Modelado del establecimiento de la conexión entre dos dispositivos bluetooth usando las redes de Petri coloreadas

Modeling of the connection establishment between two bluetooth devices using colored Petri nets

María E. Villapol, Dra.
Laboratorio de Redes Móviles, Inalámbricas y Distribuidas (ICARO), Facultad de Ciencias, Escuela de Computación, Universidad Central de Venezuela
maria.villapol@ciens.ucv.ve

Recibido para revisión: 8 de Marzo de 2008, Aceptado: 28 de Noviembre de 2008, Versión final: 17 de Diciembre de 2008

Resumen—**Bluetooth es una tecnología de comunicación que proporciona comunicación entre dispositivos vía radio frecuencias en un área de alrededor de los 10 metros. La especificación de Bluetooth incluye un conjunto de protocolos, adoptados y propios, organizados de forma jerárquica. Uno de los protocolos propios de esta tecnología es el bandabase. El establecimiento de una conexión es parte de la funciones de dicho protocolo. La especificación de este procedimiento es poco clara y ambigua y hace poco uso de herramientas para la descripción de protocolos tales como las tablas de estado. En este trabajo, las *Redes de Petri Coloreadas* (*Coloured Petri Nets, CPNs*), las cuales son una técnica formal, se utilizan para modelar el establecimiento de una conexión entre dos dispositivos Bluetooth, uno maestro y otro esclavo. Dicho modelo es entonces validado y depurado usando la técnica del grafo de estado. El análisis inicial muestra que el modelo se comporta acorde a lo esperado dadas las asunciones del modelo y las hechas para fines del análisis. La mayor contribución de este trabajo es el haber logrado una especificación clara y precisa del procedimiento a través del uso de la CPNs.**

Palabras Clave—**Bluetooth, Establecimiento de la conexión bandabase, Piconet, CPNs.**

Abstract—**Bluetooth provides communication between devices via radio frequency in an area of around 10 meters. The Bluetooth specification includes a set of, adopted and fundamental, protocols hierarchically structured. Baseband is a fundamental protocol. Connection establishment is one of the functions of the baseband protocol. The protocol specification is not clear and ambiguous and hardly uses protocol specification tools such as state tables. In this paper, *Coloured Petri Nets (CPNs)*, which are formal techniques, are used to model the baseband connection establishment between a master device and a slave device. Then the model is validated and debugged using the state graph. The initial analysis shows that the model is behaved as expected and according to the model and analysis assumptions. The main contribution of this work is a clear and precise specification of the baseband connection establishment procedure using CPNs.**

Keywords—**Bluetooth, Baseband connection establishment, Piconet, CPNs.**

I. INTRODUCCIÓN

Bluetooth es una estándar para el soporte de la comunicación inalámbrica entre diversos tipos de dispositivos a distancias de aproximadamente 10 metros. Dada su corta cobertura, Bluetooth califica entre los estándares para las *Redes de Área Personal Inalámbricas (Wireless Personal Area Networks, WPAN)*, cuyo propósito es permitir la comunicación entre equipos en áreas de cobertura pequeñas que pueden abarcar, por ejemplo, un cuarto, una oficina o un automóvil. La especificación de Bluetooth, actualmente en la versión 2.1 [6], ha sido desarrollada por un grupo denominado Bluetooth *Special Interest Group (SIG)* que se forma a partir de fabricantes de dispositivos electrónicos de reconocida trayectoria como lo son Ercisson, IBM, Intel, Nokia y Toshiba [4]. Uno de los mayores inconvenientes de la especificación de Bluetooth es que hace poco uso de técnicas para la descripción de protocolos, tales como las tablas de estados, siendo la misma ampliamente narrativa. Como consecuencia, alguna de sus partes, tales como la descripción del establecimiento de una conexión bandabase, son ambiguas y difíciles de entender. Este hecho puede claramente afectar las implementaciones de esta tecnología.

Por otra parte, hay varias publicaciones que plantean el problema de la lentitud del establecimiento de una conexión Bluetooth a la vez que tratan de proponer mecanismos para aliviar este problema [20]. También se ha hablado de la incompatibilidad de ciertos equipos Bluetooth que fallan al momento del establecimiento de la conexión [7]. Dada la complejidad de la tecnología y el poco uso de herramientas para la descripción de protocolos que se hace en la especificación de Bluetooth, es posible que parte de los problemas antes mencionados sean debidos a la falta de una clara especificación de los procedimientos de los protocolos.

Los métodos formales proporcionan técnicas para soportar el diseño y mantenimiento de los protocolos de comunicación [2]. Las *Redes de Petri Coloreadas (Coloured Petri Nets, CPNs)*[11] son técnicas formales con bases matemáticas sólidas las cuales ya han sido utilizadas para el modelado de diversos sistemas tales como los protocolos de comunicación [11][13], y cuya utilización ha permitido obtener una descripción mas precisa de los mismos. En este trabajo se usan las CPNs para modelar el establecimiento de una conexión Bluetooth bandabase a nivel funcional, para lograr una especificación más clara y menos ambigua de este procedimiento. Adicionalmente, se valida el modelo usando una técnica de análisis como lo es la técnica del grafo de estado [11].

Hay varias razones para usar las CPNs para modelar y analizar Bluetooth. Las principales se resumen a continuación. Las Redes de Petri son una técnica madura. Eso se puede observar en los miles de artículos en revistas e informes de investigación generados en más de 30 años de trabajo teórico y práctico. Las Redes de Petri son soportadas por un estándar internacional [10] y varios libros de texto [16][18][19]. Ellas son una herramienta gráfica bien definida, que permiten el análisis formal. Esta técnica formal ya ha sido utilizada para modelar de forma satisfactoria otros protocolos de comunicación cuyas especificaciones presentaban problemas similares a los encontrados en la especificación de Bluetooth [8][15][21].

El autor ha encontrado pocos trabajos que incluyen la aplicación de técnicas formales a las actividades de ingeniería de protocolos asociadas a la tecnología Bluetooth. Uno de estos trabajos es el de Feldmann et al. [12], quien modela una *scatternet* Bluetooth completa usando las CPNs, para estudiar el rendimiento de la red. Otro fue nuestro propio trabajo, donde se desarrolló un modelo inicial del establecimiento de la conexión Bluetooth para un dispositivo maestro y otro esclavo usando CPNs [22]. Siguiendo una aproximación incremental, el modelo anterior ha sido revisado y redefinido para incluir nuevas características tales como el medio de transmisión y para extender la conexión a dos dispositivos, uno maestro y otro esclavo.

Con la finalidad de alcanzar los objetivos planteados, este artículo se divide en las siguientes secciones. La sección dos introduce brevemente la tecnología Bluetooth haciendo especial hincapié en el establecimiento de la conexión bandabase. Luego, en la sección tres, se describe la metodología utilizada para lograr la especificación formal y validación del procedimiento del establecimiento de la conexión bandabase. En la sección cuatro se realiza una descripción del modelo junto con las asunciones tomadas para el desarrollo del mismo. En la sección cinco se analiza el modelo a fin de validarlo usando la técnica de grafo de estado. Finalmente, la sección seis concluye este artículo así como también presenta los trabajos futuros y las recomendaciones que se desprenden del desarrollo del mismo.

II. DESCRIPCIÓN GENERAL DE BLUETOOTH

Bluetooth [3][6][14] es una tecnología de radio frecuencia (*Radio Fraquency, RF*) que ofrece conectividad a corta distancia para equipos personales, portables, PDAs, entre otros. Bluetooth está orientado al reemplazo de interfaces tradicionales, tales como RS-232 y conectores propietarios, a proporcionar una interfaz uniforme para acceder servicios de voz y datos, a proporcionar acceso a una red de área amplia usando un *gateway* personal, tal como un teléfono celular, y a proporcionar una comunicación sin infraestructura, que se puede usar para el soporte a grupos colaborativos (reuniones, conferencias).

Los dispositivos Bluetooth trabajan en la frecuencia de 2.4 GHz (más específicamente la banda de frecuencia en la mayoría de países es de 2.4 – 2.4835 GHz) también conocida como la banda para uso Industrial, Científico y Médico (o banda *Industrial, Scientific and Medical*, *ISM*). La transmisión de la señal ocurre usando una técnica de saltos de frecuencia elegidos de forma aleatoria [3][5][6], entre 79 canales físicos de 1 MHz en que se divide el ancho de banda usado por esta tecnología, siendo la tasa de transmisión de 1 Mbps.

A. Pila de Protocolos

La especificación de Bluetooth [5] incluye una descripción del núcleo que indica los detalles de los diversos protocolos que conforman la pila de protocolos; y una especificación de los perfiles que incluye los detalles del uso de la tecnología para soportar varias aplicaciones e indica cuales de los aspectos de la especificación del núcleo son obligatorios, opcionales y no aplicables. La Figura 1 muestra la pila de protocolos que conforman el estándar. La misma divide los protocolos en los siguientes niveles:

a) **Protocolos fundamentales de Bluetooth (protocolos del núcleo):** son específicos de Bluetooth y han sido desarrollados por el SIG de Bluetooth.
b) **Protocolos de sustitución de cable:** suministran señalización de control que emula el tipo de señalización que se asocia usualmente con los enlaces de cable.
c) **Protocolos de control de telefonía:** definen la señalización de control de llamada para establecer llamadas de voz y datos con dispositivos Bluetooth. También define un protocolo (Comandos AT) que especifica como puede controlarse un MODEM y un teléfono móvil.
d) **Protocolos adoptados:** son protocolos existentes que se utilizan para diversos fines en las capas superiores.

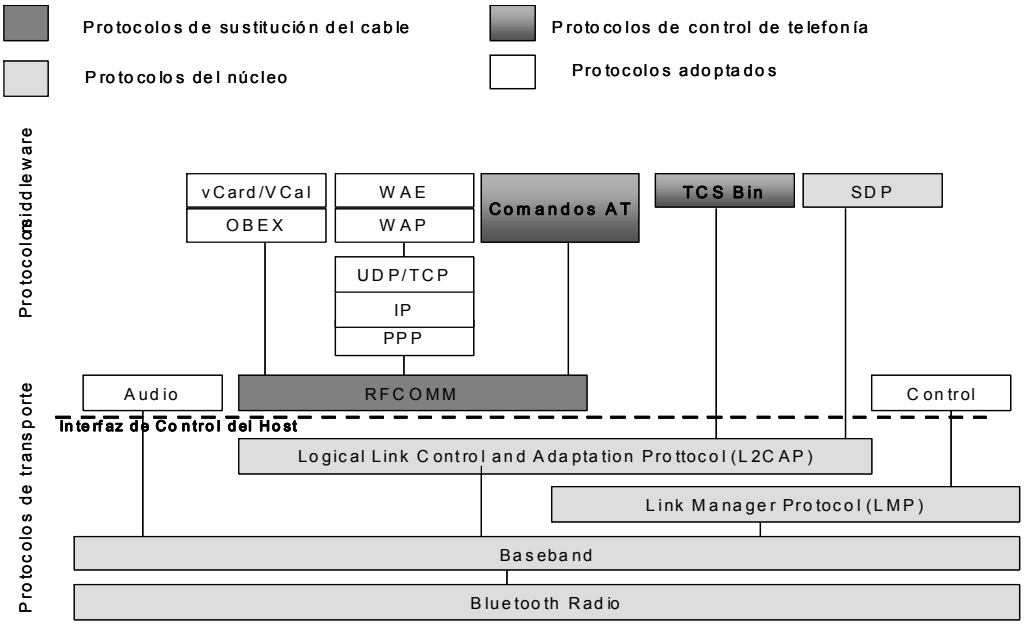


Figura 1: Pila de Protocolos de Bluetooth

B. Establecimiento de una Conexión Bandabase

En esta sección se describe brevemente el procedimiento de establecimiento de una conexión bandabase que es modelado como parte de esta investigación. Una descripción del funcionamiento detallado de este y de los demás protocolos de Bluetooth esta fuera del alcance de este trabajo y se puede encontrar en [5][6][14].

En Bluetooth, una *piconet* es una colección de dispositivos que pueden comunicarse. La *piconet* se forma de una forma ad hoc y contiene un dispositivo maestro y a lo sumo 7 dispositivos esclavos (ver Figura 2). Adicionalmente, un dispositivo en una *piconet* puede ser parte de otra *piconet* (como maestro o esclavo). Esta especie de solapamiento se conoce como *scatternet* (ver Figura 2).

La operación de Bluetooth se basa en el establecimiento, manejo y terminación de una conexión. La Figura 3 muestra el diagrama de transición de estados involucrados en el establecimiento de una conexión bandabase, el cual ha sido tomado de la especificación de Bluetooth [5]. Dichos estados se agrupan en estado de prevenido (*standby*), estado de indagación (*inquiry*), estado de *page* y estado de conexión (*connection*). El estado de *standby* es el estado inicial en que se encuentra un dispositivo el cual no ha establecido una conexión. En el estado de *inquiry*, un dispositivo colecta información acerca de otros dispositivos cercanos, tal como la dirección Bluetooth del dispositivo y valores del reloj. Está compuesto por varios sub estados; el de *inquiry*, ejecutado por el potencial maestro y los estados de *inquiry scan* y *inquiry response* ejecutados por los potenciales esclavos.

En el sub estado de *inquiry*, un potencial maestro transmite paquetes de indagación los cuales son recibidos por los esclavos en el sub estado de *inquiry scan*. Ya que durante el procedimiento de establecimiento de una conexión bandabase, los roles del maestro y del esclavo no están definidos, se denomina un *potencial maestro* aquel dispositivo que inicia un proceso de indagación destinado a establecer una conexión. En el sub estado de *inquiry scan*, un dispositivo busca mensajes de *inquiry* enviados por un potencial maestro. Una vez recibido un mensaje de *inquiry* un potencial esclavo debe entrar al estado de *inquiry response*.

En el estado de *page*, un dispositivo invita a otro a juntarse a su *piconet*. Similarmente al estado de indagación, el estado de *page* está compuesto por varios sub estados. Los sub estados de *page* y *master response* los cuales son ejecutado por el potencial maestro y los de *page scan* y *slave response* ejecutados por los esclavos. En el sub estado de *page*, un maestro puede activar y conectarse a un esclavo que está en el sub estado de *page scan*. Un esclavo entra en el sub estado de *slave response* cuando recibe un mensaje de *page*. En este estado el esclavo espera recibir un mensaje de *master response*. Después de recibir dicho mensaje, responde con otro mensaje y entra al estado de *connection* (es decir, está conectado con el dispositivo maestro). En el sub estado de *page scan*, el esclavo escucha por mensajes de *page* del esclavo. Un maestro entra en el estado de *master response* una vez recibido un mensaje de *page response* del esclavo. El maestro transmite un paquete conteniendo la información necesaria para que el esclavo pueda entrar en el estado de *connection*. Una vez que recibe una respuesta del esclavo, el maestro puede entrar al estado de *connection*.

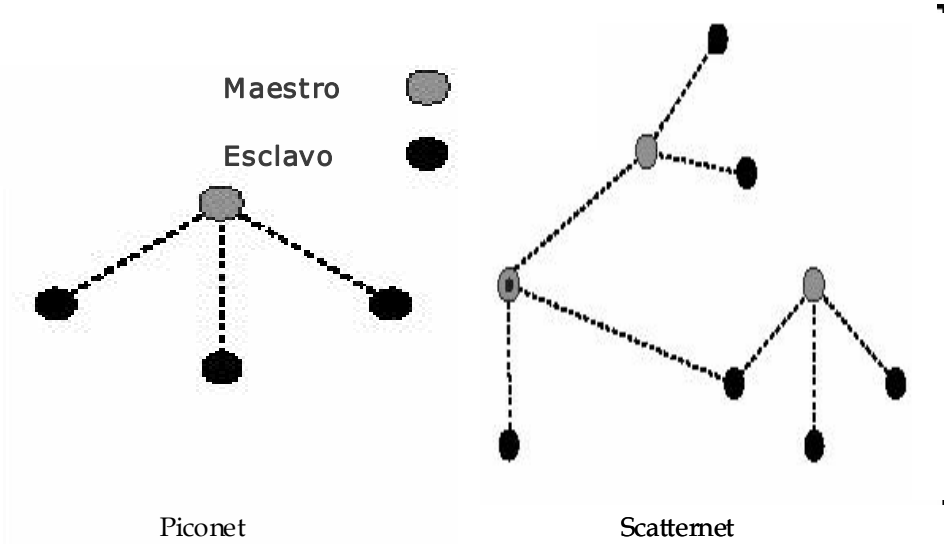


Figura 2: Ejemplo de una piconet y *scatternet*

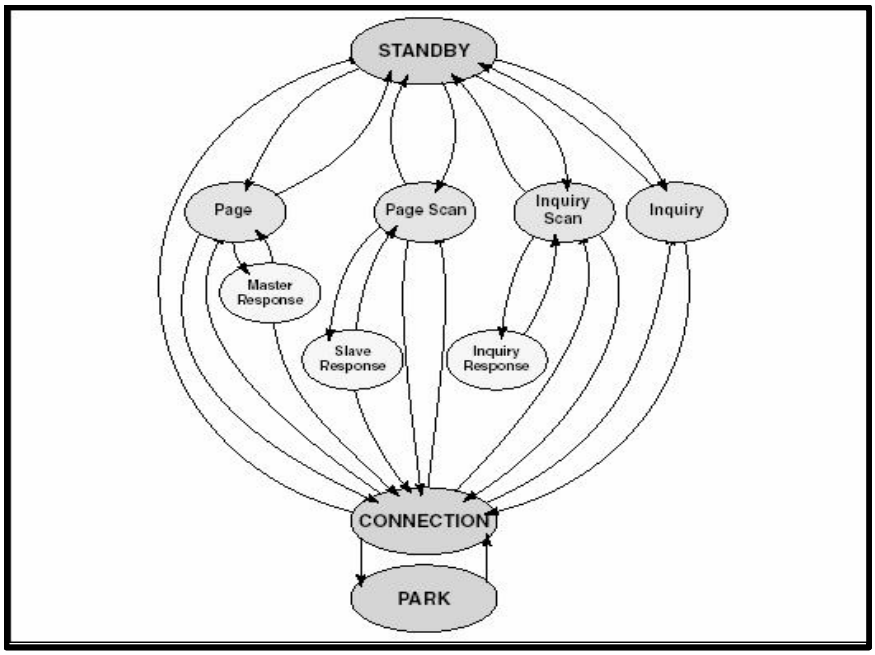


Figura 3: Diagrama de transición de estados del establecimiento de una conexión bandabase

Una vez conectado, un dispositivo puede encontrarse en alguno de los siguientes estados: un estado *activo* donde participa en una *piconet*. En este estado escucha, transmite y recibe paquetes; un estado de *husmeo* (*sniff*) donde escucha en *slots* específicos; un estado de *sostenido* (*hold*) que es un estado de potencia reducida, donde aun puede participar en el intercambio de paquetes síncronos; y un estado de *estacionado* (*park*) donde no participa en la *piconet*, pero es retenido como parte de ella. Finalmente, el dispositivo puede desconectarse en cualquier momento.

III. ESPECIFICACIÓN Y VALIDACIÓN DE PROTOCOLOS

La especificación y verificación formal de protocolos implica un conjunto de actividades, que se extienden desde la descripción de la arquitectura de protocolo hasta la verificación del modelo propuesto [2][9]. Billington et al. [1] presentan estas actividades como un conjunto de pasos sistemáticos y los denominan *metodología de verificación de protocolos*. Dicha metodología ha sido aplicada en [1][21] exitosamente. En este trabajo, esta metodología se utiliza como marco para la

especificación y la validación del establecimiento de una conexión bandabase Bluetooth. En vista de que esta metodología es bastante amplia, solo se han seguido los pasos destinados a la especificación y la validación del protocolo.

Las actividades implicadas en el modelado y análisis de la especificación del protocolo se muestran en la Figura 4. Por simplicidad, solo las actividades de la metodología que son implementadas en este trabajo son descritas a continuación, sin embargo la metodología es completamente explicada en [21].

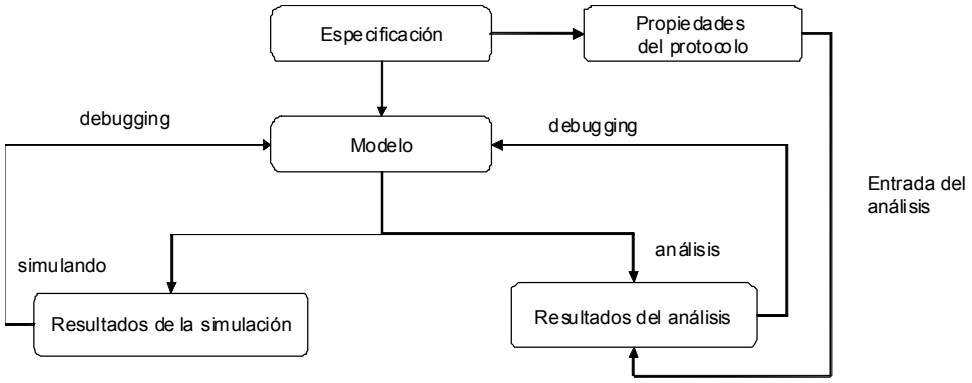


Figura 4: Diagrama mostrando las actividades de modelado y análisis de la especificación del protocolo

a) **Especificación:** describe el protocolo usando, por ejemplo, narrativa, diagramas de bloque o tablas de estado. Bluetooth es especificado en [5][6].
b) **Modelo: s**e desarrolla un modelo basado en la especificación usando una técnica formal. La técnica formal usada en este trabajo es las CPNs, con la ayuda de la herramienta de software CPN Tools [17].
c) **Propiedades Definidas:** una propiedad se refiere a una característica particular, que debe estar presente en un protocolo. Por ejemplo, Holzmann [9] define un conjunto de las características generales, que pueden aplicarse a cualquier protocolo, tales como: abrazos mortales inesperados (*deadlocks*) y no *livelocks*. Las propiedades dinámicas de las CPNs [11], tales como acotamiento (*boundedness*), vivacidad (*liveness*) y propiedades locales (*home properties*), también pueden ser estudiadas para validar el comportamiento del protocolo como se ha hecho en este trabajo y en [21].
d) **Resultados de la Simulación:** las simulaciones se pueden utilizar para eliminar errores iniciales del modelo. Por ejemplo, la simulación automática o interactiva proporcionada por CPN Tools puede ayudar a encontrar errores tales como secuencias de eventos del protocolo erróneas. Mientras se encuentren errores en la simulación, se modifica el modelo y las actividades de simulación se repiten según lo mostrado en la Figura 4.
e) **Resultados del Análisis:** el análisis del modelo del establecimiento de la conexión bandabase de Bluetooth consiste en validar el protocolo según las propiedades dinámicas de las CPNs. Varios métodos para el análisis formal de los protocolos de comunicación se han definido (por ejemplo, análisis del grafo de estado, invariantes del sistema, lógica temporal, chequeo de modelos). Holzmann [9] describe algunos de estos métodos. En esta investigación se emplea el método del grafo de estado incluido en CPN Tools. Los resultados del análisis pueden arrojar algunos errores. Los errores necesitan ser analizados para determinar sus causas y pueden ser una consecuencia de, por ejemplo, un error en el modelo, una inexactitud de la especificación o de las asunciones hechas. Así, el modelo puede o no ser modificado. Si el modelo requiere ser modificado entonces las actividades de simulación y análisis se deben repetir según se muestra en la Figura 4.

IV. MODELO CPN DEL ESTABLECIMIENTO DE UNA CONEXIÓN BANDABASE

En la especificación de Bluetooth no se describen claramente las transiciones entre los estados mostradas en la Figura 3. A continuación se presenta un modelo del establecimiento de una conexión bandabase basado en la interpretación del autor de dicha especificación [5] y en la descripción dada en [14]. El modelo es creado usando las CPNs con la ayuda de CPN Tools versión 2.2.0 [17].

A. Alcance

La especificación del protocolo bandabase de Bluetooth es compleja [5]. Por lo cual se ha tomado una aproximación incremental para la realización del modelo del establecimiento de una conexión bandabase. Así, en este artículo, se presenta una versión mejorada del modelo presentado en [22] e incluye el procedimiento realizado para el establecimiento de una conexión entre dos dispositivos (ver Figura 5). En este modelo, solo el establecimiento de una conexión Bluetooth entre un maestro y un esclavo es considerada. Vale destacar que esta es

una limitación que se encuentra en varios de los dispositivos móviles con capacidades restringidas tales como los teléfonos celulares. Adicionalmente, aunque las CPNs soportan el modelado de las restricciones temporales, en esta versión del modelo ellas han sido omitidas. Esto es porque inicialmente solo se está interesado en la especificación funcional de todas las transiciones mostradas en la Figura 3.

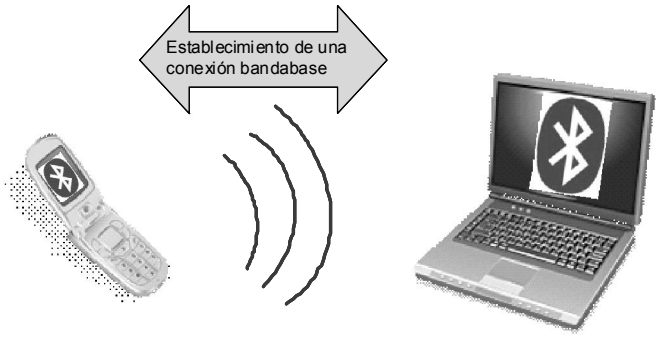


Figura 5: Topología de la *piconet* Bluetooth modelada

B. Modelo Jerárquico

Similarmente a otros modelos complejos de protocolos de comunicación (tales como el presentado en [21]), en este trabajo se utilizan los constructores jerárquicos de las CPNs [11][13]. Las jerarquías se construyen usando la noción de *transición de sustitución*, la cual puede ser considerada como una macro expansión. El modelo se inicia con un diagrama CPN en el nivel superior, el cual proporciona una visión general del sistema que está siendo modelado y su ambiente. En las CPNs jerárquicas, este diagrama en el nivel superior contendrá un número de transiciones de sustitución. Cada una de estas transiciones de sustitución es refinada en otro diagrama CPN, el cual puede a su vez contener transiciones de sustitución. El diagrama en el nivel superior y cada una de la transiciones de sustitución es definida por un modulo, denominado *página*.

El modelo del establecimiento de una conexión Bluetooth a un alto nivel de abstracción se muestra en la Figura 6. Este incluye una transición de sustitución (dibujada en forma de rectángulo) para el procedimiento de establecimiento de la conexión que se realiza en el maestro (MASTER) y otra para el que se realiza en el esclavo (SLAVE), las cuales son definidas en sus propias páginas. El medio de comunicación que en el caso de Bluetooth es compartido y es representado por la transición COMMUNICATION_CHANNEL.

Otro componente básico de una CPN es una *plaza* la cual es dibujada en forma de circulo o elipse y puede representar una condición o un estado. Cada plaza tiene un *tipo* asociado o un conjunto de colores (*colour set*), el cuál determina el tipo de datos que la plaza puede tener. En la Figura 6, las plazas MASTERTOSLAVE y SLAVETOMASTER son del tipo PACKET (el cual es descrito en la Sección IV.C) y representan los paquetes bandabase que viajan hacia el esclavo o hacia el maestro, respectivamente. Las plazas y transiciones de sustitución están conectadas por *arcos*, los cuales indican el tipo de data requerida o producida por las transiciones de sustitución.

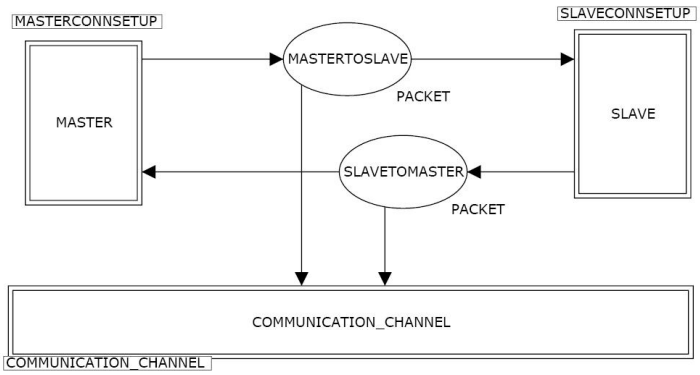


Figura 6: Vista jerárquica del establecimiento de una conexión Bluetooth

Las páginas y sub páginas de una CPN se interconectan a través de plazas *puertos* y *zócalos*. Las sub páginas tienen plazas puertos, que les permiten recibir y entregar marcas (ver Sección V.A) de las páginas de un nivel más alto. Las plazas conectadas con la transición de substitución MASTER (Figura 6), tiene una plaza entrante y una saliente (SLAVETOMASTER y MASTERTOSLAVE, respectivamente), que se llaman zócalos. Los zócalos son relacionados con las plazas puertos en las correspondientes sub páginas proporcionando las asignaciones de puertos. Similarmente pasa con las otras transiciones de sustitución mostradas en la figura.

C. Declaración Global

El modelo presentado en este trabajo también incluye una Declaración Global. Esta define las declaraciones requeridas por las inscripciones CPN. La Figura 7 muestra los conjuntos de colores (tipos) y variables de la declaración global. Los colores BOOL, STRING e INT representan los tipos *boolean*, *string* y entero, respectivamente, presentes en otros lenguajes de programación. El color STATE es del tipo enumerado y representa los estados en que un dispositivo intentando establecer una conexión bandabase puede estar. Estos estados fueron explicados en la Sección 2.2. El color TYPE es un enumerado y representa los tipos de paquetes involucrados en el intercambio de mensajes entre maestro y esclavos que están intentando establecer una conexión. El color AC es también un tipo enumerado y contiene el parámetro de control de acceso usado para identificar dispositivos durante el establecimiento de una conexión [5]. El color PACKET es el producto del tipo TYPE y AC y representa un paquete bandabase. El color TIND es un enumerado con un solo valor usado para controlar ciertas acciones en el modelo. Las demás declaraciones corresponden a variables (var) usadas en el modelo.

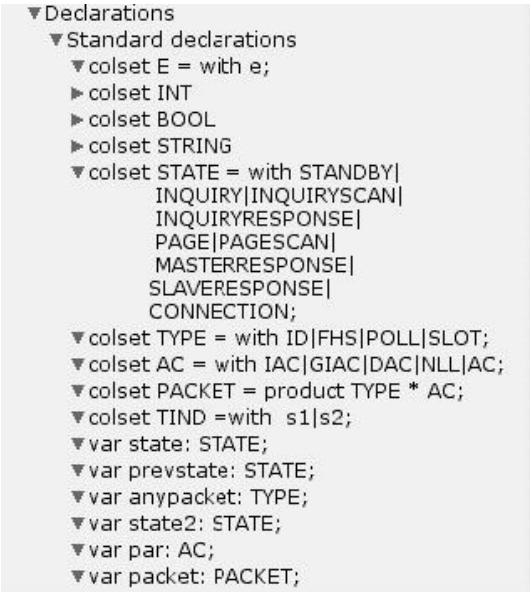


Figura 7: Declaración global

D. Página de Establecimiento de una Conexión en el Maestro

La jerarquía del modelo del establecimiento de una conexión bandabase consiste de ocho (8) páginas incluyendo la mostrada en la Figura 6. No todas las páginas del modelo pueden ser descritas en este artículo debido a limitaciones de espacio. Así, solo las páginas relacionadas al establecimiento de una conexión en un dispositivo maestro son descritas. Ellas son representativas del comportamiento del modelo CPN.

La Figura 8 muestra la página de establecimiento de una conexión en el maestro. La misma incluye dos (2) transiciones de sustitución, INQUIRY y PAGE, las cuales modelan los dos grandes procedimiento envueltos en el establecimiento de una conexión, es decir *Inquiry* y *Page* (ver Sección II.B), respectivamente. Ellas se expanden en sus respectivas páginas. La plaza MASTERSTATE es del tipo STATE y representa los estados del establecimiento de una conexión bandabase en los cuales un dispositivo maestro puede estar. Las otras plazas son plazas puertos de las plazas descritas en la Sección IV.B.

Figura 8: Página de establecimiento de una conexión en el maestro

E. Página de Inquiry

La Figura 9 muestra la página que modela el procedimiento de *Inquiry* realizado por un dispositivo que pasa a ser un potencial maestro. La misma incluye siete (7) *transiciones*. En las CPNs las transiciones se dibujan en forma de rectángulo y representan las acciones del sistema. Similarmente a las transiciones de sustitución, ellas se conectan a las plazas a través de arcos. En la Figura 9, las transiciones modelan las acciones para pasar de un estado a otro. Un dispositivo puede entrar al modo de Inquiry automáticamente basado en un rango periódico especificado o manualmente cuando se invoca un comando de HCI Inquiry [6]. Las transiciones Inquiry_Period-Ends y HCI_Inquiry modelan estas dos acciones respectivamente. Dichas transiciones hacen que un dispositivo salga del estado de STANDBY y entre en el estado INQUIRY y que se transmitan paquetes de indagación (ID). El dispositivo continua indagando a través del envío de paquetes ID, lo cual es modelado por la transición Inquiry-Continues. Entre las transmisiones de estos paquetes, el dispositivo puede escuchar por respuestas (es decir paquetes FHS) y esto es modelado por la transición ResponseReceived. El período de indagación termina cuando se genera un comando de cancelación del procedimiento de indagación, cuando se ha alcanzado un *time out* (*inquiryTO*) o cuando es parado por el Manejador de Recursos BandaBase [5] porque se han recibido suficientes respuestas. Estas acciones son modeladas por las transiciones HCI_Inquiry_Cancel, InquiryTO y SufficientNumberResponses, respectivamente. Cuando alguna de estas transiciones ocurre el dispositivo regresa al estado de STANDBY.

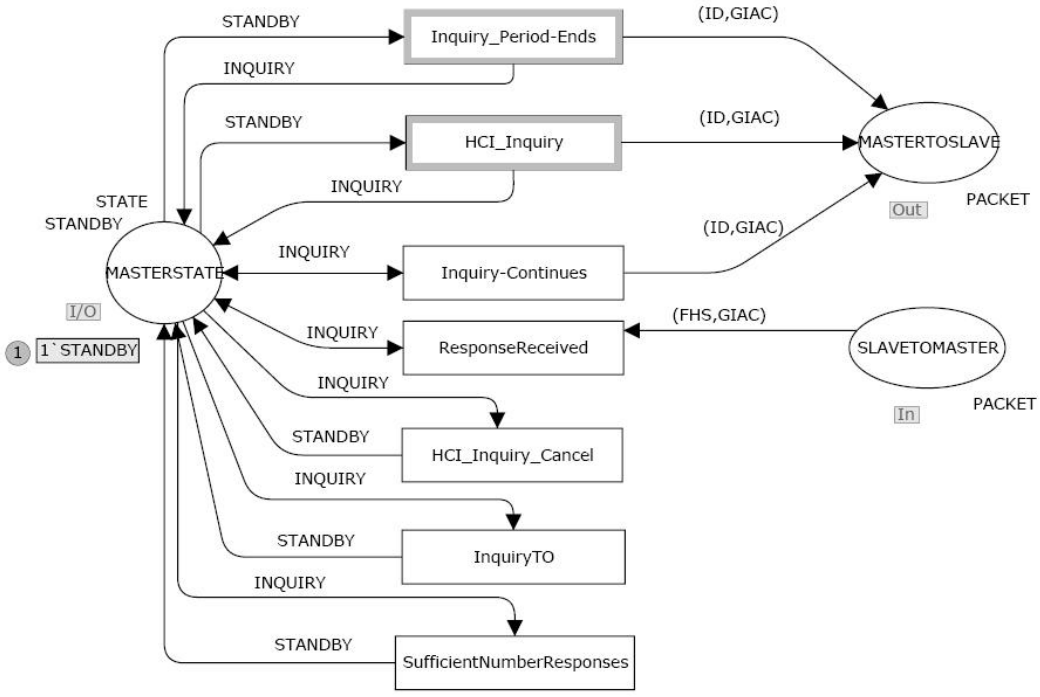


Figura 9. Página de Inquiry

F. Página de Page

La Figura 10 muestra la página que modela el procedimiento de *Page* realizado por el potencial maestro. La misma contiene diez (10) transiciones. La transición HCI_Create_Connection modela las acciones ejecutadas cuando se invoca un comando HCI_Create Connection [6] el cual causa que un dispositivo comience un proceso de *Page* para establecer una conexión. Cuando esta transición ocurre el dispositivo pasa al estado del estado de STANDBY al estado de PAGE y comienza a transmitir paquetes ID con el código de acceso del dispositivo (DAC) esclavo. En el estado de PAGE el dispositivo continua transmitiendo paquetes de *page* (es decir paquetes ID) tal como es modelado por la transición Paging. Si un *time out* (pageTO) es excedido, el dispositivo debe retornar al estado de STANDBY. Esto es modelado por la transición pageTO. Por el contrario, la transición SlaveResponse modela la acciones ejecutadas por un dispositivo cuando una respuesta (ID,DAC) del potencial esclavo es recibida. En este caso el dispositivo entra en el estado de MASTERRESPONSE y envía un paquete FHS al esclavo. Después de que el maestro ha enviado este paquete debe esperar por una segunda respuesta del esclavo, reconociendo la recepción del paquete FHS. Esto es modelado por la transición ScndSlaveResponse. En este caso el maestro pasa al estado de conectado (CONNECTION) y transmite su primer paquete (un paquete de POLL).

En caso, de que el maestro no reciba una segunda respuesta del esclavo, el maestro debe retransmitir el paquete FHS. Esto es modelado por la transición noScndSlaveResponse. Este proceso debe continuar hasta que se recibe una respuesta o se excede un *time out* (*pagerespTO*), tal cual es modelado por la transición pageRespTO. En cuyo caso el dispositivo debe retornar al estado de PAGE.

El maestro debe esperar por una respuesta del esclavo al paquete de POLL, si el mismo no es recibido dentro de *newconnectionTO* número de *slots*, el maestro debe retornar al estado de PAGE. Esto es modelado por la transición newconnectionTO. Por el contrario, la transición FirstSlavePacktRecv modela las acciones ejecutadas cuando esta respuesta del esclavo ha sido recibida. Finalmente, un dispositivo puede desconectarse, Disconneting, y retornar al estado de STANDBY.

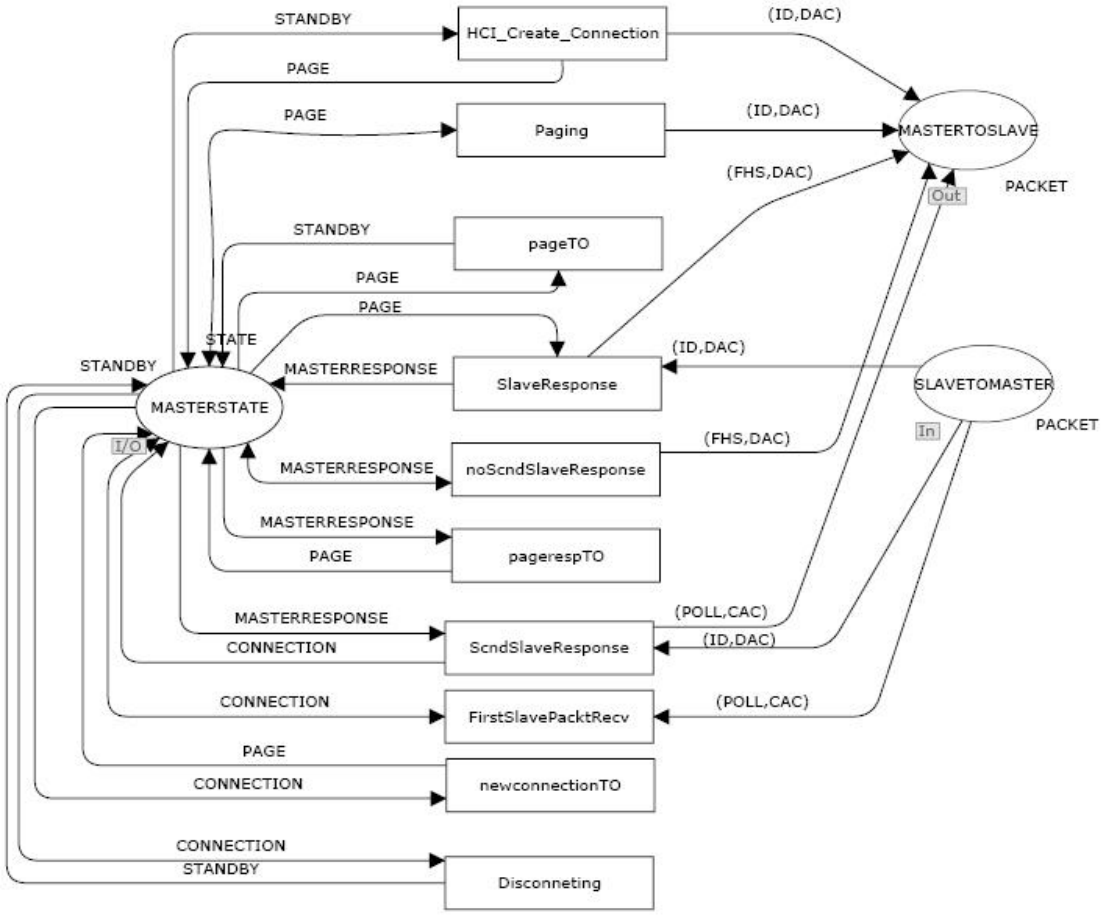


Figura 10. Página de Page

G. Página de Communication Channel

La Figura 11 muestra la página que modela el canal de comunicación compartido entre los dispositivos Bluetooth. La misma tiene dos (2) transiciones. Los paquetes intercambiados entre dispositivos pueden perderse o dañarse en el camino, esto es modelado por las transiciones MASTERLOST y SLAVELOST.

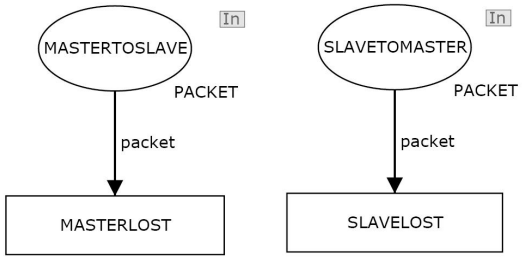


Figura 11. Página de Communication_Channel

V. ANÁLISIS Y SIMULACIÓN DEL MODELO CPN

El modelo descrito anteriormente es simulado y analizado para estudiar su comportamiento. En esta sección se describe como es el comportamiento dinámico de un sistema CPN y el análisis del modelo usando la técnica del grafo de estado.

A. Comportamiento Dinámico de las CPNs

El comportamiento dinámico de un sistema CPN se puede describir como el cambio del marcado de la red según las ocurrencias de las transiciones, que dependen de las expresiones en sus arcos circundantes. Un *marcado* de una plaza comprende un (*multi*) conjunto de valores (conocidos como *tokens* o *marcas*) tomadas del tipo de la plaza. El *marcado inicial* incluye la distribución de *tokens* en cada una de las plazas del modelo.

Una transición puede ocurrir si está *habilitada* (*enabled*). Para que una transición este habilitada en el marcado actual, debe ser posible *asociar* (asignar) valores de los datos a las variables que aparecen en las expresiones circundantes al arco y en el *guard* y ciertas condiciones debe ser satisfechas. Una transición puede tener una expresión *boolean* asociada a ella denominada *guard* y se incluye entre corchetes. Similarmente a la expresión de un arco, un *guard* puede tener variables. Así, para que una transición ocurra, en primer lugar, cada una de las expresiones de los arcos entrantes evalúa a las marcas que están presentes en las plazas de entrada correspondientes. En segundo lugar, si hay cualquier *guard*, debe evaluar a verdad.

La ocurrencia de una transición remueve marcas de las plazas entrantes y agrega marcas a las plazas salientes. Las marcas removidas son el resultado de evaluar las expresiones en los arcos entrantes correspondientes, mientras que los valores de las marcas agregadas son el resultado de evaluar las expresiones del arco en los arcos salientes correspondientes.

B. Análisis del Modelo

El modelo descrito anteriormente es analizado generando el grafo de estado o *Grafo de Ocurrencias* (*Occurrence Graph, OG*) y su correspondiente grafo de *Componentes Fuertemente Conectados* (*Strongly Connected Component, SCC*). El grafo de estado incluye todos los marcados posibles que se puedan alcanzar desde el marcado inicial y se representa como un grafo dirigido donde los nodos representan los marcados y los arcos los elementos de asociación que ocurren.

Por otra parte, un *Componente Fuertemente Conectado* (*Strongly Connected Component, SCC*) del grafo de estado es un sub-grafo máximo, cuyos nodos son mutuamente accesibles entre cada uno de ellos [11]. Un grafo SCC tiene un nodo por cada SCC y arcos que conectan cada nodo del SCC con uno o más nodos SCC. Un SCC sin arcos entrantes se llama SCC inicial, y un SCC sin arcos salientes se llama SCC terminal. Cada nodo en el grafo de estado pertenece solamente a un SCC, así que el grafo SCC será más pequeño o igual que el grafo de estado correspondiente.

En este trabajo, el grafo de estado es usado para investigar algunas propiedades dinámicas de las CPNs [11], tales como acotamiento (*boundedness*), vivacidad (*liveness*) y propiedades locales (*home properties*), así como también para chequear al protocolo. Para la generación del grafo de estado y del reporte con los resultados de las propiedades anteriores se usó CPN Tools versión 2.2.0 [17]. Esta herramienta corrió en una máquina con procesador Intel Core 2 de 2,13 GHZ y 3 GB de memoria RAM corriendo el sistema operativo Windows XP versión 2002.

El modelo CPN descrito en la Sección 4, genera un grafo de estado infinito debido a los paquetes periódicos (de *inquiry* y *page*) que envía el maestro, y al hecho de que las plazas de comunicación (MASTERTOSLAVE y SLAVETOMASTER) no están acotadas. Así, se ha modificado el modelo usando una aproximación estándar [13][21], de forma tal que las plazas de comunicación tengan una capacidad finita.

La Figura 12 muestra el modelo modificado para la página de INQUIRY descrita en la Sección 4.5. El paquete tipo SLOT ha sido incluido en el *color set* del tipo TYPE (ver Sección 4.3). Cada vez que se desea enviar un paquete debe existir suficiente capacidad en la plaza de comunicación MASTERTOSLAVE; dicha capacidad viene determinada por el número de *tokens* (SLOT,NLL). Cada vez que se recibe un paquete de la plaza SLAVETOMASTER debe colocarse un *token* (SLOT,NLL). Todas las páginas del modelo han sido modificadas de forma similar.

1) Marcado Inicial

A fin de analizar el modelo del establecimiento de la conexión bandabase la plazas MASTERSTATE y SLAVESTATE son inicializadas con el estado de STANDBY. Las plazas MASTERTOSLAVE y SLAVETOMASTER son inicializadas incrementalmente para permitir un máximo de 2,3,4 y 5 paquetes Bluetooth. Las demás plazas no tienen ningún *token* en el marcado inicial.

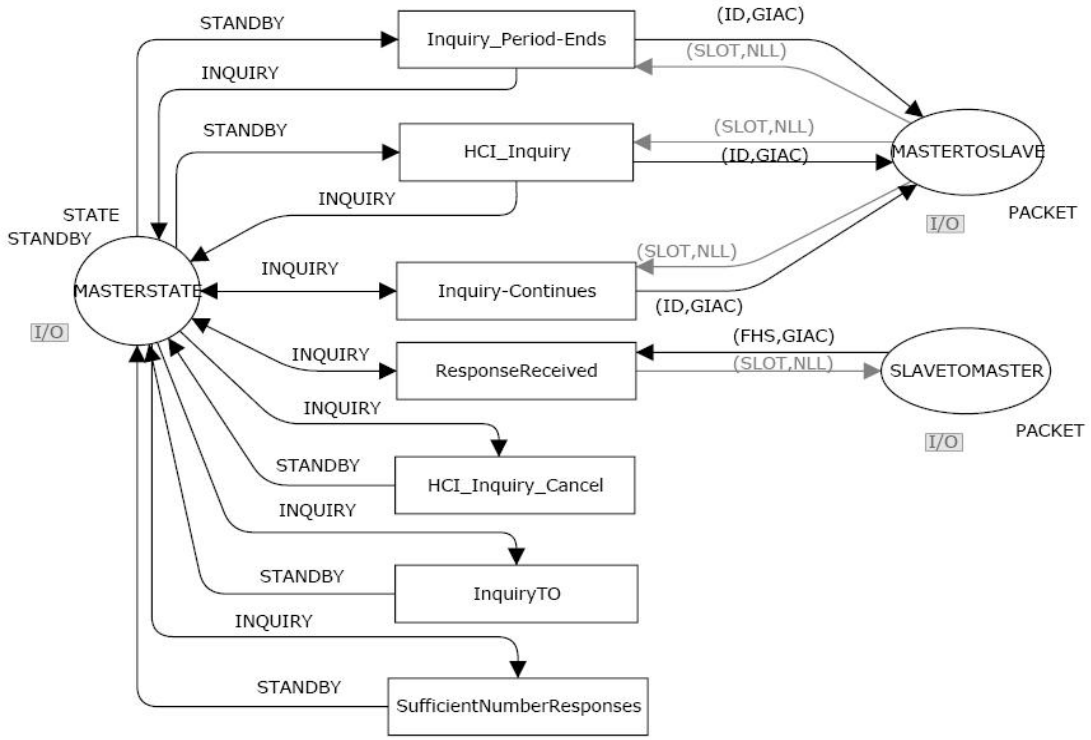


Figura 12. Página de Inquiry modificada

2) Estadística del Grafo de Ocurrencia

Par analizar el modelo se siguió una aproximación donde se fue aumentando incrementalmente la capacidad máxima de las plazas que comunican los dispositivos, es decir MASTERTOSLAVE y SLAVETOMASTER, de 2 paquetes hasta 5 paquetes. Esto va incrementando la confianza de que el modelo se comporta acorde a lo esperado a la vez que el grafo de estado se mantiene en un tamaño manejable para su análisis.

Para cada marcado inicial se generaron los reportes de estado completos para el análisis del modelo CPN. La información estadística que incluye el tamaño del grafo de estado (OG) y el grafo SCC se muestran en la Tabla 1. Puede observarse que a medida que la capacidad máxima de las plazas de comunicación aumenta el número de arcos y nodos aumenta. Sin embargo, el tamaño del SCC en todos los casos es uno. Esto es esperado ya que un dispositivo debe poder retornar al estado inicial desde cualquier estado alcanzable, tal como se muestra en la Figura 3.

Tabla 1. Información estadística del grafo de estado

Número máximo de paquetes en un canal de comunicación representados por las plazas de comunicación	2		3		4		5	
Información estadística	OG	Grafo SCC	OG	Grafo SCC	OG	Grafo SCC	OG	Grafo SCC
Número de Nodos	2764	1	16340	1	64664	1	200364	1
Número de Arcos	16126	0	111682	0	488565	0	1622989	0
Tiempo de cálculo (hh:mm:ss)	(00:00:02)	(00:00:00)	(00:00:61)	(00:00:02)	(00:15:12)	(00:00:23)	(02:16:01)	(00:06:02)

C. Propiedades Generales

Las propiedades de acotamiento y vivacidad [11] son investigadas para validar y depurar el modelo y para estudiar el comportamiento del protocolo bandabase en lo que respeta al establecimiento de una conexión. Esta información es tomada del reporte de grafo de estado generado por CPN Tools [17].

1) Acotamiento

Las *cotas enteras* y las *cotas de los multi-conjuntos* son analizadas para las plazas del modelo. Esta información es resumida en la Tabla 2 para todos los grafos de estados generados. Las cotas enteras superiores describen el máximo número de *tokens* que pueden estar en una plaza, mientras que las cotas de los multi-conjuntos indican que *tokens* pueden estar en una plaza [11]. Las plazas MASTERSTATE y SLAVESTATE puede tener una máximo de un *token* ya que un dispositivo solo puede encontrarse en un estado en un momento dado. El valor de las cotas de los multi-conjuntos superiores indican todos los estados en que puede estar un maestro y un esclavo. Estos valores están acordes a lo explicado en la Sección II..B y a la especificación de Bluetooth [5]. El número máximo de paquetes que pueden intercambiarse entre un maestro y un esclavo está acorde al número máximo de paquetes en un canal de comunicación representados por las plazas de comunicación y establecido en el marcado inicial. Los múltiples paquetes se generan, primero, porque un dispositivo en un estado (tal como PAGE) puede siempre retornar al estado anterior (tal como sucede en la Figura 3). Segundo, los procedimientos de indagación (*inquiry)* y de *page* ejecutados por un maestro son periódicos generando múltiples copias de un paquete.

Tabla 2. Cotas superiores de las plazas del modelo

Plaza	Cota Entera	Cota de los Multi-Conjuntos
MASTERSTATE	1	1`STANDBY++ 1`INQUIRY++ 1`PAGE++ 1`MASTERRESPONSE++ 1`CONNECTION
SLAVESTATE	1	1`STANDBY++ 1`INQUIRYSCAN++ 1`INQUIRYRESPONSE++ 1`PAGESCAN++ 1`SLAVERESPONSE++ 1`CONNECTION
MASTERTOSLAVE	4	n`(ID,GIAC)++ n`(ID,DAC)++ n`(FHS,DAC)++ n`(POLL,CAC)++ n`(SLOT,NLL)
SLAVETOMASTER	4	n`(ID,DAC)++ n`(FHS,GIAC)++ n`(POLL,CAC)++ n`(SLOT,NLL)
InTransit	1	1`s1++ 1`s2

n = Número máximo de paquetes en un canal de comunicación representados por las plazas de comunicación.

La plaza InTransit es de control y puede solo tener un *token* con el valor s1 o s2, lo cual está acorde a los valores presentes en la Tabla.

2) Propiedades Locales y de Vivacidad

La Tabla 3 muestra las propiedades locales y de vivacidad. Un *marcado muerto (dead marking)* es un marcado sin elementos de asociación habilitados, es decir, ninguna transición puede ocurrir a partir de dicho marcado. En el modelo del establecimiento de una conexión, no hay marcados muertos. Esto quiere decir que el sistema no llega a ningún estado en el cual no puede seguir avanzando. Esto es esperado ya que el modelo está basado en el diagrama mostrado en la Figura 3.

Una transición está *viva* si puede ocurrir al menos una vez en una secuencia de ocurrencias para cada marcado de la red que es alcanzable desde el marcado inicial. Todas las transiciones del modelo están vivas y el sistema siempre retorna al estado donde tanto el maestro como el esclavo están en el estado de STANDBY con no paquetes en las plazas de comunicación.

Un *marcado local* (*home marking*) es un marcado que puede ser siempre alcanzado por el resto de los marcados alcanzables. En este caso todos los marcados son locales, lo que asegura que no importa que estado alcance el sistema siempre se pueda alcanzar cualquier otro estado, en particular el estado inicial.

Una *transición muerta* es aquella que no está habilitada en ningún marcado alcanzable [11]. El reporte del grafo de estado generado por CPN Tools muestra que no hay transiciones muertas. Esto es esperado ya que no debería haber "código muerto" en la especificación.

Tabla 3. Propiedades locales y de vivacidad

Propiedad	Valor
Marcados Muertos	Ninguno
Marcados Locales	Todos
Transiciones Muertas	Ninguna
Transiciones Vivas	Todas

VI. CONCLUSIONES

En este artículo, las CPNs han sido utilizadas para desarrollar un modelo del establecimiento de una conexión Bluetooth a nivel del protocolo bandabase entre dos dispositivos, asumiendo un medio de comunicación donde los paquetes pueden perderse o corromperse. Ya que la especificación del procedimiento de establecimiento de una conexión bandabase es compleja, se ha seguido una aproximación incremental, a través de la cual se irán incluyendo más características (tales como las restricciones de tiempo presentes en el procedimiento del establecimiento de una conexión Bluetooth) y casos de estudios al modelo a fin de aumentar la confiabilidad de que el mismo está correcto.

La especificación del establecimiento de una conexión bandabase, similarmente a otras partes de la especificación Bluetooth, se caracteriza por ser poco clara y en algunos casos ambigua. Adicionalmente, se ha encontrado que existen algunos errores de transcripción en la especificación. El modelo presentado en este trabajo define claramente, con la ayuda de un método formal, tal como los son las CPNs, el establecimiento de una conexión bandabase entre un dispositivo maestro y otro esclavo. El análisis inicial del modelo basado en el grafo de estado y las propiedades generales de una CPN muestra que los resultados son los esperados y el procedimiento de establecimiento de una conexión funciona acorde a lo especificado.

Los trabajos futuros comprenden analizar el modelo en función de ciertas propiedades específicas del protocolo bandabase usando técnicas tales como el chequeo de modelos (*model checking*). También, se desea incorporar restricciones de tiempo al modelo de forma tal de realizar algunos análisis en términos de la sincronización entre el maestro y los esclavos. A partir del modelo existente es muy sencillo incluir otros dispositivos para formar una *piconet* más grande a la analizada en este artículo. Esto indudablemente va a influir en el tamaño del grafo de estado y su tiempo de generación. Por lo tanto, los trabajos futuros deben incluir el estudiar técnicas de simplificación de grafos de estado, que se puedan aplicar a este caso. Finalmente, se desea comparar los resultados obtenidos en la simulación del modelo con los obtenidos en pruebas reales usando equipos Bluetooth (con la ayuda de una herramienta de captura de paquetes Bluetooth) para conocer si la implementación está acorde a la especificación.

RECONOCIMIENTO

Este trabajo fue desarrollado con la ayuda financiera del Consejo de Desarrollo Científico y Humanístico (CDCH) de la Universidad Central de Venezuela (UCV) como parte del proyecto Nro. PI03-00-6224-2006.

REFERENCIAS

[1] Billington, J., M.C. Wilbur-Ham, and Bearman, M.T., 1986. Automated Protocol Verification. Protocol Specification, Testing, and Verification. M. Diaz (editor). Elsevier Science Publisher, pp 59-70.

[2] Billington, J., 1991. Formal Specification of Protocols: Protocol Engineering. Encyclopaedia of Microcomputers, Marcel Dekker, New York, Vol. 7, pp 299-314.

[3] Bisdikian, C., 2002. An Overview of the Bluetooth Wireless Technology. IEEE Communications Magazine. December. pp 86-95.

[4] Bluetooth. History of Bluetooth Technology. http://www.bluetooth.com/Bluetooth/SIG/History_of_the_SIG.htm. 2008.

[5] Bluetooth SIG, 2003. Inc. Specification of the Bluetooth System version 2.0. November.

[6] Bluetooth SIG, 2007. Specification of Bluetooth System, Covered Core Packing version 2.1. 26 July.

[7] Cruz, D., 2005. Evaluación de los Mecanismos y Protocolos de Seguridad Desarrollados para Redes Bluetooth. TEG, Escuela de Computación, UCV.

[8] Han, B. and Billington, J., 2002. Validating TCP Connection Management. Proceedings of the Workshops on Software Engineering and Formal Methods and Formal Methods Applied to Defense Systems, Adelaide, Australia, pp 47-55.

[9] Holzmann G., 1991. Design and Validation of Computer Protocols. Prentice Hall. 1991.

[10] ISO/IEC, 204. 15909-1:2004 Software Engineering - High-level Petri Nets - Concepts, Definitions and Graphical Notation, Standard.

[11] Jensen, K., 1997. Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use. Vol. 1, 2 and 3. Springer-Verlag, 2nd edition.

[12] Feldmann, S., Hartmann, T. y Kyamakya, K., 2003. Modeling and Evaluation of Scatternets Performance by using Petri Nets. In Proceedings of the International Conference on Wireless Networks, ICWN '03, June 23 - 26, Las Vegas, Nevada, USA. CSREA Press. ISBN 1-932415-03-3.

[13] Kristensen, L.M., Christensen S. and Jensen K., 1998. The practitioner's guide to coloured Petri nets. International Journal on Software Tools for Technology Transfer, Springer, Vol. 2, Number 2, pp 98-132.

[14] Millar, B.A., Bisdikian, C., 2000. Bluetooth Revealed: The Insider's Guide to an Open Specification for Global Wireless Communications. Prentice Hall.

[15] Ouyang, C., Kristensen, L.M. and Billington, J. A., 2002. Formal and Executable Specification of Internet Open Trading Protocol. Third International Conference, EC-Web 2002 Aix-en-Provence, France, LNCS 2455, pp 377-387.

[16] Peterson, J., 1998. Petri Net Theory and Modeling of Systems. Prentice-Hall.

[17] Ratzer, A.V, Wells, L, at al., 2003. CPN Tools for Editing, Simulating, and Analysing Coloured Petri Net. Lecture Notes in Computer Science, Volume 2679, pp. 450 – 462.

[18] Reisig, W., 1985. Petri Nets: An Introduction. Springer-Verlag.

[19] Reisig, W. and Rozenberg, G. (Eds.), 1998. Lectures on Petri Nets I: Basic Models, Advances in Petri Nets, Springer-Verlag, Vol.1, LNCS 1491.

[20] Salonidis, T.; Bhagwat, P. y Tassiulas, L., 2002. Proximity awareness and fast connection establishment in Bluetooth. 2000 First Annual Workshop on Mobile and Ad Hoc Networking and Computing. MobiHOC.

[21] Villapol, M.E., 2003. Modelling and Analysis of the Resource Reservation Protocol Using Coloured Petri Nets. Doctoral Thesis, University of South Australia.

[22] Villapol, M.E., 2006. Modelado y Análisis Inicial del Establecimiento de una Conexión Bluetooth Usando las Redes de Petri Coloreadas. en Proceedings of the Thirty-Second Latin American Computing Conference, CLEI 2006, Santiago de Chile, Chile.

Maria Elena Villapol B., es Licenciada en Computación de la Universidad Central de Venezuela (UCV), Caracas, Venezuela, en 1991, Maestría en Ciencias de la Computación de la UCV, 1996, Maestría en Comunicaciones Digitales obtenido en la Universidad de Monash, Melbourne, Australia, 1998, y Doctorado de la Universidad del Sur de Australia, Adelaide, Australia, 2003. Es profesora de la UCV desde 1992 y actualmente esta en el escalafón de Asociado. Su investigación y docencia se han orientado a las Tecnologías de las Redes de Computadores que incluyen estudios en ATM, IPv6 y RSVP y más recientemente WLANs (802.11), Bluetooth y WiMax. Los resultados de su investigación han sido publicados en más de una decena de congresos nacionales e internacionales, así como también en revistas especializadas en el área. Durante su carrera ha trabajado en varios proyectos de investigación incluyendo el denominado FedSat, como miembro investigador del Instituto para las Investigaciones en Telecomunicaciones (ITR), Adelaide, Australia. Actualmente, es coordinadora del proyecto de formación del laboratorio de Redes Móviles, Inalámbricas y Distribuidas (ICARO) de la UCV financiado por FIDETEL y es investigadora invitada de la Universidad de Florida Central.

Universidad Nacional de Colombia Sede Medellín
Facultad de Minas

Escuela de Ingeniería de Sistemas

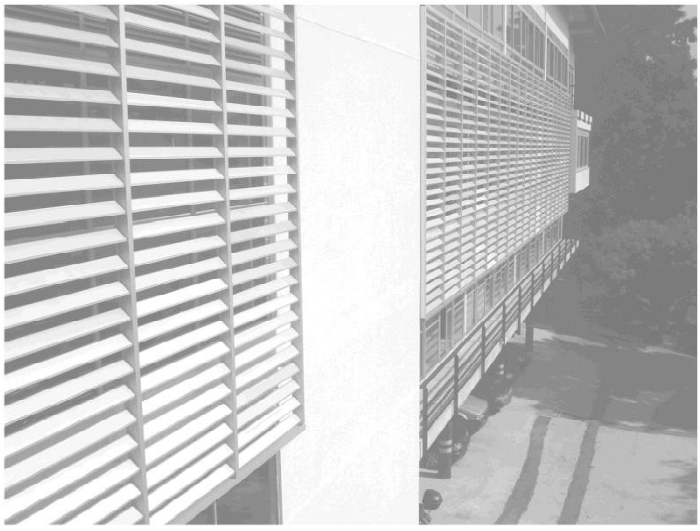
Grupos de Investigación

Grupo de Investigación en Sistemas e Informática

Categoría A de Excelencia Colciencias
2004 - 2006 y 2000.

GIDIA: Grupo de Investigación y Desarrollo en Inteligencia Artificial

Categoría A de Excelencia Colciencias
2006 – 2009.



Grupo de Ingeniería de Software

Categoría C Colciencias 2006.

Grupo de Finanzas Computacionales

Categoría C Colciencias 2006.

Centro de Excelencia en Complejidad

Colciencias 2006

Escuela de Ingeniería de Sistemas
Dirección Postal:
Carrera 80 No. 65 - 223 Bloque M8A
Facultad de Minas. Medellín - Colombia
Tel: (574) 4255350 Fax: (574) 4255365
Email: esistema@unalmed.edu.co
http://pisis.unalmed.edu.co/