



UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE CIENCIAS EXACTAS Y NATURALES
DEPARTAMENTO DE COMPUTACIÓN
Año 2017 - 2^{do} Cuatrimestre

SIMULACIÓN DE EVENTOS DISCRETOS

TRABAJO PRÁCTICO N° <1>
TEMA:<Manejo del Inventario de una Industria>
FECHA:<27 de septiembre de 2017>

INTEGRANTES:

CERVETTO, Marcos	- #FIUBA
<cervettomarcos@gmail.com>	
MARCHI, Marcos	- #FIUBA
<edgardo.marchi@gmail.com>	
PECKER MARCOSIG, Ezequiel	- #FIUBA
<ezepecker@gmail.com>	

Índice

1. Objetivo y Enunciado	2
2. Modelo Conceptual	2
2.1. Motivación	2
2.2. Descripción funcional	3
2.2.1. Demanda (cliente)	3
2.2.2. Atención al cliente y control de calidad	5
2.2.3. Control de inventario	6
2.2.4. Inventario	7
2.2.5. Proveedores	8
3. Descripción Formal	10
3.1. Top Model	10
3.2. Tipo de Cliente A	12
3.3. Tipo de Cliente B	13
3.4. Tipo de Cliente C	14
3.5. Control de inventario	16
3.6. Control de Calidad	18
3.7. Atención al cliente	21
3.8. Inventario	22
3.9. Proveedor Inmediato	24
3.10. Proveedor por múltiplo de cantidad fija	26
3.11. Proveedor por Encargo	28
4. Modelado y simulación	30
4.1. Pruebas parciales	30
4.1.1. Cliente A	30
4.1.2. Control de Inventario	32
4.1.3. Control de Calidad	32
4.1.4. Atención al cliente	33
4.2. Pruebas de integración	34
4.3. Reporte y conclusiones	35
A. Código Implementado	35

1. Objetivo y Enunciado

El objetivo de este Trabajo Práctico es demostrar la comprensión del modelado y simulación utilizando el formalismo DEVS, junto a la aplicación de las técnicas mediante la herramienta CD++. Se deberá identificar un sistema del mundo real que pueda ser representado usando DEVS y construir luego un modelo del mismo. Finalmente se ejecutarán simulaciones del sistema analizado. El sistema estudiado puede ser natural o artificial, y puede existir en la realidad o no.

2. Modelo Conceptual

2.1. Motivación

Una compañía que vende un único producto está interesada en estimar cuántas unidades del mismo debería tener en inventario en los próximos n meses (n es un parámetro de entrada fijo).

La compañía cuenta básicamente con tres proveedores de diferentes materias primas. Cada producto se fabrica usando 1 unidad de cada materia prima.

La planta está compuesta por un departamento de control de inventario, un departamento de atención al cliente y control de calidad de producto (ya que los productos son perecederos), un stock de productos finalizados y tres stocks de materias primas (uno para cada material). Las demandas de la clientela se producen aleatoriamente en pedidos de 1 a 4 unidades (también aleatorios). Existen 3 tipos diferentes de demanda.

En la Figura 1 se propone un esquema del proceso. Se puede observar que el top-model consiste de tres jerarquías. En la jerarquía top se encuentran: los tres proveedores, los tres clientes y la industria en sí. En la siguiente se encuentran el inventario, el departamento de atención al cliente y el departamento de control de inventario. Finalmente, en el último nivel (dentro de Atención al Cliente) se encuentran la ventanilla de atención al cliente y el departamento de Control de Calidad del producto. Cada cliente consiste de un modelo atómico distinto para representar tres comportamientos distintos. De la misma manera se utilizan tres modelos atómicos para modelizar cada tipo de proveedor. Finalmente, para los departamentos de control de calidad, ventanilla de atención al cliente y control de inventario se utilizarán tres modelos atómicos.

La pregunta a responder mediante simulaciones es: *¿Qué política de reposición nos llevará a un manejo óptimo del nivel de stock logrando la mayor satisfacción del cliente y el menor costo de mantenimiento y reposición del inventario?*

Una pregunta adicional que surge a partir de la anterior es: *¿Cómo ponderar adecuadamente ambos requisitos en conflicto?*

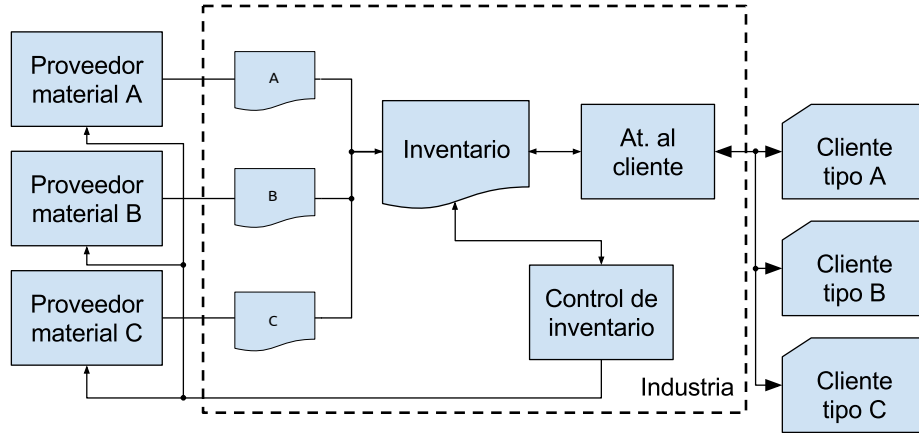


Figura 1: Esquema del problema planteado.

2.2. Descripción funcional

A continuación se describen cada uno de los bloques de la Figura 1.

2.2.1. Demanda (cliente)

Los tiempos entre demandas por tipo de cliente son variables aleatorias exponenciales e independientemente distribuidas con una media de 0,12 mes. Los tamaños de las demandas (que pueden ir de uno a cuatro unidades del producto), D , son variables aleatorias independientemente distribuidas e independientes, cuya probabilidad de ocurrencia se muestra en la Tabla 1.

D (# Unidades)	Probabilidad
1 unidad	1/6
2 unidades	1/3
3 unidades	1/3
4 unidades	1/6

Tabla 1: Demanda.

Existen 3 tipos de políticas de demanda: A, B y C. Un cliente tipo A encarga siempre n productos y los retira a principio de mes. Un cliente tipo B pide n productos pero se lleva solamente los que haya disponibles en stock en ese momento y los restantes para completar su pedido quedan encargados. Finalmente un cliente tipo C pide n productos pero solamente compra si están disponibles, sino rechaza la compra.

El modelo que representa el comportamiento de los clientes tipo A se puede observar en la Figura 2, donde la salida E_OUT indica la cantidad de productos encargados.

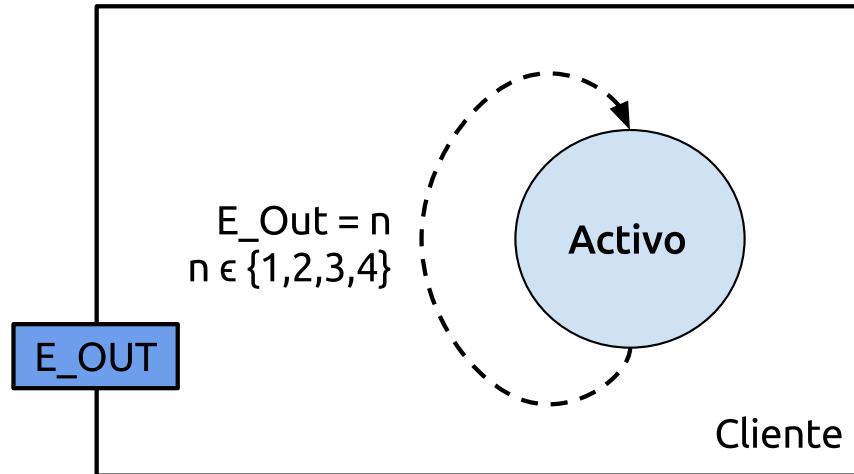


Figura 2: Modelo de cliente tipo A.

El modelo utilizado para el comportamiento de los clientes de tipo B se muestra en la Figura 3. Las salidas P_OUT, E_OUT y Q_OUT indican respectivamente la cantidad de productos retirados, la cantidad de productos encargados y la cantidad de productos pedidos.

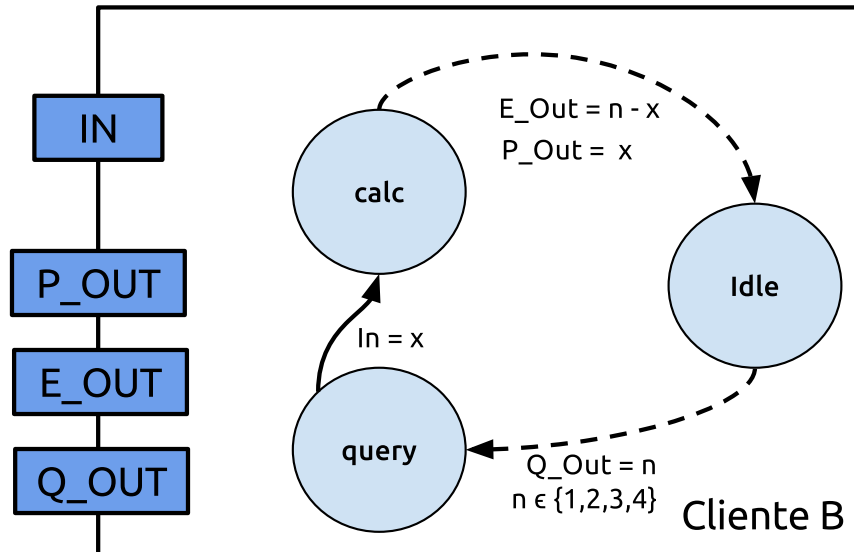


Figura 3: Modelo de cliente tipo B.

Finalmente, el modelo del comportamiento de los clientes de tipo C se muestra en la Figura 4.

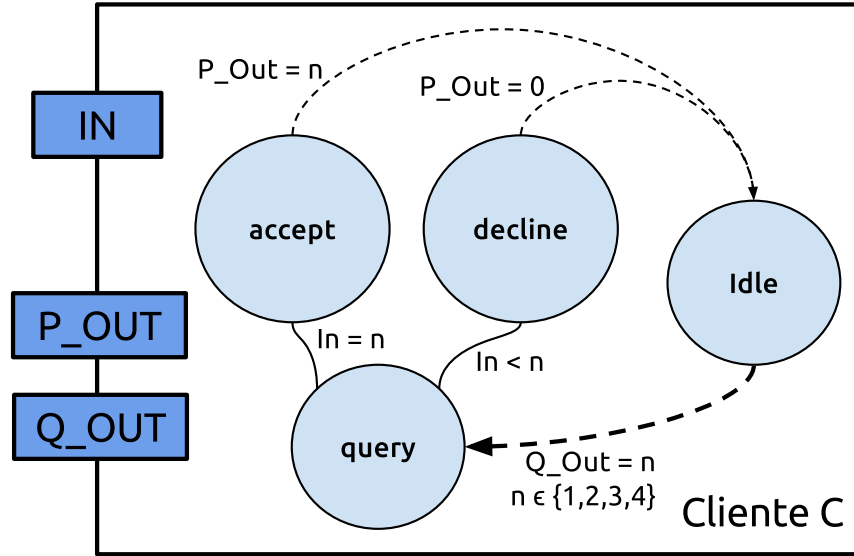


Figura 4: Modelo de cliente tipo C.

2.2.2. Atención al cliente y control de calidad

En la Figura 5 se puede observar la interfaz del módulo de Atención al Cliente y Control de Calidad. El departamento de Control de Calidad se encarga de examinar los productos al momento de venta y verificar su fecha de vencimiento. Los insumos con los que se elabora el producto son perecederos y la fecha de vencimiento del mismo se calcula con la fecha de vencimiento mínima de los insumos con los que se lo elaboró. La compañía descubre que una unidad no sirve más solamente cuando la examina debido a una venta. Si una unidad no sirve más, se desecha y se examina la siguiente unidad existente en el inventario. El departamento de atención al cliente se encarga de realizar la negociación con los distintos tipos de clientes. Es decir, consulta al inventario la cantidad de productos disponibles al recibir una consulta del cliente y le informa dicho dato al mismo. También procesa el pedido de compra de cada cliente.

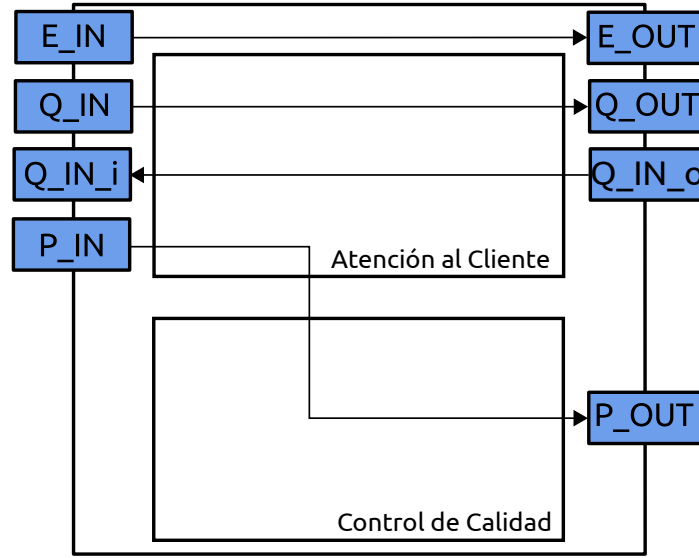


Figura 5: Atención al cliente.

2.2.3. Control de inventario

Al comienzo de cada mes, la compañía revisa el nivel de inventario y decide cuántas nuevas unidades pedir a sus proveedores. La compañía ordena Z unidades a todos los proveedores por igual. El costo costo será $K + pZ$, donde $K = 350$ pesos es el costo fijo del pedido y $p = 25$ pesos es el costo incremental por unidad. Naturalmente, si $Z = 0$ no se incurre en ningún costo. La compañía usa la siguiente política para decidir cuántas unidades pedir: si I es el nivel de inventario al comienzo del mes, S el inventario máximo posible (capacidad de almacenamiento) y s el parámetro de política de reposición, Z estará dado por:

$$Z = \begin{cases} S-I & \text{si } I < s \\ 0 & \text{si } I \geq s \end{cases} \quad (1)$$

El modelo está dado por la Figura 6, donde se observa que hay tres estados: Query, Calc y Wait.

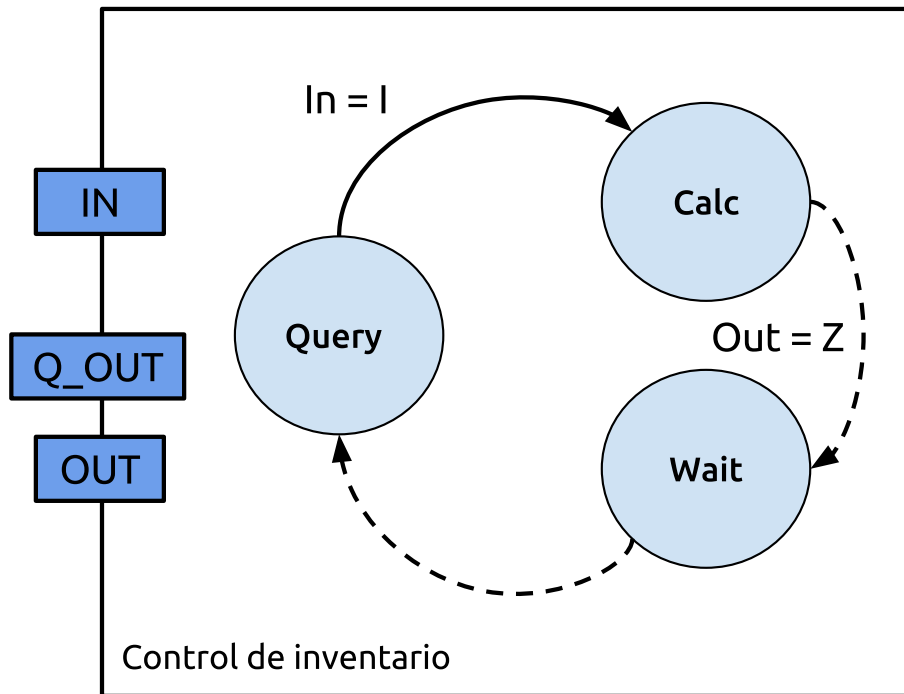


Figura 6: Esquema del control de inventario.

2.2.4. Inventario

Ante una consulta de un cliente el inventario responde con la cantidad de unidades disponibles en ese momento. En base a esto y a la política de cada cliente, al inventario se le pide una cantidad determinada de unidades para retiro y/o encargo. El inventario lleva una lista de unidades disponibles y una de unidades encargadas. Cuando una reposición llega, se usa primero para eliminar todos los pedidos pendientes (si hay) posibles; el resto de la reposición (si queda) se suma al inventario.

En la Figura 7 se observa el modelo con que se representa al Inventario. El mismo consiste de una cola FIFO (*first in - first out*). Se denomina N al largo de la cola. Suponemos también que la compañía incurre en un costo de mantenimiento del producto $h = 8$ pesos por unidad por mes mantenido en inventario (positivo). Este costo incluye alquiler del depósito de almacenamiento, seguro, impuestos y mantenimiento, al igual que costo de oportunidad de tener capital inmovilizado en inventario en vez de invertido en otra parte. Supongamos además que hay un costo de *demanda insatisfecha* $n = 50$ pesos por unidad pedida por cliente y no suministrada y por mes. Este costo contabiliza tanto el costo de mantener registros adicionales para unidades ya asignadas a clientes y aún no suministradas como un cálculo de costo por pérdida de la buena voluntad del cliente.

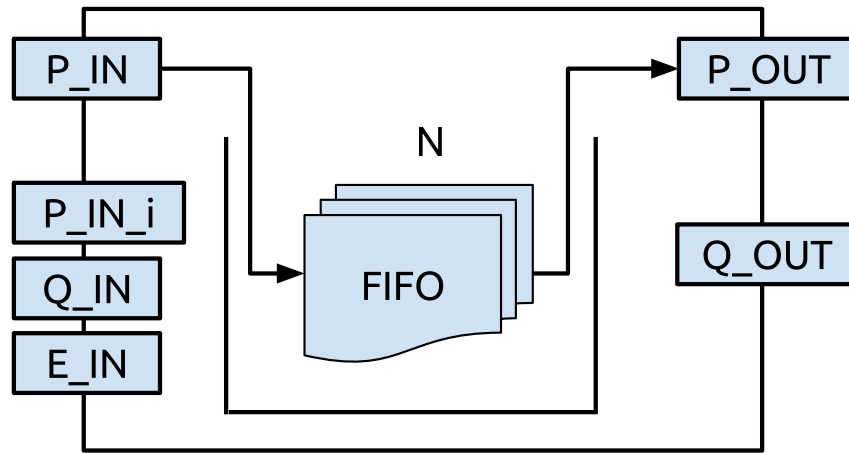


Figura 7: Inventario.

2.2.5. Proveedores

Los proveedores se comportan de 3 formas diferentes:

- i) inmediato.
 - ii) por múltiplo de cantidad fija.
 - iii) por encargo.
- i) Al hacer un pedido al proveedor inmediato de n unidades, entrega las n unidades en forma instantánea. Las n unidades asociadas tienen un tiempo de vencimiento asociado (Figura 8).
 - ii) El proveedor que funciona por múltiplo de cantidad fija, sólo entrega pedidos múltiplos de un número entero prefijado, digamos r (Figura 9). Es decir, que al hacer un pedido de n productos, pueden pasar 2 cosas:
 - 1) El número pedido n equivale a un múltiplo entero de r , $n = kr$, para algún $k \geq 0$, en cuyo caso se entregan exactamente n unidades.
 - 2) El número pedido n es mayor a un múltiplo entero de r , es decir, $n > kr$ y además $n < (k + 1)r$, para algún $k \geq 0$, en cuyo caso se entregan $(k + 1)r$ unidades.

Los productos de este proveedor no tienen vencimiento.

- iii) El tercer proveedor (Figura 10) funciona por encargo para el mes siguiente. Es decir, todo pedido será servido en el siguiente mes. En ese momento, además, se realizará el nuevo pedido.

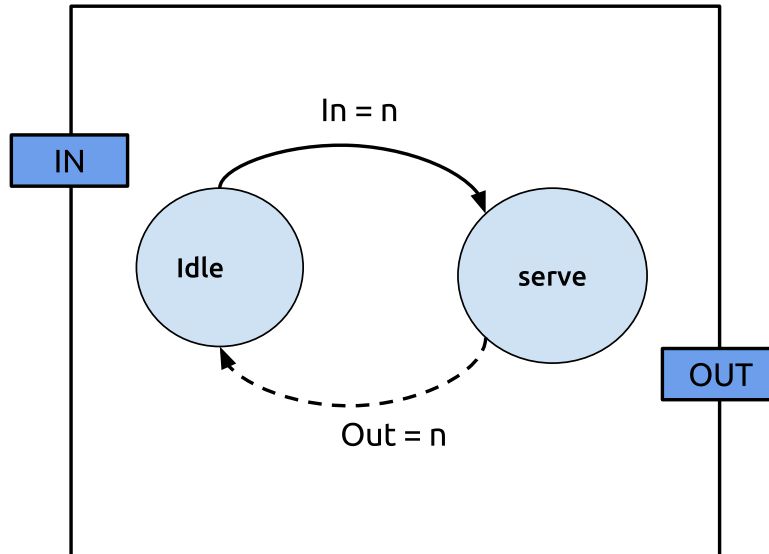


Figura 8: Proveedor 1.

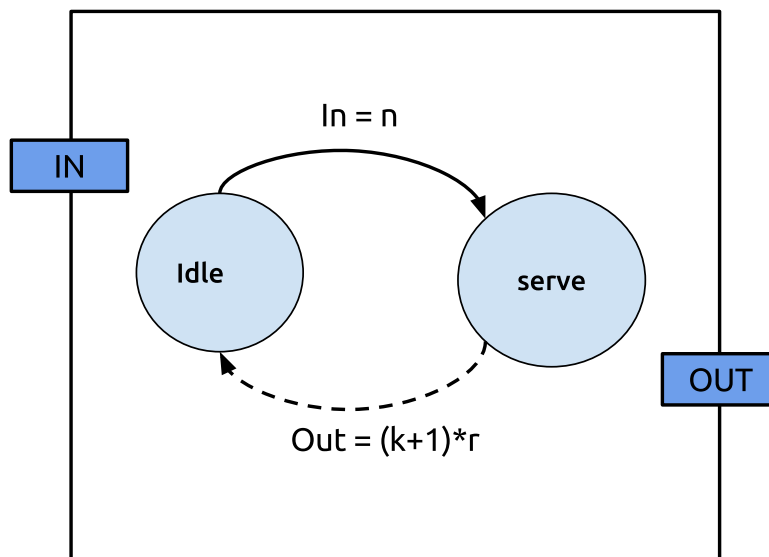


Figura 9: Proveedor 2.

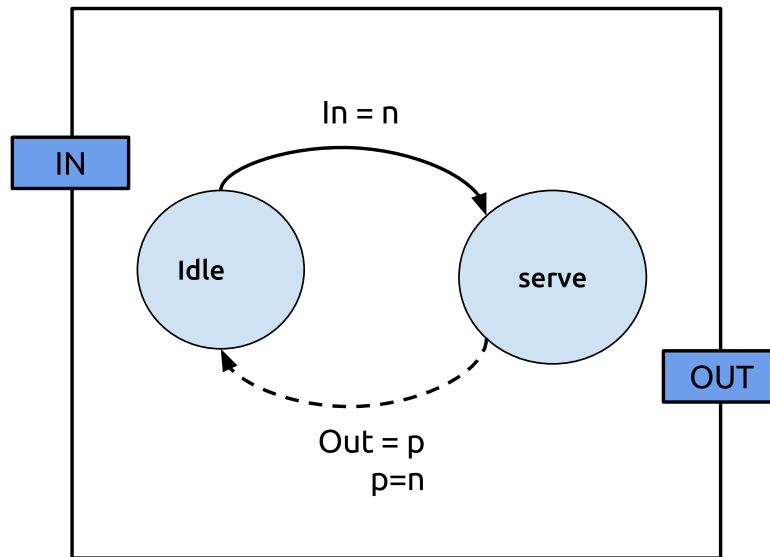


Figura 10: Proveedor 3.

3. Descripción Formal

3.1. Top Model

El diagrama de bloques correspondiente a cada uno de los atómicos descritos en la especificación funcional se puede ver en la figura 11.

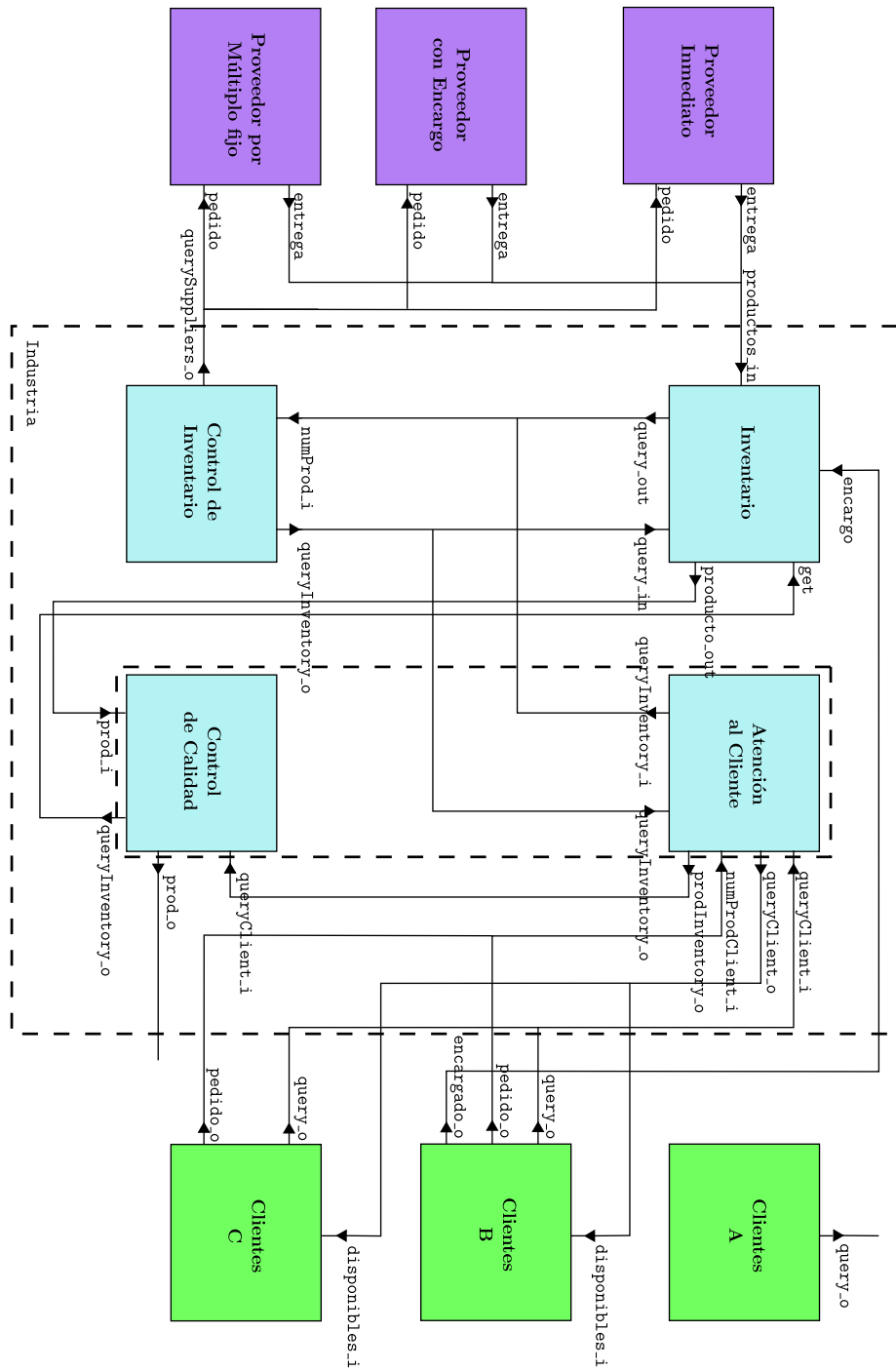


Figura 11: Interconexión de bloques en el top model.

3.2. Tipo de Cliente A

El modelo atómico correspondiente al bloque del cliente tipo A (CTA) queda definido de la siguiente manera:

$$\begin{aligned}
 CTA &= \langle S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_A \rangle \\
 S &= \{QUERY\} \\
 X &= \{\emptyset\} \\
 Y &= \{query_o\} \in \mathbb{N}_0^+ \\
 \delta_{int} &: \delta_{int}(QUERY) = QUERY \\
 \delta_{ext} &: \emptyset \\
 \lambda &: \lambda(QUERY) = \{query_o = N\} \\
 t_A(QUERY) &= EXP_VAR
 \end{aligned}$$

En las figuras 12 y 13 se pueden observar los puertos de entrada/salida y el diagrama de estados respectivamente.

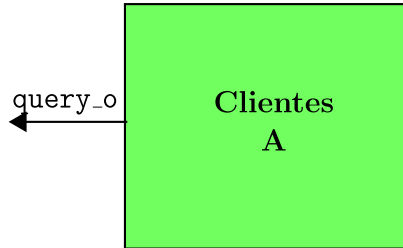


Figura 12: Entradas y salidas del modelo del cliente tipo A.

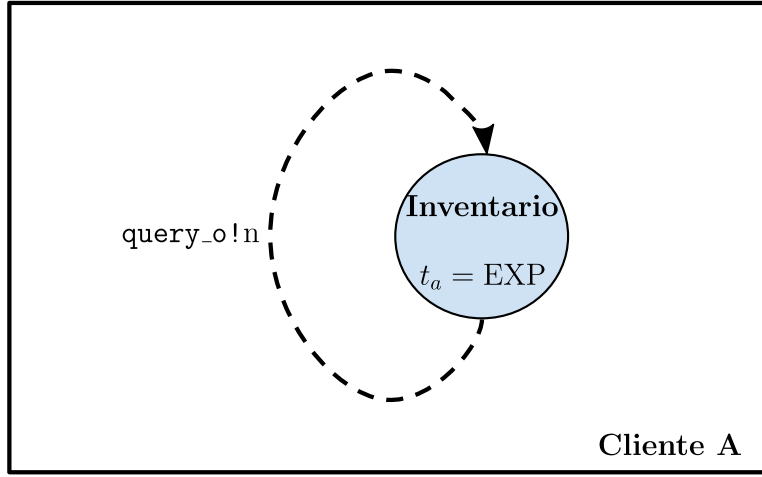


Figura 13: Diagrama de estados para el modelo del cliente tipo A.

3.3. Tipo de Cliente B

El modelo atómico correspondiente al bloque del cliente tipo B (CTB) queda definido de la siguiente manera:

$$\begin{aligned}
 CTB &= \langle S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_A \rangle \\
 S &= \{IDLE, QUERY, CALC\} \\
 X &= \{disponibles_i\} \in \mathbb{N}_0^+ \\
 Y &= \{query_o, pedido_o\} \in \mathbb{N}_0^+ \\
 \delta_{int} : \delta_{int}(S) &\begin{cases} \delta_{int}(IDLE) = QUERY \\ \delta_{int}(CALC) = IDLE \end{cases} \\
 \delta_{ext} : \delta_{ext}(S, X) &\begin{cases} Si \ S = QUERY \wedge disponibles_i = P \\ \rightarrow \delta_{ext}(QUERY, disponibles_i) = CALC \end{cases} \\
 \lambda : \lambda(S) &\begin{cases} \lambda(IDLE) = \{query_o = N\} \\ \lambda(CALC) = \{pedido_o = P\} \end{cases} \\
 t_A(S) &= \begin{cases} 0 \ si \ S = \{CALC\} \\ \infty \ si \ S = \{QUERY\} \\ EXP_VAR \ si \ S = \{IDLE\} \end{cases}
 \end{aligned}$$

En las figuras 14 y 15 se pueden observar los puertos de entrada/salida y el diagrama de estados respectivamente.

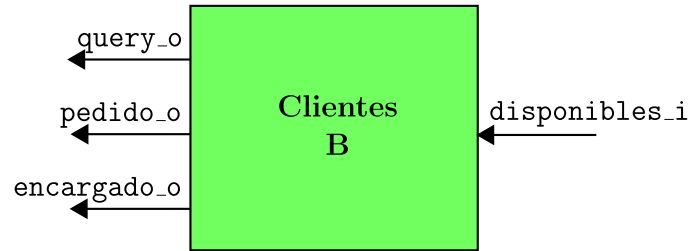


Figura 14: Entradas y salidas del modelo del cliente tipo B.

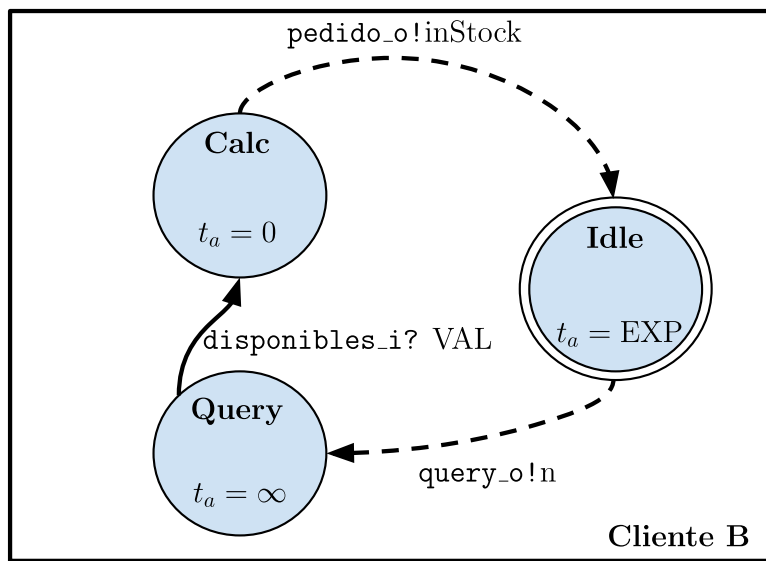


Figura 15: Diagrama de estados para el modelo del cliente tipo B.

3.4. Tipo de Cliente C

El modelo atómico correspondiente al bloque del cliente tipo C (CTC) queda definido de la siguiente manera:

$$\begin{aligned}
 CTC &= \langle S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_A \rangle \\
 S &= \{IDLE, QUERY, ACCEPT, DECLINE\} \\
 X &= \{disponibles_i\} \in \mathbb{N}_0^+ \\
 Y &= \{query_o, pedido_o\} \in \mathbb{N}_0^+ \\
 \delta_{int} : \delta_{int}(S) &\begin{cases} \delta_{int}(IDLE) = QUERY \\ \delta_{int}(DECLINE) = IDLE \\ \delta_{int}(ACCEPT) = IDLE \end{cases} \\
 \delta_{ext} : \delta_{ext}(S, X) &\begin{cases} Si\ S = QUERY \wedge disponibles_i = N \\ \quad \rightarrow \delta_{ext}(QUERY, disponibles_i) = ACCEPT \\ Si\ S = QUERY \wedge disponibles_i < N \\ \quad \rightarrow \delta_{ext}(QUERY, disponibles_i) = DECLINE \end{cases} \\
 \lambda : \lambda(S) &\begin{cases} \lambda(IDLE) = \{query_o = N\} \\ \lambda(DECLINE) = \{pedido_o = 0\} \\ \lambda(ACCEPT) = \{pedido_o = N\} \end{cases} \\
 t_A(S) &= \begin{cases} 0\ si\ S = \{ACCEPT, DECLINE\} \\ \infty\ si\ S = \{QUERY\} \\ EXP_VAR\ si\ S = \{IDLE\} \end{cases}
 \end{aligned}$$

En las figuras 16 y 17 se pueden observar los puertos de entrada/salida y el diagrama de estados respectivamente.

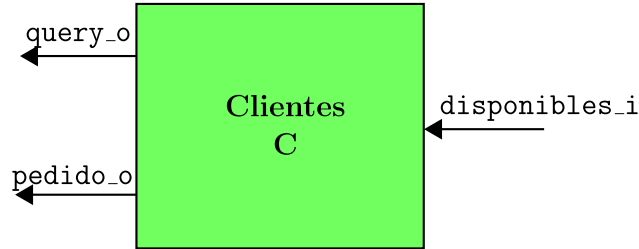


Figura 16: Entradas y salidas del modelo del cliente tipo C.

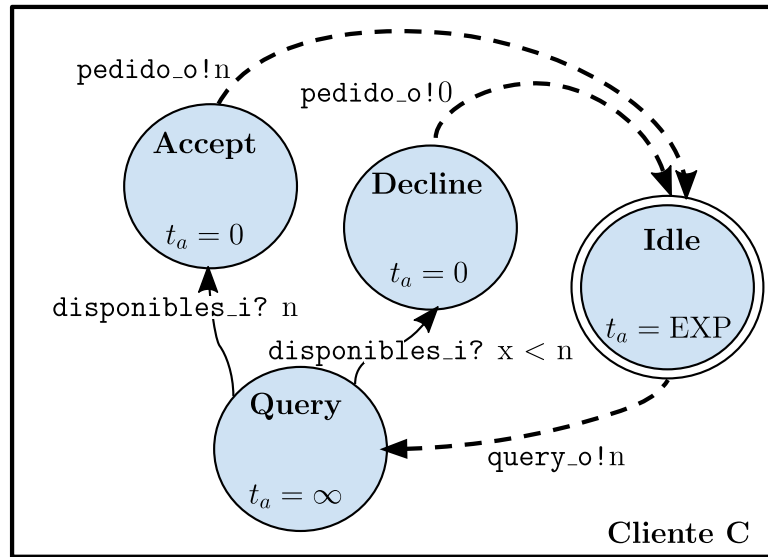


Figura 17: Diagrama de estados para el modelo del cliente tipo C.

3.5. Control de inventario

El modelo atómico correspondiente al bloque de control de inventario (CI) queda definido de la siguiente manera:

$$\begin{aligned}
 CI &= \langle S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_A \rangle \\
 S &= \{WAITING, QUERY, CALC\} \\
 X &= \{numProd_i\} \in \mathbb{N}_0^+ \\
 Y &= \{queryInventory_o, querySuppliers_o\} \in \mathbb{N}_0^+ \\
 \delta_{int} : \delta_{int}(S) &\begin{cases} \delta_{int}(WAITING) = QUERY \\ \delta_{int}(CALC) = WAITING \end{cases} \\
 \delta_{ext} : \delta_{ext}(S, numProd_i) &\begin{cases} Si S = QUERY \wedge numProd_i = N \\ \rightarrow \delta_{ext}(S, numProd_i) = CALC \end{cases} \\
 \lambda : \lambda(S) &\begin{cases} \lambda(WAITING) = \{queryInventory_o = 1\} \\ \lambda(CALC) = \{querySuppliers_o = \begin{cases} S - N & si \ numProd_i \\ 0 & \forall \ numProd_i \end{cases} < s \ \} \end{cases} \\
 t_A(S) &= \begin{cases} 1 & si \ S = WAITING \\ 0 & si \ S = CALC \\ \infty & si \ S = QUERY \end{cases}
 \end{aligned}$$

En las figuras 18 y 19 se pueden observar los puertos de entrada/salida y el diagrama de estados respectivamente.



Figura 18: Entradas y salidas del modelo control de inventario

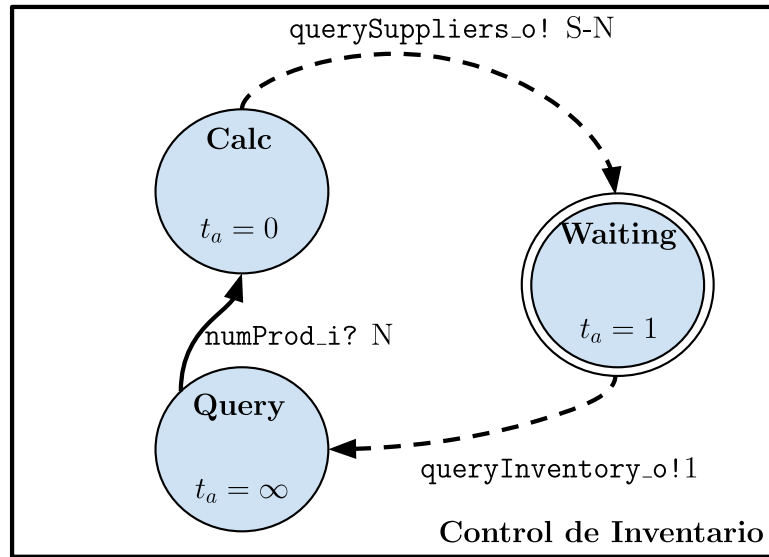


Figura 19: Diagrama de estados para el modelo de control de inventario.

3.6. Control de Calidad

El modelo atómico correspondiente al bloque de control de calidad (CC) queda definido de la siguiente manera:

$$\begin{aligned}
 CC &= \langle S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_A \rangle \\
 S &= \{WAITING, QUERY, INV_WAIT, CHECK, SEND\} \\
 X &= \{prod_i \in \mathbb{R}^+, queryClient_i \in \mathbb{N}_0^+\} \\
 Y &= \{prod_o \in \mathbb{R}^+, queryInventory_o \in \mathbb{N}_0^+\} \\
 \delta_{int} : \delta_{int}(S) &= \begin{cases} \delta_{int}(QUERY) = INV_WAIT \\ \delta_{int}(CHECK) = \begin{cases} QUERY \\ si (numPassProd < \\ numClientQuery) \wedge \\ (\neg invEmpty) \\ SEND \forall otra condicion. \end{cases} \\ \delta_{int}(SEND) = WAITING \end{cases} \\
 \delta_{ext} : \delta_{ext}(S, X) &= \begin{cases} Si S = WAITING \wedge queryClient_i = N \\ \rightarrow \delta_{ext}(QUERY, queryClient_i) = QUERY \\ Si S = CHECK \wedge size(prod_i) = P \\ \rightarrow \delta_{ext}(CHECK, prod_i) = SEND \end{cases} \\
 \lambda : \lambda(S) &= \begin{cases} \lambda(QUERY) = \{queryInventory_o = 1\} \\ \lambda(CHECK) = \{\emptyset\} \\ \lambda(SEND) = \{prod_o = K\} \end{cases} \\
 t_A(S) &= \begin{cases} 0 si S = \{QUERY, CHECK, SEND\} \\ \infty si S = \{WAITING, INV_WAIT\} \end{cases}
 \end{aligned}$$

Donde:

$numPassProd$ = es el número de productos que pasan el control de calidad (i.e. su fecha de expiración es mayor al tiempo en que se produce el control).

$numClientQuery$ = es el número de productos que solicitó el cliente.

$invEmpty$ = es un Flag que indica si el inventario está vacío (i.e. devolvió un producto con fecha de expiración 0,0)¹

En las figuras 20 y 21 se pueden observar los puertos de entrada/salida y el diagrama de estados respectivamente.

¹el símbolo \neg refiere a la condición lógica de negación

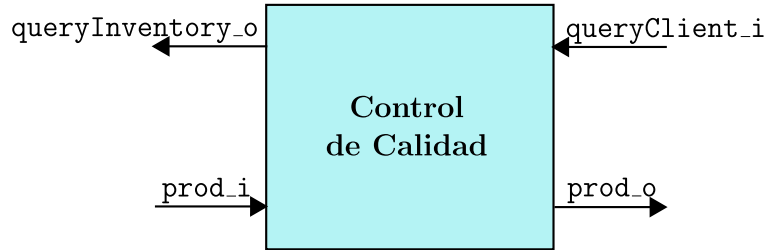


Figura 20: Entradas y salidas del modelo control de calidad.

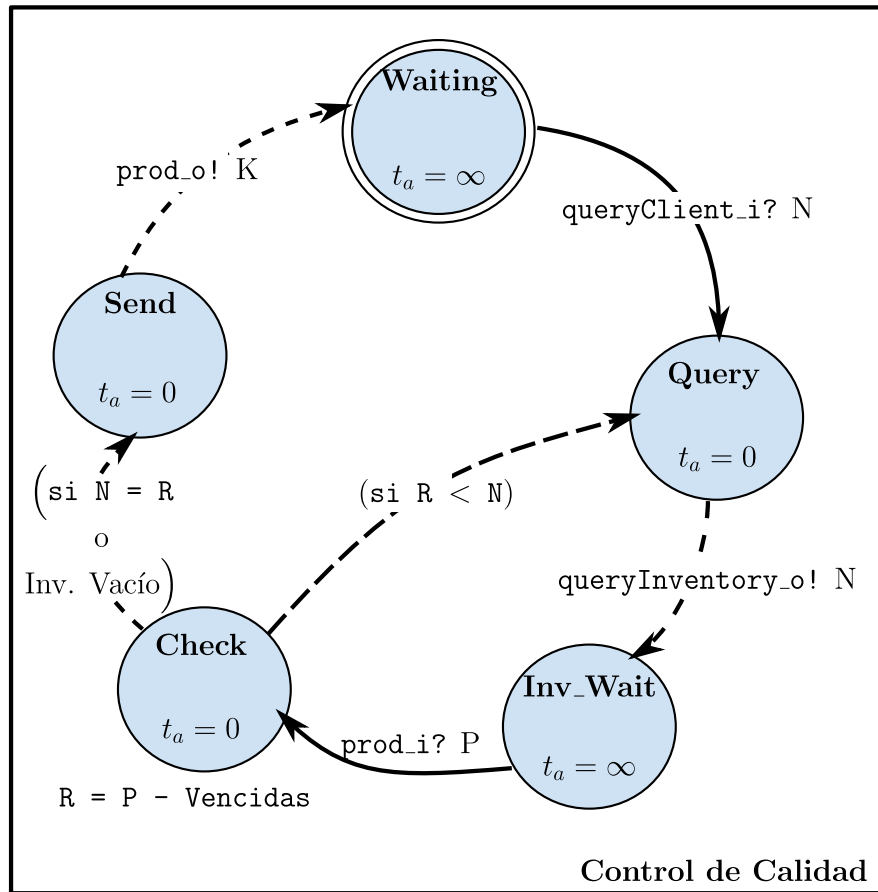


Figura 21: Diagrama de estados para el modelo de control de calidad.

3.7. Atención al cliente

El modelo atómico correspondiente al bloque de atención al cliente (AC) queda definido de la siguiente manera:

$$\begin{aligned}
 AC &= \langle S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_A \rangle \\
 S &= \{WAITING, QUERY, INV_WAIT, \\
 &\quad CLI_RPLY, CLI_WAIT, INV_GET\} \\
 X &= \{numProdClient_i, queryClient_i, queryInventory_i\} \in \mathbb{N}_0^+ \\
 Y &= \{prodInventory_o, queryInventory_o, queryClient_o\} \in \mathbb{N}_0^+ \\
 \delta_{int} : \delta_{int}(S) &\begin{cases} \delta_{int}(QUERY) = INV_WAIT \\ \delta_{int}(CLI_RPLY) = CLI_WAIT \\ \delta_{int}(INV_GET) = WAITING \end{cases} \\
 \delta_{ext} : \delta_{ext}(S, X) &\begin{cases} Si S = WAITING \wedge queryClient_i = N \\ \quad \rightarrow \delta_{ext}(WAITING, queryClient_i) = QUERY \\ Si S = INV_WAIT \wedge queryInventory_i = P \\ \quad \rightarrow \delta_{ext}(INV_WAIT, queryInventory_i) = CLI_RPLY \\ Si S = CLI_WAIT \wedge numProdClient_i = K \\ \quad \rightarrow \delta_{ext}(CLI_WAIT, numProdClient_i) = INV_GET \end{cases} \\
 \lambda : \lambda(S) &\begin{cases} \lambda(QUERY) = \{queryInventory_o = N\} \\ \lambda(CLI_RPLY) = \{queryClient_o = P\} \\ \lambda(INV_GET) = \{prodInventory_o = K\} \end{cases} \\
 t_A(S) &= \begin{cases} 0 & si S = \{QUERY, CLI_RPLY, INV_GET\} \\ \infty & si S = \{WAITING, INV_WAIT, CLI_WAIT\} \end{cases}
 \end{aligned}$$

En las figuras 22 y 23 se pueden observar los puertos de entrada/salida y el diagrama de estados respectivamente.



Figura 22: Entradas y salidas del modelo atención al cliente.

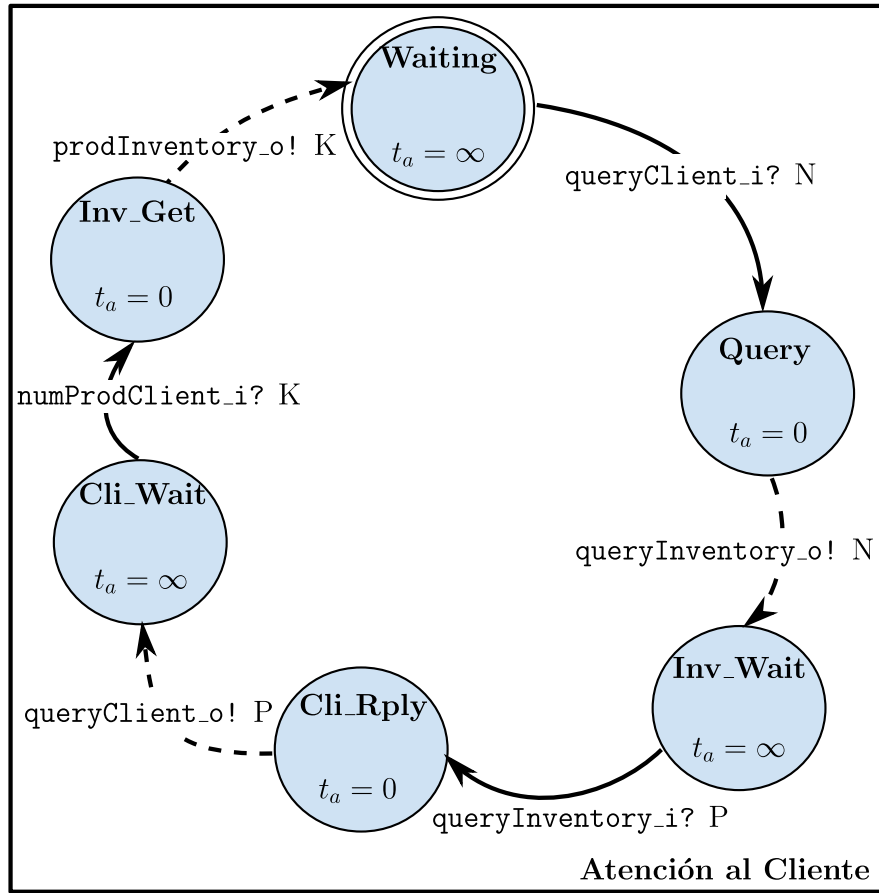


Figura 23: Diagrama de estados para el modelo de atención al cliente.

3.8. Inventario

El modelo atómico correspondiente al bloque de inventario (I) queda definido de la siguiente manera:

$$I = \langle S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_A \rangle$$

$$S = \{IDLE, QUERY, ENCARGO, PUSH, GET\}$$

$$X = \{query_in, encargo, producto_in_A, \\ producto_in_B, producto_in_C, encargo, get\} \in \mathbb{N}_0^+$$

$$Y = \{producto_out, query_out\} \in \mathbb{N}_0^+$$

$$\delta_{int} : \delta_{int}(S) \begin{cases} \delta_{int}(QUERY) = IDLE \\ \delta_{int}(ENCARGO) = IDLE \\ \delta_{int}(PUSH) = IDLE \\ \delta_{int}(GET) = IDLE \end{cases}$$

$$\delta_{ext} : \delta_{ext}(S, X) \begin{cases} Si \ S = IDLE \wedge \begin{cases} product_in_A = Na \\ product_in_B = Nb \\ product_in_C = Nc \end{cases} \\ \quad \wedge \ size(col_i) == 0 \ \exists i \in \{A, B, C\} \\ \rightarrow \delta_{ext}(IDLE, \{product_in_A, product_in_B \\ \quad product_in_C\}) = QUERY \\ Si \ S = IDLE \wedge encargo = Ne \\ \rightarrow \delta_{ext}(IDLE, encargo) = ENCARGO \\ Si \ S = IDLE \wedge \begin{cases} product_in_A = Na \\ product_in_B = Nb \\ product_in_C = Nc \end{cases} \\ \quad \wedge \ size(col_i) \neq 0 \ \forall i \in \{A, B, C\} \\ \rightarrow \delta_{ext}(IDLE, \{product_in_A, product_in_B \\ \quad product_in_C\}) = PUSH \\ Si \ S = IDLE \wedge get = Ng \\ \rightarrow \delta_{ext}(IDLE, get) = GET \end{cases}$$

$$\lambda : \lambda(S) \begin{cases} \lambda(QUERY) = \{query_out = size(col) - encargos_q\} \\ \lambda(ENCARGO) = \{\emptyset\} \\ \lambda(PUSH) = \{producto_out = \min(get_q, size(col))\} \\ \lambda(GET) = \{producto_out = \min(encargos_q, size(col))\} \end{cases}$$

$$t_A(S) = \begin{cases} 0 \ si \ S = \{QUERY, ENCARGO, PUSH, GET\} \\ \infty \ si \ S = \{IDLE\} \end{cases}$$

Donde:

cola = es la cola de productos terminados.

encargo_q = es la cantidad de productos encargados por clientes que no estaban disponibles para entrega inmediata.

get_q = es la cantidad de productos pedidos después de consultar la cantidad de productos disponibles.

La negociación entre los estados IDLE y GET es una *negociación segura*, es decir, se produce después de consultar al inventario la cantidad de productos disponibles. En caso que entre esta consulta y la compra ya no estén disponibles, porque en el medio se produjo una venta o porque alguno de los productos estaba vencido, se indica al control de calidad devolviendo un 0. Esos productos indicados con 0 no se reintegran a futuro.

En las figuras 24 y 25 se pueden observar los puertos de entrada/salida y el diagrama de estados respectivamente.



Figura 24: Entradas y salidas del modelo inventario.

3.9. Proveedor Inmediato

El modelo atómico correspondiente al bloque del proveedor inmediato (PI) queda definido de la siguiente manera:

$$\begin{aligned}
 PI &= \langle S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_A \rangle \\
 S &= \{IDLE, SERVE\} \\
 X &= \{pedido\} \in \mathbb{N}_0^+ \\
 Y &= \{entrega\} \in \mathbb{N}_0^+ \\
 \delta_{int} &: \delta_{int}(SERVE) = IDLE \\
 \delta_{ext} &: \delta_{ext}(S, X) \begin{cases} Si \ S = IDLE \wedge pedido = N \\ \rightarrow \delta_{ext}(IDLE, pedido) = SERVE \end{cases} \\
 \lambda &: \lambda(SERVE) = \{entrega = N\} \\
 t_A(S) &= \begin{cases} 0 & si \ S = \{SERVE\} \\ \infty & si \ S = \{IDLE\} \end{cases}
 \end{aligned}$$

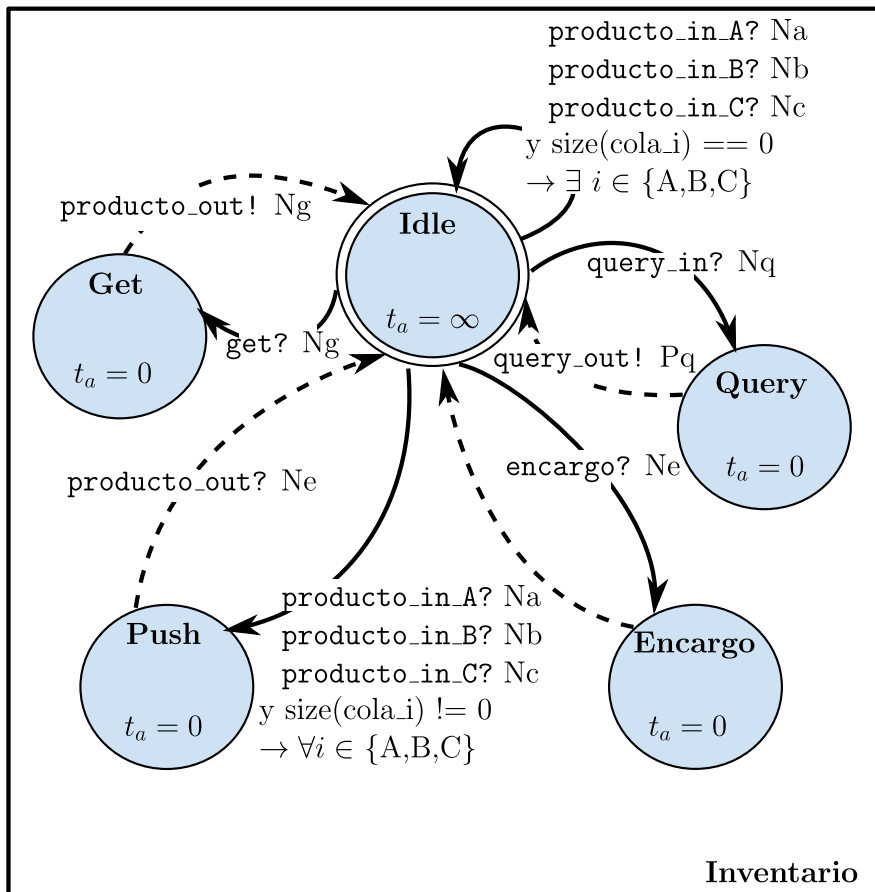


Figura 25: Diagrama de estados para el modelo de inventario.

En las figuras 26 y 27 se pueden observar los puertos de entrada/salida y el diagrama de estados respectivamente.

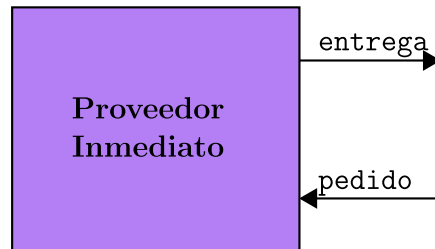


Figura 26: Entradas y salidas del modelo del proveedor inmediato

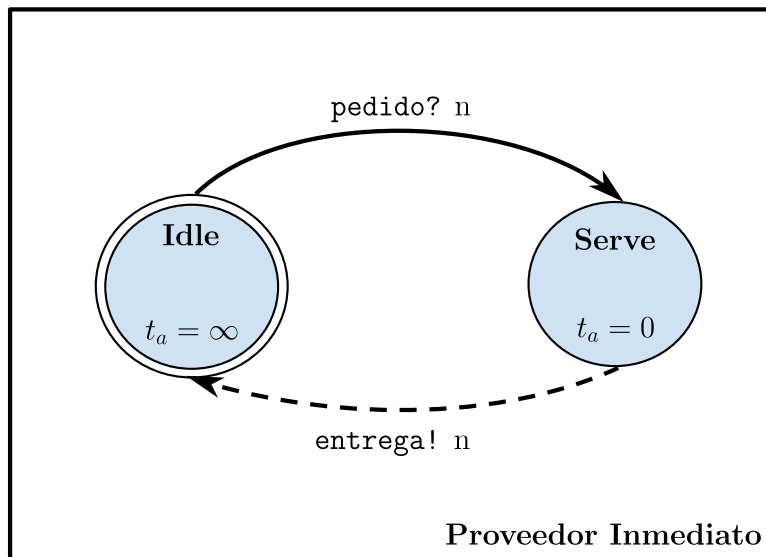


Figura 27: Diagrama de estados para el modelo del proveedor inmediato.

3.10. Proveedor por múltiplo de cantidad fija

El modelo atómico correspondiente al bloque del proveedor por múltiplo de cantidad fija (PF) queda definido de la siguiente manera:

$$\begin{aligned}
 PF &= \langle S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_A \rangle \\
 S &= \{IDLE, SERVE\} \\
 X &= \{pedido\} \in \mathbb{N}_0^+ \\
 Y &= \{entrega\} \in \mathbb{N}_0^+ \\
 \delta_{int} &: \delta_{int}(SERVE) = IDLE \\
 \delta_{ext} &: \delta_{ext}(S, X) \begin{cases} Si S = IDLE \wedge pedido = N \\ \rightarrow \delta_{ext}(IDLE, pedido) = SERVE \end{cases} \\
 \lambda &: \lambda(SERVE) = \{entrega = \begin{cases} N Si N \% numProdPkg = 0 \\ \lfloor N / numProdPkg \rfloor \\ + N \% numProdPkg \forall N \end{cases} \} \\
 t_A(S) &= \begin{cases} 0 si S = \{SERVE\} \\ \infty si S = \{IDLE\} \end{cases}
 \end{aligned}$$

Donde:

$numProdPkg$ = es el número de productos por paquete cerrado que comercializa este proveedor.

En las figuras 28 y 29 se pueden observar los puertos de entrada/salida y el diagrama de estados respectivamente.

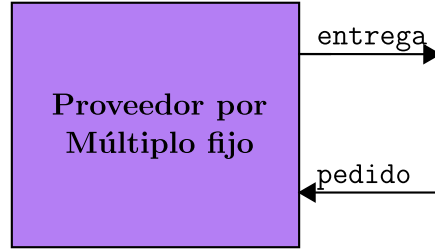


Figura 28: Entradas y salidas del modelo del proveedor por múltiplo de cantidad fija

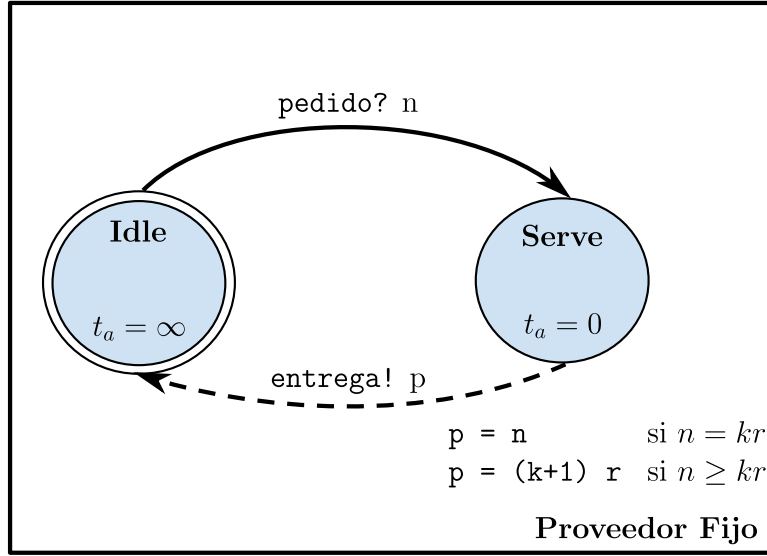


Figura 29: Diagrama de estados para el modelo del proveedor por múltiplo de cantidad fija.

3.11. Proveedor por Encargo

El modelo atómico correspondiente al bloque del proveedor por encargo (PE) queda definido de la siguiente manera:

$$\begin{aligned}
 PE &= \langle S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_A \rangle \\
 S &= \{IDLE, SERVE\} \\
 X &= \{\text{pedido}\} \in \mathbb{N}_0^+ \\
 Y &= \{\text{entrega}\} \in \mathbb{N}_0^+ \\
 \delta_{int} &: \delta_{int}(SERVE) = IDLE \\
 \delta_{ext} &: \delta_{ext}(S, X) \begin{cases} \text{Si } S = IDLE \wedge \text{pedido} = N \\ \rightarrow \delta_{ext}(IDLE, \text{pedido}) = SERVE \end{cases} \\
 \lambda &: \lambda(SERVE) = \{\text{entrega} = N\} \\
 t_A(S) &= \begin{cases} 1 & \text{si } S = \{SERVE\} \\ \infty & \text{si } S = \{IDLE\} \end{cases}
 \end{aligned}$$

En las figuras 30 y 31 se pueden observar los puertos de entrada/salida y el diagrama de estados respectivamente.

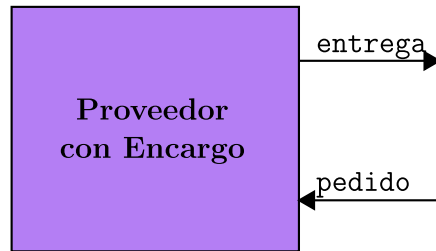


Figura 30: Entradas y salidas del modelo del proveedor por encargo

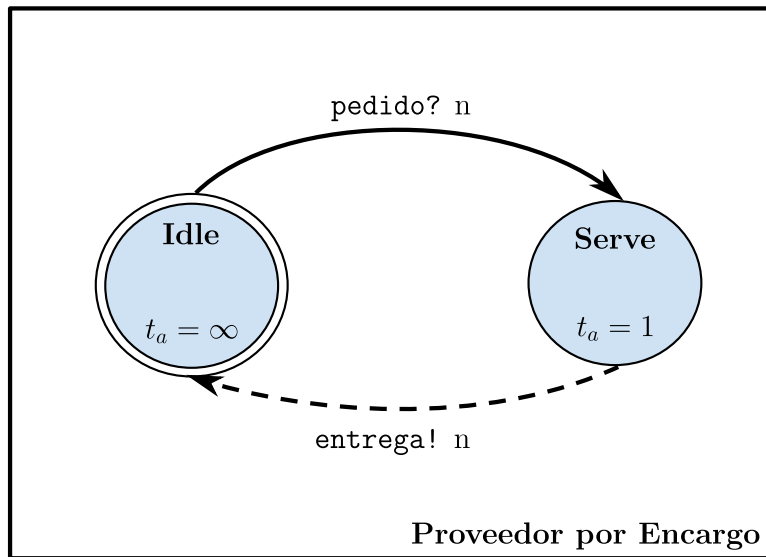


Figura 31: Diagrama de estados para el modelo del proveedor por encargo.

4. Modelado y simulación

4.1. Pruebas parciales

4.1.1. Cliente A

En la Figura 32 y Figura 33 se observan los histogramas que representan a la tasa de arribos de pedidos de clientes de tipo A y a la cantidad de unidades pedidas. El eje de tiempos de la Figura 32 se utiliza en segundos en lugar de en meses.

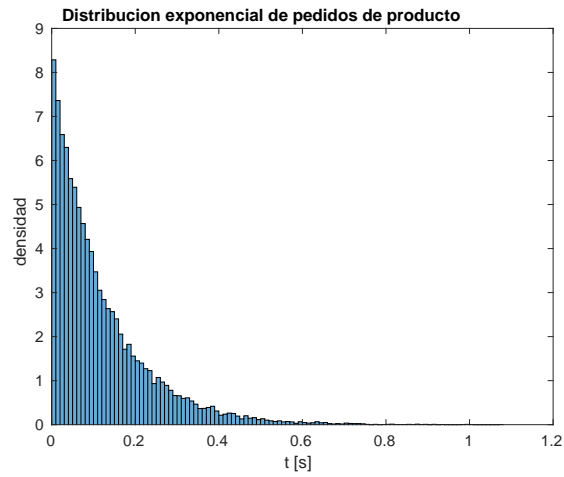


Figura 32: Función de densidad de probabilidad del tiempo entre pedidos de clientes tipo A.

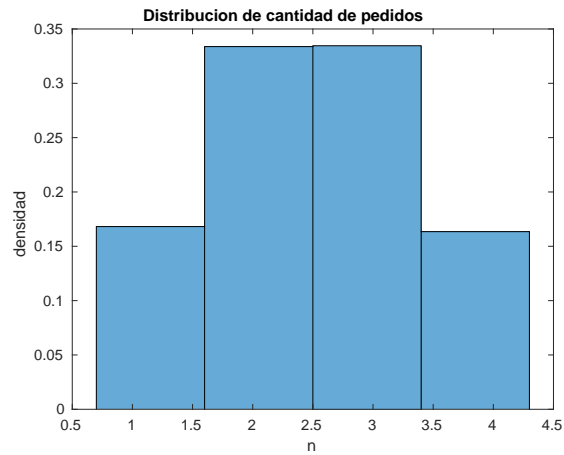


Figura 33: Función de densidad de probabilidad del número de unidades pedidas n .

4.1.2. Control de Inventario

Para la prueba del bloque de control de inventario se propuso el siguiente estímulo:

```
00:00:01:10 numProd_i 8
00:00:02:10 numProd_i 190
00:00:03:10 numProd_i 49
```

Dada la política de pedidos, se busca probar las condiciones:

- La cantidad de productos en el inventario es menor al valor requerido por la política de reposición, por lo tanto se debe pedir materiales a los proveedores.
- La cantidad de productos en el inventario es menor al máximo pero no alcanza el mínimo de la política de reposición por lo que no se deben pedir materiales.

La salida de la prueba es la siguiente:

```
00:00:01:000:0 Control Inventario – Consulta a Inventario
00:00:01:010:0 Control Inventario – Inventario responde: 8
00:00:01:010:0 Control Inventario – Pedido a proveedores: 92
00:00:02:010:0 Control Inventario – Consulta a Inventario
00:00:02:010:0 Control Inventario – Inventario responde: 190
00:00:02:010:0 Control Inventario – Pedido a proveedores: 0
00:00:03:010:0 Control Inventario – Consulta a Inventario
00:00:03:010:0 Control Inventario – Inventario responde: 49
00:00:03:010:0 Control Inventario – Pedido a proveedores: 51
00:00:04:010:0 Control Inventario – Consulta a Inventario
```

Donde se observa que se cumplen las condiciones mencionadas en la sección 3.5, ya que si el número de unidades se encuentra por debajo del umbral de política, el bloque pide, y de lo contrario no.

4.1.3. Control de Calidad

Para la prueba del bloque de control de calidad se propuso el siguiente estímulo:

```
00:00:01:00 queryClient_i 2
00:00:01:10 prod_i [1.2,1.3]
00:00:02:00 queryClient_i 3
00:00:02:10 prod_i [1.5,2.2,2.3]
00:00:02:20 prod_i [2.3]
00:00:03:00 queryClient_i 3
00:00:03:10 prod_i [3.2,3.3,0]
```

Se busca probar las condiciones:

- Un cliente compra N productos y el inventario devuelve N productos correctos. Por lo tanto se deberían enviar todos.
- Un cliente compra N productos y el inventario devuelve $N - 1$ productos correctos y uno vencido. Se debe descartar ese producto y pedir uno nuevo al inventario. El inventario devuelve un producto válido. Se deberían enviar los N productos válidos.
- Un cliente compra N productos pero el inventario dispone sólo de $N - 1$ y devuelve 0 en la última posición. El control de calidad debe interpretar la salida y despachar sólo los dos productos correctos.

La salida de la prueba es la siguiente:

```
00:00:01:000:0 Control Calidad - le piden: 2 productos
00:00:01:000:0 Control Calidad - pide: 2 productos
00:00:01:010:0 Control Calidad - recibe 2 productos
00:00:01:010:0 Control Calidad - se queda con 2 productos
00:00:01:010:0 Control Calidad - envia: 2 productos
00:00:02:000:0 Control Calidad - le piden: 3 productos
00:00:02:000:0 Control Calidad - pide: 3 productos
00:00:02:010:0 Control Calidad - recibe 3 productos
00:00:02:010:0 Control Calidad - se queda con 2 productos
00:00:02:010:0 Control Calidad - pide: 1 productos
00:00:02:020:0 Control Calidad - recibe 1 productos
00:00:02:020:0 Control Calidad - se queda con 3 productos
00:00:02:020:0 Control Calidad - envia: 3 productos
00:00:03:000:0 Control Calidad - le piden: 3 productos
00:00:03:000:0 Control Calidad - pide: 3 productos
00:00:03:010:0 Control Calidad - recibe 3 productos
00:00:03:010:0 Control Calidad - se queda con 2 productos
00:00:03:010:0 Control Calidad - el inventario esta vacio
00:00:03:010:0 Control Calidad - envia: 2 productos
```

Donde se observa que se cumplen las condiciones mencionadas en la sección 3.6.

4.1.4. Atención al cliente

Para la prueba del bloque de atención al cliente se propuso el siguiente estímulo:

```
00:00:01:00      queryClient_i 1
00:00:01:10  queryInventory_i 2
00:00:01:15  numProdClient_i 1
00:00:02:00      queryClient_i 3
00:00:02:10  queryInventory_i 2
00:00:02:15  numProdClient_i 2
00:00:03:00      queryClient_i 3
00:00:03:10  queryInventory_i 2
00:00:03:15  numProdClient_i 0
```

Se busca probar las condiciones:

- Un cliente solicita N productos y el inventario devuelve una cantidad $> N$ que se le informa al cliente. Por lo tanto el cliente compra N productos.
- Un cliente solicita N productos. El inventario responde con $P < N$ y el cliente decide comprar P productos.
- Un solicita N productos. El inventario responde con $P < N$ y el cliente decide no comprar ningún producto.

La salida de la prueba es la siguiente:

```
00:00:01:000:0  Atencion al cliente - le piden 1 productos
00:00:01:000:0  Atencion al cliente - pregunta por 1 productos
00:00:01:010:0  Atencion al cliente - le dicen que hay 2 stock
00:00:01:010:0  Atencion al cliente - informa a cliente que hay 2 stock
00:00:01:015:0  Atencion al cliente - cliente pide 1 productos
00:00:01:015:0  Atencion al cliente - pide al inventario 1 productos
00:00:02:000:0  Atencion al cliente - le piden 3 productos
00:00:02:000:0  Atencion al cliente - pregunta por 3 productos
00:00:02:010:0  Atencion al cliente - le dicen que hay 2 stock
00:00:02:010:0  Atencion al cliente - informa a cliente que hay 2 stock
00:00:02:015:0  Atencion al cliente - cliente pide 2 productos
00:00:02:015:0  Atencion al cliente - pide al inventario 2 productos
00:00:03:000:0  Atencion al cliente - le piden 3 productos
00:00:03:000:0  Atencion al cliente - pregunta por 3 productos
00:00:03:010:0  Atencion al cliente - le dicen que hay 2 stock
00:00:03:010:0  Atencion al cliente - informa a cliente que hay 2 stock
00:00:03:015:0  Atencion al cliente - cliente pide 0 productos
```

Donde se observa que se cumplen las condiciones mencionadas en la sección 3.7

4.2. Pruebas de integración

4.3. Reporte y conclusiones

A. Código Implementado

<https://github.com/TwinT/DEVS-Inventario>