la ayuda de sensores y actuar en ese medio utilizando actuadores. Un agente humano tiene ojos, oídos y otros órganos sensoriales además de manos, piernas, boca y otras partes del cuerpo para actuar. Un agente robot recibe pulsaciones del teclado, archivos de información y paquetes vía red a modo de entradas sensoriales y actúa sobre el medio con mensajes en el monitor, escribiendo ficheros y enviando paquetes por la red. Un agente se considera inteligente cuando es capaz de percibir su entorno, procesar tales percepciones y responder o actuar en su entorno de manera racional, es decir, de manera correcta y tendiendo a maximizar un resultado esperado. La Figura 1. Estructura básica de un agente.

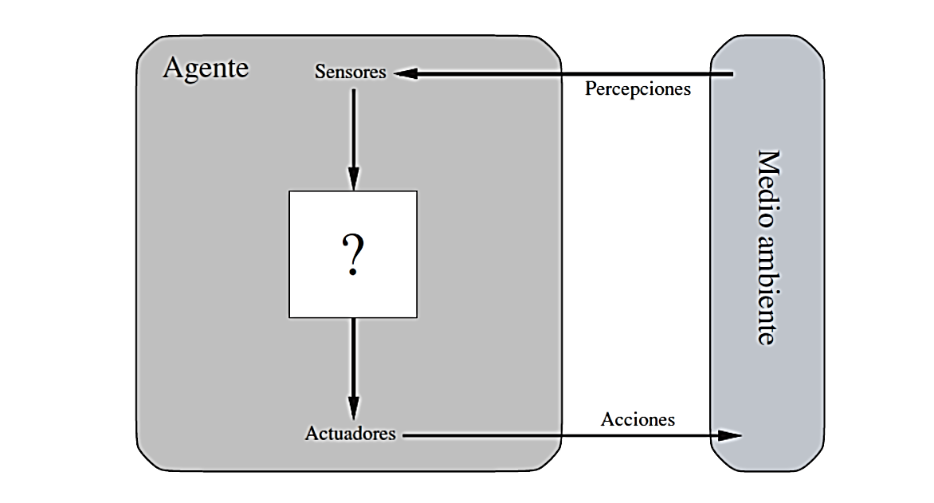


Figura 1. Estructura básica de un agente

De acuerdo a lo expresado en el libro *"Intelligent agents: theory and practice” se* puede definir como agente inteligente un sistema computacional que reúna las siguientes propiedades:

1. **Autonomía**

Los agentes operan sin la intervención directa de un ser humano y poseen determinado control sobre sus acciones y estados internos.

1. **Habilidad social**

Los agentes deben ser capaces de comunicarse con otros agentes.

1. **Proactividad**

Los agentes no solo actúan en respuesta al medio ambiente, tienen un comportamiento orientado a cumplir determinado objetivo.

1. **Reactividad**

Los agentes deben percibir su entorno y ser capaces de actuar para modificar el mismo.

## Sistemas Multi-Agente

Un Sistema Multi-Agente (*Multi-Agent System,* MAS) consiste en un conjunto de agentes autónomos y sociales que interactúan entre sí en determinado sistema. Estos agentes se comunican no solo mediante el intercambio de datos, sino a través de analogías de actividad social real como cooperación, negociación y coordinación.

Los agentes de un MAS interactúan entre sí para realizar determinadas acciones o lograr el mejor resultado posible de su objetivo individual. Es esa flexibilidad la que hace a los MAS adecuados para modelar y resolver problemas de varias disciplinas como lo son la ciencia de computación y la ingeniería.

## Arquitectura BDI

La arquitectura BDI (Belief, Desire, Intention) está caracterizada por el hecho de que los agentes que la implementan están dotados de los estados mentales de Creencias, Deseos e Intenciones. El éxito de esta arquitectura se debe posiblemente a que combina elementos interesantes: un modelo filosófico del razonamiento humano fácil de comprender, un número considerable de implementaciones y una semántica lógica abstracta y elegante, que ha sido aceptada por gran parte de la comunidad científica. Las nociones de complejidad y cambio tienen un gran impacto en la forma en que se construyen los sistemas computacionales y por tanto en los agentes. Los agentes y en particular los agentes BDI incorporan componentes que permiten el desarrollo de sistemas que se integren adecuadamente en el mundo real. Muchas de las aplicaciones de sistemas informáticos son algorítmicas y trabajan con información exacta. Pero la mayoría de las aplicaciones requieren sistemas más complejos, sistemas capaces de relacionarse con un entorno cambiante y con un cierto grado de incertidumbre. Los agentes y los sistemas multiagentes tienen por tanto que ser capaces de proporcionar soluciones a este tipo de problemas. El modelo BDI se ha desarrollado para proporcionar soluciones en entornos dinámicos o inciertos, en los que el agente o los agentes sólo tienen una visión parcial del problema (el acceso a la información está limitado) y posiblemente manejen un número limitado de recursos (recursos informáticos finitos). Las creencias, los deseos, las intenciones, y los planes son una parte fundamental del estado de ese tipo de sistemas.

Las nociones de complejidad y cambio tienen un gran impacto en la forma en que se construyen los sistemas computacionales y por tanto en los agentes. Los agentes y en particular los agentes BDI incorporan componentes que permiten el desarrollo de sistemas que se integren adecuadamente en el mundo real. Muchas de las aplicaciones de sistemas informáticos son algorítmicas y trabajan con información exacta. Pero la mayoría de las aplicaciones requieren sistemas más complejos, sistemas capaces de relacionarse con un entorno cambiante y con un cierto grado de incertidumbre. Los agentes y los sistemas multiagentes tienen por tanto que ser capaces de proporcionar soluciones a este tipo de problemas. El modelo BDI se ha desarrollado para proporcionar soluciones en entornos dinámicos o inciertos, en los que el agente o los agentes sólo tienen una visión parcial del problema (el acceso a la información está limitado) y posiblemente manejen un número limitado de recursos (recursos informáticos finitos). Las creencias, los deseos, las intenciones, y los planes son una parte fundamental del estado de ese tipo de sistemas. En el campo de la inteligencia artificial, las creencias representan el conocimiento que se tiene del entorno. Desde un punto de vista informático, son la forma de representar el estado del entorno, por ejemplo, el valor de una variable, los valores de una base de datos relacional, o expresiones simbólicas del cálculo de predicados. Las creencias son esenciales en esta arquitectura. En medios dinámicos es necesario mantener información sobre los eventos pasados al mismo tiempo que se debe permitir su adaptación y evolución, por lo que una buena gestión de estos elementos es vital para construir agentes eficientes. Los deseos (objetivos) son otro componente esencial en esta arquitectura. En términos informáticos, un objetivo puede simplemente ser el valor de una variable, un registro, o una expresión simbólica en alguna lógica. Un objetivo representa un estado final deseado. El software convencional está “orientado a la tarea” en lugar de “al objetivo”, de forma que cada tarea (o subrutina) se ejecuta sin ningún recuerdo de por qué ha comenzado su ejecución. Esto significa que el sistema no puede recuperarse ante fallos automáticamente (a menos que esto sea específicamente codificado por el programador) y no puede descubrir ni aprovechar oportunidades que surjan inesperadamente. Por ejemplo, la razón de que un humano se recupere de, resuelva la pérdida de un tren o un pinchazo inesperado de la rueda de su coche, es porque conoce dónde está (a través de sus creencias) y recuerda qué quiere conseguir (a través de sus objetivos). La semántica fundamental de los objetivos, sin tener en cuenta cómo se representa computacionalmente, se reflejaría en una lógica de los deseos. Para alcanzar los objetivos propuestos, a partir de las creencias existentes es necesario definir un mecanismo de planificación que nos permita identificar las intenciones. En este sentido, tenemos que tener claro que los agentes están inmersos en sistemas dinámicos, en los que en ocasiones será necesario decidir si replanificar o no, ante cambios en el entorno, durante la ejecución del plan inicialmente seleccionado. Los adherentes a la teoría de la decisión clásica dicen que siempre se debería replanificar, mientras que los diseñadores de software convencional orientado a la tarea obligarían a continuar hasta el final con el plan inicialmente previsto. ¿Cuál es el enfoque correcto? Posiblemente ni la teoría de la decisión clásica ni la aproximación convencional son, por sí solas, apropiadas. El sistema necesita comprometerse con los planes y subobjetivos, pero también ser capaz de reconsiderar éstos en los momentos clave. Estos planes vinculados a la consecución de un objetivo constituyen las intenciones del agente. Las intenciones son simplemente un conjunto de caminos de ejecución (“hebras”, en inglés threads) que pueden ser interrumpidos de una forma apropiada al recibir información acerca de cambios en el entorno.

## Simulación basada en agentes inteligentes

La simulación basada en agentes inteligentes permite crear modelos que recrean procesos de la vida real en un entorno computacional para poder evaluar comportamientos, probar y tomar decisiones respecto a dicho proceso. La simulación basada en agentes inteligentes se apoya en técnicas de la inteligencia artificial para crear acciones dinámicas, reacciones y protocolos de intercomunicación entre los agentes en un entorno compartido, para evaluar su diseño y rendimiento y obtener información sobre su comportamiento y propiedades emergentes.

La simulación basada en agentes inteligentes tiene el potencial de complementar modelos de sistemas complejos y modelos matemáticos, que suelen ser menos realistas debido a las fuertes restricciones impuestas por las hipótesis simplificadoras que son necesarias para poder resolverlos. Usando técnicas de modelado computacional como la simulación basada en agentes y la dinámica de sistemas tenemos el potencial de construir modelos que combinan la riqueza descriptiva de los modelos verbales con el rigor formal de los modelos matemáticos más abstractos.

La simulación social basada en agentes es una técnica de simulación, en donde la principal entidad es el agente inteligente como ente dentro del paradigma de Simulación basada en agentes (ABM), por sus siglas en inglés, al utilizar exhaustivamente la proactividad y la habilidad social. Estos métodos tienen la capacidad de estimar la plausibilidad del comportamiento de los agentes, la forma en que interactúan, y los efectos de ese comportamiento e interacción.

## Modelos matemáticos de control de pandemias

Un modelo de control de pandemia es un modelo matemático que aprovecha el conocimiento cada vez mayor de los mecanismos y las estimaciones de los parámetros para explorar posibles escenarios futuros, incluidos los resultados del peor de los casos, y determinando la acción preventiva que se tomará.

Uno de los modelos más utilizados en este campo es el modelo SEIR. Los estados de este modelo incluyen a los individuos susceptibles a ser infectados (S), los que han estado expuestos al contagio con un infectado (E), pero que no presentan manifestaciones de la enfermedad, los infectados (I) que pueden contagiar a otros y los recuperados de la enfermedad (R). La Figura 2. Estructura del modelo SEIR original, muestra la estructura básica de un modelo SEIR.

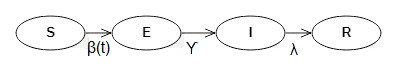
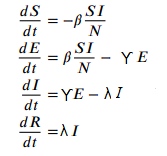


Figura 2. Estructura del modelo SEIR original [13]

Los coeficientes {β, ϒ, λ} representan la tasa de infección (probabilidad de infección), tasa de progresión (tiempo de incubación) y tasa de recuperación (probabilidad de curarse), la fórmula general del modelo SEIR se muestra en la Ecuación 1. Ecuación diferencial ordinaria del modelo SEIR.



Ecuación 1. Ecuación diferencial ordinaria del modelo SEIR [4]

Esta clase de modelos son ampliamente utilizados y se han vuelto muy populares en los últimos tiempos con la aparición de virus como los coronavirus, el ébola, entre otros, para hacer predicciones sobre el comportamiento de los mismo en grandes extensiones como ciudades o países. Por otro lado, no funcionan bien al aplicarlo en lugares donde un virus se comporta de manera muy irregular, en estos casos se debería modelar independientemente en los lugares con comportamiento irregular, por ejemplo, al inicio de la pandemia en China, se demostró su efectividad al aplicarse sobre el país completo, solo si se excluía la provincia de Wuhan, ya que sus parámetros diferían demasiado del resto y se encontraban muy por encima de la media.

### Modelos SEIR modificados

Para simular el comportamiento del virus en la población, se analizan modelos extraídos de diferentes estudios presentados en en la Tabla 1. Variables de estados en modelos matemáticos y Tabla 2. Variables de tasas en modelos matemáticos. En ambas tablas se muestran las variables presentes en estos modelos.

Tabla 1. Variables de estados en modelos matemáticos

|  |  |
| --- | --- |
| Nombre | Descripción |
| NS | No susceptible, inmunidad ante el virus |
| S | Susceptible |
| E | Expuesto, posee el virus pero no es infeccioso |
| A | Asintomático e infeccioso |
| I | Sintomático e infeccioso |
| Q | En cuarentena e infeccioso |
| R | Recuperado de progresión sintomática |
| R | Recuperado de progresión asintomática |
| D | Fallecido |

Tabla 2. Variables de tasas en modelos matemáticos.

|  |  |
| --- | --- |
| Nombre | Descripción |
| α | Tasa de protección |
| β(t) | Tasa de infección |
| β’(t) | Tasa de infección asintomática |
| ϒ | Tasa de progresión |
| δ | Tiempo de cuarentena |
| λ | Tasa de recuperación |
| λ’ | Tasa de recuperación asintomática |
| k | Tasa de muerte |
| ρ | Tasa de des inmunización |

1. M1: Modelo SEIR modificado que tiene en cuenta las personas inmunes a la enfermedad y en cuarentena. También, se agrega el estado de fallecido en comparación con el modelo SEIR clásico como se muestra en la Figura 3. Variante M1 del modelo SEIR.

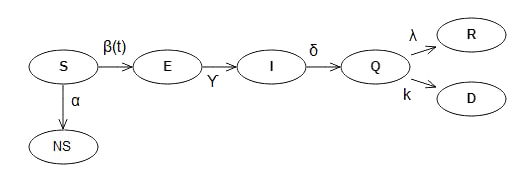


Figura 3. Variante M1 del modelo SEIR

1. M2: Modelo mostrado en la Figura 4. Variante M2 del modelo SEIR, este tiene en cuenta individuos asintomáticos infecciosos con su propia tasa de infección, se asume que estos se vuelven sintomáticos

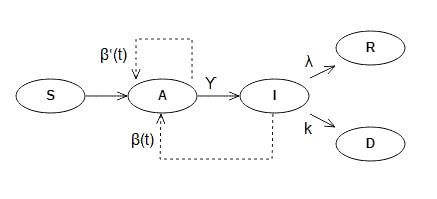


Figura 4. Variante M2 del modelo SEIR

1. M3: Similar al M2, posibilita que los individuos asintomáticos se recuperen sin volverse sintomáticos, dichos casos quedarán sin reportar. Se muestra en la Figura 5. Variante M3 del modelo SEIR.

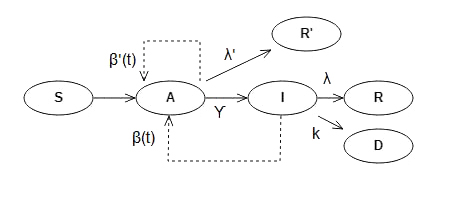


Figura 5. Variante M3 del modelo SEIR

1. M4: Similar al M2, permite que los individuos recuperados vuelvan a ser susceptibles a la enfermedad como se muestra en la Figura 6. Variante M4 del modelo SEIR.

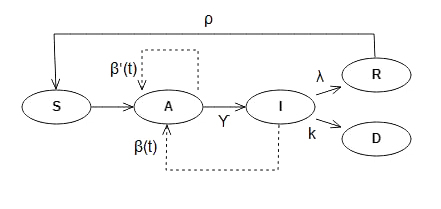


Figura 6. Variante M4 del modelo SEIR

Mientras que los modelos matemáticos brindan descripciones a nivel de población, los modelos basados en agentes se centran en formulaciones de las propiedades y dinámicas de los individuos. Investigaciones muestran que las predicciones obtenidas para varios modelos utilizando los números de casos informados pueden estar sujetas a una incertidumbre sustancial. Más importante aún, las estimaciones a menudo no eran realistas y los intervalos de confianza/credibilidad no cubrían los valores plausibles de los parámetros críticos obtenidos utilizando diferentes enfoques.

## Simulación basada en agentes usando modelos SEIR

En el área de la simulación basada en agentes, muchos han sido los sistemas que se han desarrollado para modelar el contagio del virus y su impacto en países como Ucrania y Vietnam o con propósito más general, de manera que se adapte a distintas circunstancias. Además, se cuenta con el código fuente del kit de modelación COMOKIT, una herramienta de simulación de código abierto desarrollada en la plataforma GAMA para predecir el comportamiento del Covid-19 en Vietnam. Esta herramienta cuenta con una amplia gama de experimentos para modelar distintas situaciones como cuarentena total y parcial, además, tiene entre sus objetivos ser estudiada y aplicada en otros modelos de contagio y propagación, por lo que servirá de apoyo a este trabajo. No obstante, la herramienta no tiene en cuenta factores como el transporte público o el movimiento de las personas por una ciudad. Teniendo en cuenta lo anterior, hasta la fecha no existe un modelo de simulación basada en agentes desarrollada para adaptarse a la situación de Cuba.

## Pandemias y ciudades

En 2009 más de la mitad de la población mundial vivía en ciudades y se pronostica que para 2050 esta cifra aumente debido a factores como la migración, el desarrollo y las fuerzas económicas, incluida la globalización.

Las ciudades son centros de crecimiento económico e innovación, pero la alta densidad poblacional de las mismas las vuelve vulnerables a catástrofes como las pandemias. A principios de 2020, resultó evidente que existían diferencias significativas en términos de propagación y mortalidad en los brotes de Covid-19 en diferentes ciudades del mundo. Estas diferencias llevaron a estudios sobre factores que afectaban la propagación, como la polución ambiental y las características climáticas regionales.

En un estudio realizado durante los primeros ocho meses de la pandemia de Covid-19, se demostró que hay muchos factores que influyen en la propagación del virus en las ciudades, agrupados en: calidad medioambiental, impactos socio-económicos, gerencia y gobierno, y transporte y diseño urbano. Esto se aprecia más detalladamente en la Figura 7. Factores que afectan el comportamiento del Covid-19 en ciudades.

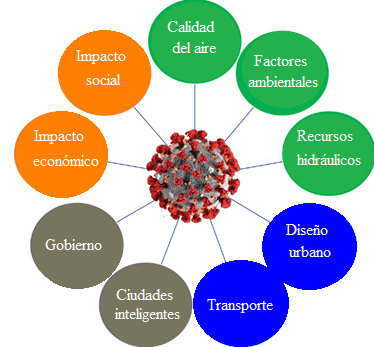


Figura 7. Factores que afectan el comportamiento del Covid-19 en ciudades [21].

En Cuba estos factores se comportan de manera diferente a otros lugares del mundo, por lo que se requiere un análisis personalizado de las características urbanísticas de Cuba en orden de desarrollar un modelo de contagio adecuado para el país.

### El caso de Cuba

En un estudio realizado desde el inicio de la pandemia hasta el 29 de mayo de 2020, se determinó que las provincias con mayor tasa de incidencia fueron: La Habana, Villa Clara, Matanzas y Ciego de Ávila, en ese orden. Ante la desigualdad de este comportamiento se indagó sobre la posible relación de estas tasas con algunos indicadores demográficos como el grado de urbanización, la densidad poblacional y el grado de envejecimiento, así como con la edad y el sexo de las personas contagiadas.

**Urbanización**

El grado de urbanización entre las provincias del país es bastante parecido, y aunque puede influir en el comportamiento del virus como se muestra en la Figura 8. Relación entre porciento de urbanización e incidencia de Covid-19 en Cuba por cada 100 mil habitantes, no permite establecer una relación directamente proporcional entre estos dos indicadores. Por lo que resalta la idea de la interrelación de diversas variables de diferentes naturalezas en la incidencia de esta enfermedad epidémica, debido a sus características y las formas de afrontamiento por parte de los gobiernos nacionales y locales.

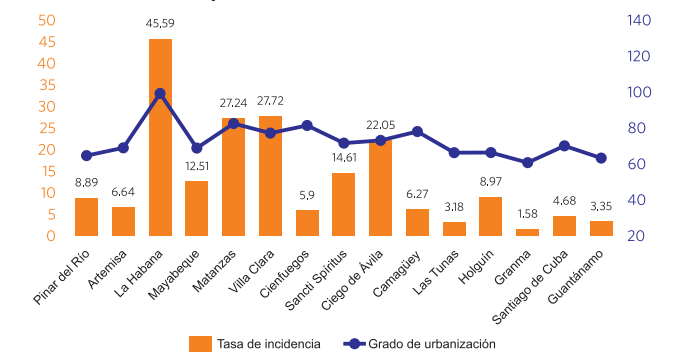


Figura 8. Relación entre porciento de urbanización e incidencia de Covid-19 en Cuba por cada 100 mil habitantes

**Edad y Sexo**

Cuba presenta un envejecimiento poblacional marcado de más de 18%, factor que vuelve a la población más vulnerable a contraer la enfermedad, como se observa en la Figura 9. Tasa de incidencia de Covid-19 por sexo y grupo de edad en provincias de Cuba. Asimismo, se determinó que las diferencias biológicas entre hombres y mujeres, y cómo responden sus organismos a las enfermedades respiratorias, afecta a la incidencia y mortalidad del virus, dejando a la población masculina más susceptible a contraer la enfermedad, como se muestra en la Figura 9. Tasa de incidencia de Covid-19 por sexo y grupo de edad en provincias de Cuba.

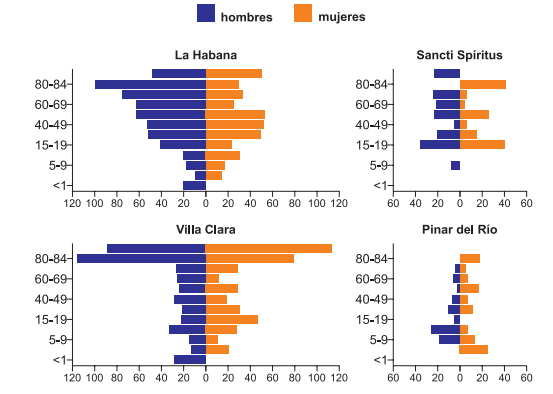
****

Figura 9. Tasa de incidencia de Covid-19 por sexo y grupo de edad en provincias de Cuba

Es de gran importancia para el desarrollo de este trabajo, tener en cuenta los factores anteriormente descritos, y otros a determinar, que afectan directamente a la forma de contagio y el comportamiento del virus en Cuba.

## Sistemas de Información Geográfica GIS

Los sistemas de información geográfica (GIS) están ligados fundamentalmente al almacenamiento de datos geográficos que permiten la organización, almacenamiento, manipulación, análisis de datos procedentes del mundo real. Es una tecnología útil para resolver problemas que permite crear una representación generalizada del mundo con medidas y datos reales.

La información geográfica se puede dividir y estructurar en diferentes capas que a su vez contienen información de sus elementos, como se muestra en la Figura 10. Estructuración en capas GIS. Cada capa posee una escala que le permite ensamblarse con el resto formando un sistema completo con proporciones y distancias reales.



Figura 10. Estructuración en capas GIS

Es evidente la necesidad de la incorporación del sistema GIS para el desarrollo del proyecto ya que va ligada con el objetivo del mismo, siendo imprescindible el uso de información geográfica para realizar un modelo de simulación en la geografía cubana.

## Herramientas para soporte GIS

Para modelar el comportamiento de personas y la propagación y contagio de un virus en un entorno parecido a la realidad cubana, es necesario incorporar elementos GIS que permitan recrear movimiento, distancias y dimensiones, así como información estructurada en capas sobre la red vial y edificaciones. Luego de analizar la comparación de herramientas realizada en la Tabla 3. Herramientas para el manejo GIS, se decidió utilizar la herramienta QGIS ya que posee los requerimientos anteriores y maneja archivos de tipo *Shapefile*, los cuáles pueden ser usados por los entornos de simulación basada en agentes.

Tabla 3. Herramientas para el manejo GIS

|  |  |  |  |
| --- | --- | --- | --- |
| Herramienta | Pago | Exportación de shapefiles | Fácil de usar |
| QGIS | Gratis | Sí | Sí |
| BatchGeo | Gratis / Versión Pro de pago | No | Sí |
| ArcGIS | Período de prueba / $100 al año | Sí | Sí |

QGIS es un sistema de información geográfica de código abierto fácil de usar, con funciones y características comunes que se utiliza para las necesidades diarias de visualización de datos GIS, para la captura de datos, para el análisis avanzado de GIS y para presentaciones en forma de mapas, atlas e informes sofisticados.

## Entornos de simulación basada en agentes

Para la selección de una herramienta de simulación basada en agentes inteligentes se proponen las siguientes:

**NetLogo**

NetLogo es un entorno de modelado multi-agente particularmente adecuado para modelar sistemas complejos que se desarrollan a lo largo del tiempo. Los modeladores pueden dar instrucciones a agentes que operan de forma independiente para explorar la conexión entre el comportamiento a nivel micro de los individuos y los patrones a nivel macro que surgen de su interacción.

**MASON**

MASON es una librería de simulación de java, diseñada para ser la base de grandes simulaciones Java personalizadas, y también para proporcionar funcionalidad más que suficiente para muchas necesidades de simulación ligeras [29].

**SimAgent**

SimAgent está diseñado principalmente para apoyar el diseño y la implementación de agentes muy complejos, cada uno compuesto de componentes interactuantes muy diferentes (como una mente humana) donde todo está incrustado en un entorno que podría ser una mezcla de objetos físicos y otros agentes de muchos tipos.

**GAMA**

GAMA es un entorno de desarrollo de simulación y modelado para construir simulaciones basadas en agentes usada principalmente para simulaciones urbanas, de transporte y epidemiológicas.

En la Tabla 4. Comparación entre herramientas de simulación, se realiza una comparación entre las herramientas anteriores teniendo en cuenta factores fundamentales como la integración con los Sistemas de Información Geográfica (GIS) y la distribución libre de la herramienta.

Tabla 4. Comparación entre herramientas de simulación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre | SO | Distribución libre | GIS | Visualización 2D |
| NetLogo | Mac OS/ Windows/ Linux | Sí | Sí | Sí |
| Mason | Mac OS/ Windows/ Linux | Sí | Sí | Sí |
| SimAgent | Linux/Windows | Sí | No | Sí |
| GAMA | Mac OS/ Windows/ Linux | Sí | Sí | Sí |

## Entorno candidato

Se recomienda la selección de GAMA como plataforma de desarrollo por las facilidades que brinda para la modelación de contagios, además de cumplir con todos los requerimientos. Esta plataforma ha sido utilizada para la realización de herramientas de simulación basada en agentes de propagación de virus en ciudades utilizando modelos SEIR. Además, impone el reto del aprendizaje del lenguaje GAML, basado en JAVA y cuenta con una extensa documentación.

### Descripción de la arquitectura GAMA

GAMA cuenta con varios paquetes de clases que conforman la estructura principal de la plataforma de simulación. En la Figura 11. Arquitectura interna de GAMA, se pueden observar los paquetes de clases fundamentales de GAMA y de los cuales se harán uso para generar los modelos de simulación deseados.

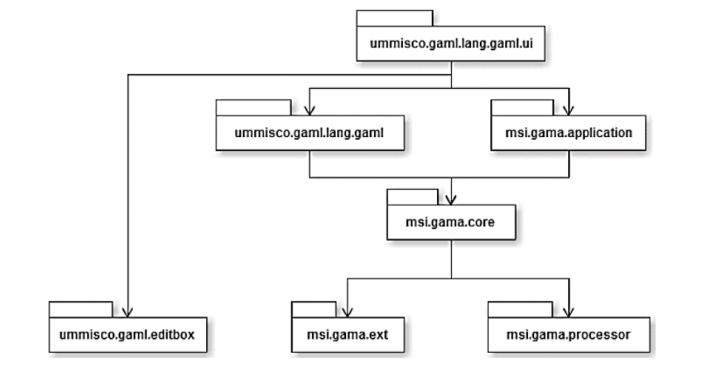


Figura 11. Arquitectura interna de GAMA

La plataforma brinda tres elementos fundamentales que deberán ser utilizados por los usuarios para desarrollar cualquier simulación. Los elementos y su relación se muestran en la Figura 12. Meta-Modelo de Gama. A continuación, se describe cada uno:

* ***<agent>***: El equivalente a un objeto en JAVA, son entidades independientes que poseen un comportamiento determinado. Son capaces de comunicarse entre ellos y percibir el mundo, afectando el comportamiento de otros y realizando acciones para cumplir su objetivo.
* ***<model>***: Un modelo contiene todas las definiciones de especies de agentes, así como las funciones y características del mundo. Se encarga de inicializar el mundo, equivalente a la clase *main* en JAVA, que a su vez inicializa a todos los agentes que participan en la simulación.
* ***<experiment>***: Ejecuta el <***model>*** principal y por tanto la simulación, genera los elementos visuales de la simulación y permite la configuración de parámetros que afectan el funcionamiento de la misma. Da la posibilidad de detener la simulación en cualquier momento.

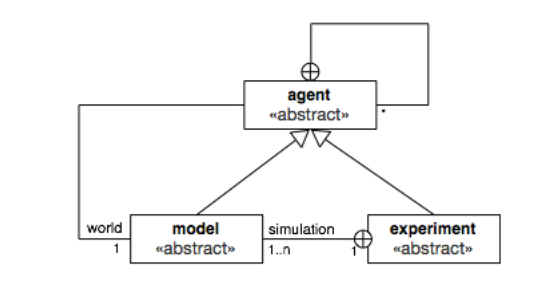


Figura 12. Meta-Modelo de Gama

## Conclusiones

En este capítulo se realizó un análisis de la simulación basada en agentes inteligentes para predecir la propagación de un virus con sistemas multi-agente utilizando modelos SEIR y arquitectura BDI. A partir de la revisión realizada se puede arribar a las siguientes conclusiones:

* Existen muchos parámetros que pueden afectar el comportamiento de un virus entre los que destacan los recogidos por los modelos de control de pandemias.
* La capacidad de interacción de los agentes inteligentes en un sistema de simulación multi-agente permite modelar el contagio de un virus.
* Los trabajos existentes desarrollados en la plataforma no tienen la implementación de una arquitectura BDI para mejorar el comportamiento de los agentes
* La plataforma de simulación GAMA es una herramienta adecuada para desarrollar un modelo de simulación utilizando modelos SEIR.