	Nombre Documento	Versión
	Net Check Manual Técnico	1.0




DIRECTV
Bogotá D. C., Colombia

Departamento de Informática


MANUAL TECNICO DEL SOFTWARE

APLICACIÓN NET CHECK

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

Contenido

1.	INTRODUCCIÓN	6
1.1	Objetivo.....	6
1.2	Alcance	6
1.3	Audiencia	6
1.4	Descripción general.....	6
1.4.1	Descripción de Proceso Técnico.....	7
1.4.1.1	Peticiones ICC.....	7
1.4.1.2	Peticiones Provisioning	7
1.4.1.3	Peticiones Robot Net	9
1.4.1.4	Peticiones OTRS.....	10
1.4.2	Despliegue.....	11
1.5	arquitectura general	12
2	BASE DE DATOS.....	12
2.1	Procedimientos almacenados.....	12
2.1.1	ICC	12
2.1.1.1	PCKDTVNETTST.stpQryInformacionCliente.....	13
2.1.1.2	stpQryInformacionHardware	13
2.1.1.3	fnObtenerIBSPorDevice	14
2.1.1.4	fnObtenerIBSPorDocumento	14
2.1.2	Robot Net.....	14
2.1.2.1	stpUpdHistoricoProvisioning.....	14
2.1.2.2	stpUpdLog	15
2.1.2.3	stpSitios.....	16
2.1.2.4	ctrHistoricoConsultaParametro	16
2.1.3	OTRS	18
2.1.3.1	stpQryTickets.....	18

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0


3 ARQUITECTURA DE APLICACION 19

3.1	Capa de Acceso a Datos	19
3.1.1	Clase CIProveedorBD.....	19
3.1.2	Clase CISQLServer.....	20
3.1.3	Clase CIOracle.....	21
3.1.4	Clase cIOleDb.....	21
3.1.5	Clase cIOdbc.....	21
3.2	Capa de negocio libDTVCOLNETTroubleshooting.....	22
3.2.1	Capa de negocio ICC.....	22
3.2.1.1	Clases entidades.....	22
3.2.1.2	Clases controladoras	23
3.2.2	Capa de negocio OTRS	27
3.2.2.1	Clases Entidades.....	27
3.2.2.2	Clases Controladoras.....	27
3.2.3	Capa de negocio ROBOT	31
3.2.3.1	Clases Entidades.....	31
3.2.3.2	Clases Controladoras.....	32
3.2.4	Capa de negocio TROUBLESHOOTING	40
3.2.4.1	Business Objects.....	40
3.2.4.2	Business Process	41
3.2.5	Otras clases	42
3.2.5.1	Clases cIDatasetFactory.....	42
3.2.5.2	Clases Enumerations	42
3.3	Capa de presentación	43
3.3.1	Páginas ASP	43
3.3.1.1	NetCheck.master.....	43
3.3.1.2	customer	43
3.3.1.3	installations_firmware	43
3.3.1.4	installations_parameters	43
3.3.2	Archivos de recursos	44
3.3.2.1	Css	44
3.3.2.2	images	44

4 CONFIGURACION DE LA APLICACIÓN 45


4.1	Parámetros de la aplicación.....	45
4.2	Conexiones a bases de datos transaccionales	48
4.3	Web service Provisioning.....	48

5 EJECUCIÓN..... 50


	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

6 GLOSARIO..... 51

6.1	Endpoint.....	51
	Información del Web Service Provisioning.	51
6.2	Entidad controladora	51
6.3	Firmware	51
6.4	ICC	51
6.5	IMSI	51
6.6	Módem.....	51
6.7	OTRS.....	51
6.8	Provisioning.....	52
6.9	Robot NET	52
6.10	Simcard	52
6.11	XML	52
6.12	WS.....	52

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

CONTROL DE VERSIONES			
Versión	Autor(es) Ordenados alfabéticamente	Descripción de la versión	Fecha de Elaboración
<i>Versión 1.0</i>	<i>Alejandro Herrera</i>	<i>Documento inicial</i>	<i>2016-10-26</i>

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

1. INTRODUCCIÓN

1.1 OBJETIVO

El objetivo del documento es describir técnicamente el funcionamiento del sistema **Net Check**.

1.2 ALCANCE

Este documento describe las características técnicas de software **Net Check**.

1.3 AUDIENCIA

Este documento está dirigido a personal involucrado en la administración del aplicativo **Net Check**, así:


- Ingenieros Sistemas & Aplicaciones.
- Desarrolladores.
- Ingenieros de soporte 2° y 3° nivel.

1.4 DESCRIPCIÓN GENERAL

NET Check es una aplicación web que presta funciones para el seguimiento y solución de problemas de conectividad de Internet en primera línea.

El funcionamiento en general de la aplicación se describe a continuación:

- La aplicación consulta información de un cliente en particular a través de sistemas como el [ICC](#), [OTRS](#), [ROBOT](#) y [Provisioning](#).
- Al obtener la información de los parámetros de conexión, el sistema guarda estos en una tabla de la base de datos del ROBOT NET. En caso de no ser exitosa la consulta, el sistema capturará el error y lo guardará en la misma tabla.
- Permite reiniciar la conexión de un CPE vinculado a un EPC.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

- Permite la actualización del **firmware** de un cliente específico.
- Genera alarmas que indican fallas a nivel general del CORE.
- Genera alarmas que indican fallas relacionadas a un cliente en específico.
- Valida parámetros de aceptación por perfil para up-grade y down-grade.
- Permite reprocesar eventos del Robot Net.
- Genera un listado con las **IMSI** que no tienen actualizado el firmware.
- Genera un listado con las IMSI que tienen parámetros por debajo de lo establecido por la compañía.

1.4.1 Descripción de Proceso Técnico

El sistema Net Check se compone de los siguientes procesos:

1.4.1.1 Peticiones ICC

Este proceso permite al sistema obtener información del cliente y hardware sobre la base de datos ICC (ORACLE), utilizando el paquete "**PCKDTVNETTST**".

Este proceso se ejecuta cada vez que se realiza una consulta desde el Net Check, ya sea por uno de los siguientes parámetros de búsqueda:


- ICC: Identificador único del subscritor en el sistema ICC.
- Documento: Número de documento del cliente.
- Serial: Número de identificación de un equipo.
- MAC: Dirección MAC de un equipo.

1.4.1.2 Peticiones Provisioning


Net Check genera mensajes **XML** codificados con información contenida que permite invocar métodos específicos en el ProvisioninNapi por medio de web service. Estos mensajes son enviados a través del protocolo HTTP al ProvisioningNAPI, el cual se encarga de procesar la petición y devolver el resultado en otro mensaje XML.

Las siguientes son las peticiones que el Net Check genera, envía y recibe al ProvisioningNAPI:

- Obtener perfil: El XML enviado contiene el método "**Provisioning.read**" y el parámetro "**serial**". Esta petición obtiene devuelta la siguiente información:

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

- Profile: Perfil de navegación.
- internet_access: Si tiene acceso a internet.
- dynamic_field: Código de navegación.
- Obtener parámetros: El XML enviado, contiene el método `"Troubleshooting.get_params"` y el parámetro `"serial"`. Esta petición obtiene los siguientes parámetros de conexión de un CPE:
 - Ip_address: Dirección IP del equipo.
 - Brand: Marca del equipo.
 - Model: Tipo de modelo. Para LTE (Indoor, Outdoor) y para BTB (1Box, 2Box).
 - Rsrp: Nivel de cobertura de la red.
 - Cinr: Nivel de señal a ruido.
 - Rsrq: Calidad de la señal.
 - Tx_power: Poder de transmisión del módem.
 - Ping_rtt: Tiempo de respuesta del Ping.
 - Versión del firmware: Versión del firmware que tiene el módem actualmente.
 - Versión del chipset del firmware: Versión del firmware chipset que tiene el módem actualmente.
 - Bsid: Código de la celda a la que está conectado el módem.
 - Bs_name: Nombre de la celda en la que está conectado el módem.
 - High_occupancy: Valor numérico que determina la ocupación de la celda (medio durante algún momento, medio y alto).
- Obtener firmware disponible: El XML enviado, obtiene el método `"Troubleshooting.get_available_firmwares"` y el parámetro `"serial"`. Esta petición obtiene la siguiente información:
 - Firmware Disponible: La versión del firmware disponible es obtenida a partir del modelo del equipo que se está consultando.
- Reiniciar conexión: El XML enviado, contiene el método `"Troubleshooting.reset_connection"` y el parámetro `"serial"`. Esta petición permite reiniciar la conexión de los CPE con tecnología LTE.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0


- Actualiza firmware: El XML enviado, contiene el método “[Troubleshooting.upload_firmware](#)” y el parámetro “[serial y la versión del firmware](#)”. Esta petición permite actualizar la versión del firmware de un CPE con tecnología LTE.
- Obtener lista firmware desactualizado: El XML enviado, contiene el método “[Troubleshooting.get_installations_outdated_firmware](#)”. Esta petición devuelve las IMSI, la dirección IP y la versión del firmware, de los registros que no tengan el firmware actualizado.
- Obtener lista de IMSI con parámetros no aceptables: El XML enviado, contiene el método “[Troubleshooting.get_installations_bad_conditions](#)”. Esta petición devuelve los registros donde los parámetros SINR y RSRP no cumplan con las políticas de la compañía.
- Obtener el detalle de las IMSI con parámetros no aceptables: El XML enviado, contiene el método “[Troubleshooting.get_installations_bad_conditions_detail](#)”. Esta petición devuelve el detalle de los suscriptores con malas condiciones de conexión.

Para ampliar esta información, consulte el manual técnico del Provisioning.

1.4.1.3 *Peticiones Robot Net*

Este proceso permite al Net Check acceder a la base de datos del Robot Net, permitiendo registrar y consultar información de la siguiente manera:

- Consulta de eventos: Esta consulta se hace a través del procedimiento almacenado “[stpUpdHistoricoProvisioning](#)”, obteniendo la siguiente información:
 - Cod Historico: Campo que es la llave primaria, identifica el registro de evento de manera única.
 - Evento: Tipo de evento ver 2.3.1 en ([Robot Internet MANUAL TECNICO.docx](#)).
 - Fecha creación: Fecha de creación del registro en la tabla tbHistorico.
 - Fecha procesado: Fecha y hora en que se finaliza el procesamiento del evento por parte de la aplicación Robot Internet.
 - Estado: Estado del evento (En espera, Procesado exitoso, Procesado con error, Procesando)
 - Respuesta: XML de la respuesta del [WS](#) Provisioning.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0


- Reprocesar eventos: Este proceso se realiza a través del procedimiento almacenado “[stpUpdHistoricoProvisioning](#)”, y este se encarga de actualizar el “CodEstProcesado = 0”, “FechaInicioProcesado = NULL” y “FechaFinProcesado = NULL”, en las tablas “[tbHistoricoProvisioning](#)” y “[tbHistorico](#)”, con el fin de que el sistema Robot Net pueda reprocesar estos eventos.
- Almacenar histórico de parámetros: Cada vez que se consulte información de un suscriptor en el sistema, el Net Check guardara esa información en la tabla “[tbHistoricoConsultaParametro](#)”. Esto se hace por medio del procedimiento almacenado “[stpUpdHistoricoConsultaParametros](#)” en la base de datos del Robot Net. En caso de que la consulta no sea exitosa, el sistema guardara de igual modo un registro con el usuario, fecha y código de error obtenido, el cual indica el motivo por el cual no se obtuvo dicha información.
- Consulta histórico de parámetros: Consulta los registros históricos de parámetros de un suscriptor en particular en la tabla “[tbHistoricoConsultaParametro](#)”. Esto se hace por medio del procedimiento almacenado “[stpUpdHistoricoConsultaParametros](#)” en la base de datos del Robot Net.
- Consulta topológica de sitios: Para esta consulta se toma el código de la celda que se obtuvo previamente en la búsqueda de la consulta de parámetros, en caso de no ser alcanzado el CPE, se consultará la última celda obtenida del histórico de consulta de parámetros. Cuando el sistema no pueda alcanzar el CPE y no existan registros en el histórico de parámetros, el sistema consultará las fallas asociadas a la celda de instalación. Esta celda es enviada como parámetro al procedimiento almacenado “[stpSitios](#)”, quien se encarga de buscar todas las celdas que pertenezcan topológicamente a ella.

1.4.1.4 *Peticiones OTRS*

Este proceso le permite al Net Check consultar tickets sobre la base de datos OTRS, utilizando el procedimiento almacenado “[stpQryTickets](#)”.

Los siguientes son los tipos de tickets que son consultados:

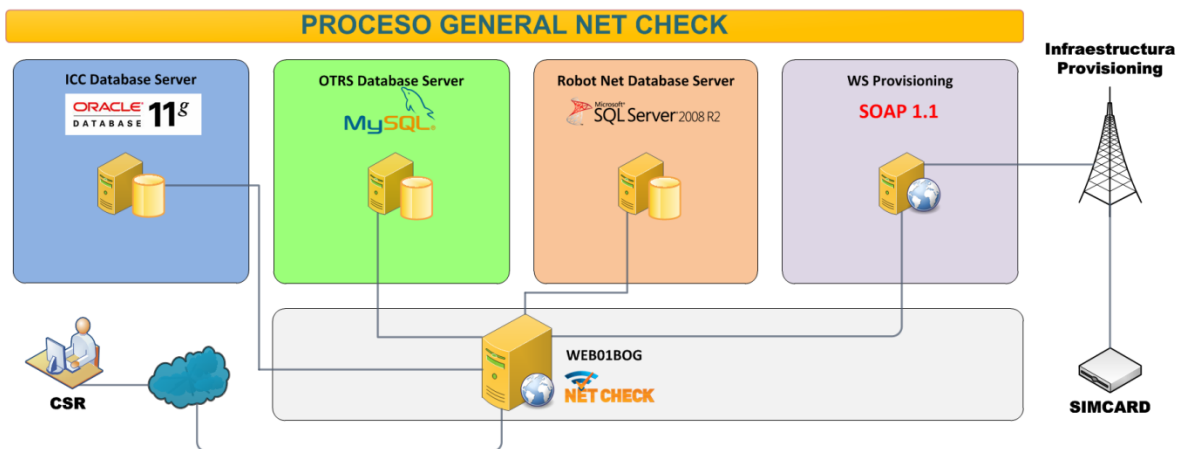
- Tickets tecnología LTE: Fallas reportadas asociadas a una celda específica, la cual es tomada de la consulta de parámetros, en caso de no ser alcanzado el CPE, se

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

consultará la última celda obtenida del histórico de consulta de parámetros. Cuando el sistema no pueda alcanzar el CPE y no existan registros en el histórico de parámetros, el sistema consultará las fallas asociadas a la celda de instalación.

- Tickets tecnología BTB: Fallas reportadas asociadas a un edificio específico. Este edificio es tomado de la consulta de parámetros.
- Tickets a nivel de CORE: Estas fallas no están asociadas a las celdas ni a los edificios a donde se conectan los CPE, estas fallas son reportadas a nivel general del CORE.

A continuación se presenta diagrama que ilustra el funcionamiento de manera general.




1.4.2 Despliegue

En la actualidad (Noviembre de 2016) la aplicación Net Check se encuentra instalada en el servidor WEB01BOG. La dirección URL para ingresar es:

<http://172.21.142.22/TSTNetCSR/>

Para el despliegue de la aplicación es necesario contar con los siguientes recursos de software:

- Oracle Instant Client 32 Bits
- Microsoft .Net Framework 4.5

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

- Internet Information Services (IIS7).

1.5 ARQUITECTURA GENERAL

El sistema **NETCHECK** se desarrolló con la siguiente Arquitectura:

- Herramienta de Desarrollo y lenguaje de Programación: Microsoft VB .NET utilizando el Framework 4.0 de Microsoft .NET.
- Arquitectura de Aplicación: Utiliza modelo de tres capas: capa de negocio, capa de acceso a datos y la capa de presentación (aplicación Web). Para el acceso a datos se utiliza las librerías de Enterprise Library.

2 BASE DE DATOS

La aplicación Net Check no utiliza un motor de datos propio para el sistema, sin embargo este puede acceder a los siguientes motores de base de datos:

- Oracle Database 11g: Corresponde al sistema ICC.
- Microsoft SQL Server 2008 R2: Corresponde al sistema Robot Net.
- MySQL 5.1.73: Corresponde al sistema OTRS.


Para ampliar esta información, dirigirse a la respectiva documentación de cada sistema.

2.1 PROCEDIMIENTOS ALMACENADOS

Net Check utiliza procedimientos almacenados para interactuar con las base de datos de cada sistema mencionado anteriormente. A continuación se describen cada uno de ellos:

2.1.1 ICC

La base de datos ICC (SERVER) Oracle Database 11g, contiene el paquete "**PCKDTVNETTST**", el cual fue creado para consultar los productos de internet, TV e información del hardware relacionado. Este paquete contiene los siguientes procedimientos y funciones:

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

2.1.1.1 PCKDTVNETTST.stpQryInformacionCliente

Este procedimiento permite consultar la información del cliente teniendo en cuenta los criterios de búsqueda.

A continuación se presenta una descripción de cada uno de los parámetros:

NOMBRE	TIPO DE DATO	TIPO PARÁMETRO	DESCRIPCIÓN
strReferenciaBusqueda	VARCHAR2	Entrada	Referencia o valor de búsqueda.
strTipoReferencia	NUMBER	Entrada	Criterio de búsqueda. 1. ICC 2. Cédula 3. Simcard 4. MAC
cur_OUT	SYS_REFCURSOR	Salida	Resultado de la búsqueda.


A continuación se describen los criterios de búsqueda:

- Id del cliente (ICC): El procedimiento buscara la información del cliente directamente con el id del cliente.
- Cedula: El procedimiento utilizara la función “[fnObtenerIBSPorDocumento](#)”, para obtener el id del cliente (ICC) que tiene asociado con la cedula y así poder realizar la consulta.
- Serial del dispositivo (Serial): El procedimiento utiliza la función “[fnObtenerIBSPorDevice](#)”, para obtener el id del cliente que tiene asociado el serial del dispositivo, de esta manera poder realizar la consulta.
- MAC: El procedimiento utiliza la función “[fnObtenerIBSPorDevice](#)”, para obtener el id del cliente que tiene asociado a la MAC, de esta manera poder realizar la consulta.

2.1.1.2 stpQryInformacionHardware

Este procedimiento permite consultar la información del hardware.

A continuación se presenta una descripción de cada uno de los parámetros:

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

NOMBRE	TIPO DE DATO	TIPO PARÁMETRO	DESCRIPCIÓN
strReferenciaBusqueda	VARCHAR2	Entrada	Referencia o valor de búsqueda.
intTipoReferencia	NUMBER	Entrada	Criterio de búsqueda. 1. ICC 2. SerialDevice 3. MAC
cur_OUT	SYS_REFCURSOR	Salida	Resultado de la búsqueda.

A continuación se describen los criterios de búsqueda:

- Id del cliente (ICC): El procedimiento buscará la información del hardware asociado directamente al id del cliente.
- Serial del dispositivo (SerialDevice): El procedimiento buscará la información del hardware asociado directamente al SerialDevice.
- MAC: El procedimiento buscará la información del hardware asociado directamente a la MAC del dispositivo.

2.1.1.3 *fnObtenerIBSPorDevice*

Esta función se encarga de obtener el id del cliente (ICC) a través del serial o MAC del equipo que tiene asociado en el sistema.

2.1.1.4 *fnObtenerIBSPorDocumento*


Esta función se encarga de obtener el id del cliente (ICC) a través del documento de identidad que se encuentra registrado en el sistema.

2.1.2 **Robot Net**

En la base de datos de Robot Net (SLQ0BOG) SQL Server 2008 R2, se han desarrollado los siguientes procedimientos almacenados:

2.1.2.1 *stpUpdHistoricoProvisioning*

Este procedimiento almacenado se encarga de consultar y re-procesar eventos desde el Net Check. Este también realiza el mantenimiento de la tabla "**tbHistoricoProvisioning**". Para ampliar esta información, dirigirse al manual técnico [Robot Net](#).

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0


A continuación se presenta una descripción de cada uno de los parámetros que utiliza Net Check:

NOMBRE	TIPO DE DATO	TIPO PARÁMETRO	DESCRIPCIÓN
@CodHistoricoProvisioning	INT	Entrada	Consecutivo de identificación del registro en la tabla. Identifica el registro de manera única.
@IMSI	VARCHAR(15)	Entrada	Serial de la Simcard .
@IMSiNuevo	VARCHAR(15)	Entrada	Serial de la nueva Simcard.
@Perfil	NVARCHAR(25)	Entrada	Velocidad del producto de Internet.
@CodigoPostal	VARCHAR(5)	Entrada	Identifica la URL del WS de Provisioning a la cual debe ser enviado el evento.
@CodEstProcesado	SMALLINT	Entrada	Código de estado del procesamiento del evento.
@FechaInicioProcesado	DATETIME	Entrada	Fecha y hora de inicio del procesamiento del evento.
@FechaFinProcesado	DATETIME	Entrada	Fecha y hora de finalización del procesamiento del evento.
@CodCliente	INT	Entrada	Código del cliente.
@tyOperacion	TINYINT	Entrada	Tipo de operación: 4. Consulta de registros. 5. Reprocesar evento

2.1.2.2 *stpUpdLog*

Este procedimiento almacenado permite consultar la tabla “**Log**”. Para ampliar esta información, dirigirse al manual técnico [Robot Net](#).

A continuación se presenta una descripción de cada uno de los parámetros que utiliza Net Check:

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

NOMBRE	TIPO DE DATO	TIPO PARÁMETRO	DESCRIPCIÓN
@ID	INT	Entrada	Consecutivo de identificación del registro en la tabla. Identifica el registro de manera única.
@Level	VARCHAR(50)	Entrada	Tipo de log (ERROR, INFO).
@Logger	VARCHAR(255)	Entrada	Aplicación que genera el log. En este momento podría ser el sistema Robot Internet o el sistema Listener Internet.
@ObjectIdentity	INT	Entrada	Hace referencia al histórico que genera el log.
@tyOperacion	TINYINT	Entrada	Tipo de operación: 4. Consulta de registros.

2.1.2.3 *stpSitios*


Este procedimiento almacenado fue creado para que Net Check identifique los sitios afectados topológicamente a partir de un sitio padre.

A continuación se presenta una descripción de cada uno de los parámetros que utiliza Net Check:

NOMBRE	TIPO DE DATO	TIPO PARÁMETRO	DESCRIPCIÓN
@Sitio	CHAR(10)	Entrada	Referencia de búsqueda (Sitio Padre).
@tyOperacion	TINYINT	Entrada	Tipo de operación: 1. Consulta de sitios topologicamente. 2. Consultar Sitio especifico.
@Result	VARCHAR(MAX)	Salida	Resultado de la búsqueda..

2.1.2.4 *ctrHistoricoConsultaParametro*


Este procedimiento almacenado fue creado para que Net Check consulte y guarde los parámetros de conexión del CPE, en la tabla "**tbHistoricoConsultaParametro**". También es

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

usado para consultar el nivel de advertencia de acuerdo a los parámetros RSRP y SINR, consultar la última celda guardada en el histórico de parámetros y consultar la tabla de validación Up grade y Down grade.

A continuación se presenta una descripción de cada uno de los parámetros que utiliza Net Check:

NOMBRE	TIPO DE DATO	TIPO PARÁMETRO	DESCRIPCIÓN
@CodHistoricoConsultaP	INT	Entrada	Consecutivo de identificación del registro en la tabla. Identifica el registro de manera única.
@Imsi	VARCHAR(15)	Entrada	Serial de la Simcard.
@Icc	VARCHAR(50)	Entrada	Código del Cliente. Es generado por el sistema IBS.
@Usuario	VARCHAR(50)	Entrada	Usuario de Windows, con el que se accedio y se consulto los parametros.
@VersionFirmware	VARCHAR(50)	Entrada	Versión del firmware del modem .
@FirmwareChipsetVersion	VARCHAR(50)	Entrada	Versión del chipset del firmware del modem.
@Bsid	VARCHAR(20)	Entrada	Codigo de la celda a la que está conectado el módem.
@DireccionIp	VARCHAR(50)	Entrada	Dirección IP del módem.
@PingRtt	VARCHAR(50)	Entrada	Time to Life del Ping.
@TxPower	VARCHAR(50)	Entrada	Poder de transmisión del módem.
@Rsrp	VARCHAR(50)	Entrada	Nivel de cobertura de la red LTE DIRECTV.
@Sinr	VARCHAR(50)	Entrada	Nivel de señal a ruido.
@Rsrq	VARCHAR(50)	Entrada	Calidad de la señal LTE DIRECTV.
@CodError	INT	Entrada	Codigo del error, si se llegara a generar en el momento de hacer la consulta.
@ReferenciaBusqueda	VARCHAR(20)	Entrada	Referencia o valor de búsqueda.
@Perfil	VARCHAR(50)	Entrada	Velocidad del producto de Internet.
@tyOperacion	TINYINT	Entrada	Criterio de búsqueda. 1. Consultar parámetros. 2. Grabar parámetros.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

			3. Obtener advertencia (RSRP y SINR). 4. Consultar celda desde el historico. 5. Consultar Up grade y Down grade por perfil.
@timeQry	INT	Entrada	Número de días maximo. Parámetro que indica que se consultaran los registros guardados con una fecha menor a esta.
@Ctg1	VARCHAR(2)	Entrada	Categoría (Instalación y Mudanza).
@Ctg2	VARCHAR(2)	Entrada	Categoría (Tolerancia).
@SpeedProfileCode	VARCHAR(50)	Entrada	Codigo del perfil de velocidad.
@error_message	VARCHAR(255)	Entrada	Error generado cuando se obtiene un valor inesperado en cualquiera de los parámetros consultados.

2.1.3 OTRS


En la base de datos OTRS (BD_OTRS) MySQL, se ha desarrollado el siguiente procedimiento almacenado:

2.1.3.1 stpQryTickets

Este procedimiento almacenado, se encarga de consultar tickets para los equipos con tecnología LTE, BTB y tickets de fallas en general a nivel del CORE.

A continuación se presenta una descripción de cada uno de los parámetros que utiliza Net Check:

NOMBRE	TIPO DE DATO	TIPO PARÁMETRO	DESCRIPCIÓN
@strReferenciaBusqueda	VARCHAR(100)	Entrada	Referencia o valor de búsqueda.
@intTimeQryTicket	INTEGER	Entrada	Número de días maximo. Parámetro que indica que se consultaran los registros que tengan una fecha de creación del ticket menor a esta.
@intTipoReferencia	INTEGER	Entrada	Criterio de búsqueda. 1. Consultar tickets con tecnología LTE. 2. Consultar tickets tickets con fallas a nivel del CORE. 3. Consultar tickets con tecnología BTB.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3 ARQUITECTURA DE APLICACION

El proyecto utiliza el concepto de “Entidad Controladora”.

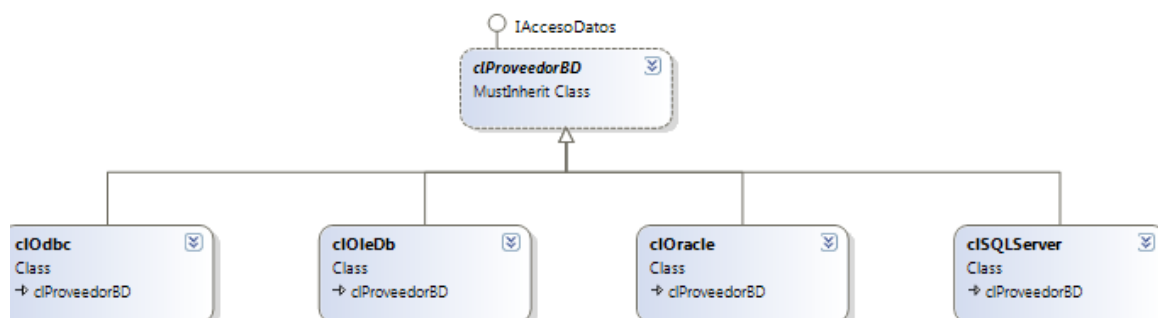
El proyecto utiliza arquitectura de tres capas, las cuales se describen a continuación:

3.1 CAPA DE ACCESO A DATOS

El proyecto hace uso de la librería DTVCol.RobotInternet.DataAccess para interactuar con el repositorio. Para la conexión con la base de datos se utiliza Enterprise Library.


El proyecto tiene una interfaz de acceso a datos llamada IAccesoDatos la cual ejecuta Sentencias SQL, Store Procedures, permite conectarse, pasar parámetros y retorna conjuntos de datos a través de Datasets.

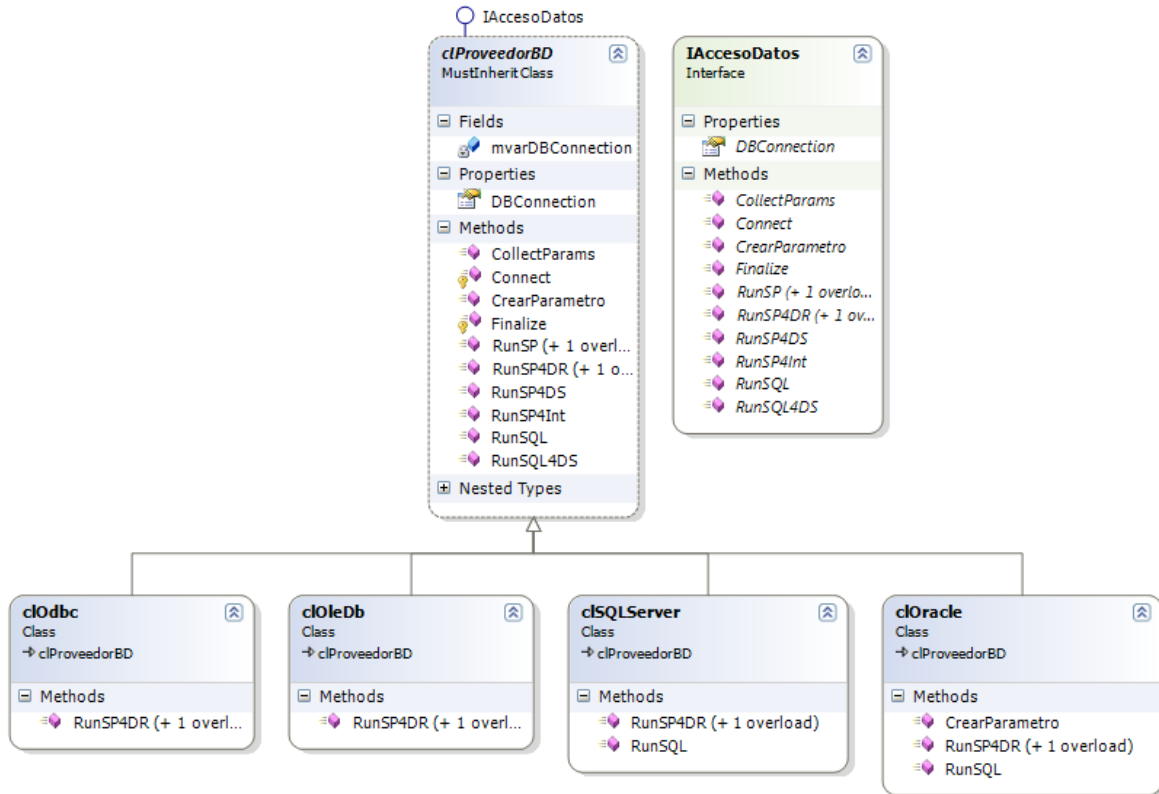
A continuación se presenta diagrama general del acceso a datos.



3.1.1 Clase CProveedorBD

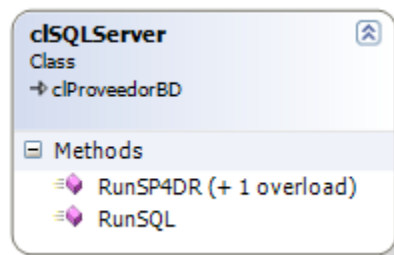
Es la clase base a partir de la cual se realiza el acceso a la base de datos de la aplicación ICC, Robot Net y OTRS. De esta clase se deriva la clase clSqlServer y clOracle, clases utilizadas para el motor específico que se usa en la aplicación.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0




3.1.2 Clase *clSqlServer*

Clase derivada de *clProveedorDb*, contiene los métodos para realizar las operaciones de acceso a la base de datos.



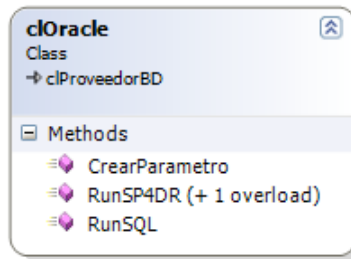
Métodos

- *RunSP4DR*: Ejecuta Store Procedure sobre la base de datos.
- *RunSQL*: Ejecuta Sentencia SQL sobre la base de datos.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.1.3 Clase *clOracle*

Clase derivada de *clProveedorDb*, contiene los métodos para realizar las operaciones de acceso a la base de datos Oracle.

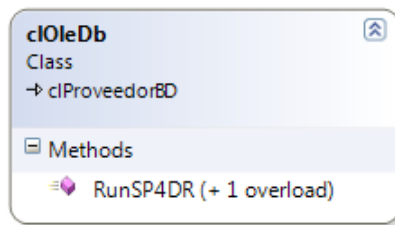


Métodos

- **CrearParametro:** Crea un parámetro con el nombre, tipo, valor y dirección.
- **RunSP4DR:** Ejecuta Store Procedure sobre la base de datos.
- **RunSQL:** Ejecuta Sentencia SQL sobre la base de datos.

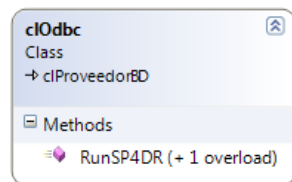
3.1.4 Clase *clOleDb*


No se utiliza para Net Check.



3.1.5 Clase *clOdbc*

No se utiliza para Net Check.



	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.2 CAPA DE NEGOCIO LIBDTVCOLNETTROUBLESHOOTING

Esta librería contiene las clases que manejan la lógica del negocio del software aplicada a cada parte del proyecto el cual está compuesto de las siguientes subcapas:

- **icc:** Contiene las clases entidades y controladoras que permiten consultar y operar la información con el sistema ICC.
- **Otrs:** Contiene las clases entidades y controladoras que permiten consultar y operar la información con el sistema OTRS.
- **Robot:** Contiene las clases entidades y controladoras que permiten consultar y operar la información con el sistema Robot.
- **Troubleshooting:** Contiene las clases que permiten consultar y operar la información con el sistema ICC por medio de web service.

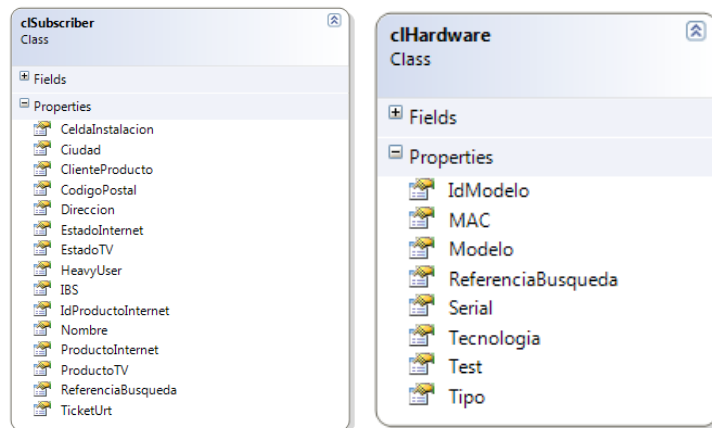
En esta librería se implementa el concepto de “Entidad Controladora” mediante el cual se tienen entidades que realizan mapeos hacia las tablas de la base de datos.


3.2.1 Capa de negocio ICC

A continuación se definen las clases entidades y las clases controladoras.

3.2.1.1 Clases entidades

Las clases entidades que hacen parte de la comunicación con el sistema ICC, son las siguientes:



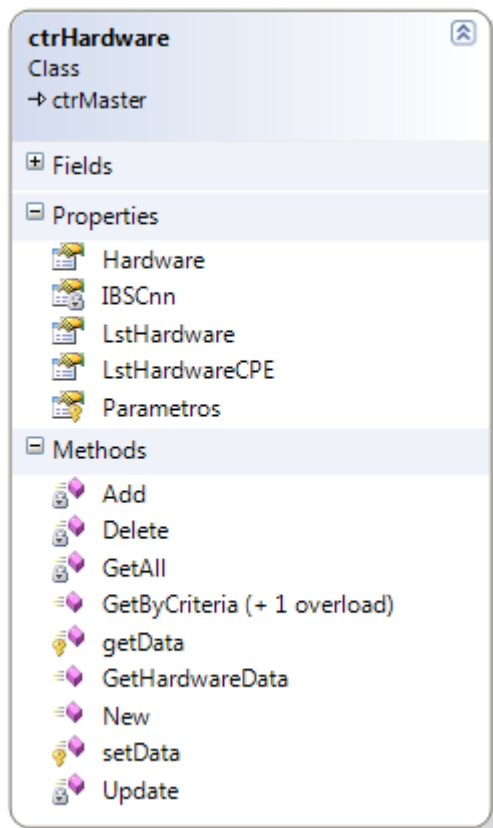
	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

Estas clases contiene los atributos y propiedades del procedimiento almacenado desarrollado en la base de datos del sistema ICC.

3.2.1.2 Clases controladoras


Las clases controladores que hacen parte de la comunicación con el sistema ICC, son las siguientes:

3.2.1.2.1 ctrHardware



Constructor

- Asigna la propiedad `strIBSCnn` en valor recibido en el constructor.
- Asigna la propiedad `strNombreStp` el valor `"PCKDTVNETTST.stpQryInformacion Cliente"`.
- Asigna la propiedad `lstModelosTestLTE` el valor parametrizado en el key `"lstModelosTestLTE"` del archivo de configuración `web.config`.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0


- Asigna la propiedad **IstModelosTestBTB** el valor parametrizado en el key "**IstModelosTestBTB**" del archivo de configuración web.config.
- Asigna la propiedad **IstModelosIndoor** el valor parametrizado en el key "**IstModelosIndoor**" del archivo de configuración web.config.
- Asigna la propiedad **IstModelosOutdoor** el valor parametrizado en el key "**IstModelosOutdoor**" del archivo de configuración web.config.
- Asigna la propiedad **IstModelosNA** el valor parametrizado en el key "**IstModelosNA**" del archivo de configuración web.config.

Propiedades

- **IBSCnn:** Devuelve el valor asignado por el constructor.
- **Parametros:** Recibe Tipo de Operación, en esta propiedad se realiza el mapeo de parámetros para ejecutar el procedimiento almacenado en la base de datos "**PCKDTVNETTST.stpQryInformacionCliente**". Para realizar la creación de parámetros llama al método "CrearParametro" de la clase **clProveedorDb**.
- **IstHardware:** Es de tipo List(Of **clHardware**), mediante esta propiedad se retorna el resultado de ejecutar el procedimiento almacenado "**PCKDTVNETTST.stpQryInformacionCliente**" con la información ordenada.
- **IstHardwareCPE:** Es de tipo List(Of **clHardware**), mediante esta propiedad se retorna el resultado de ejecutar el procedimiento almacenado "**PCKDTVNETTST.stpQryInformacionCliente**" con la información ordenada.
- **Hardware:** Es de tipo **clHardware**, mediante esta propiedad se realiza el mapeo a los parámetros del procedimiento almacenado "**PCKDTVNETTST.stpQryInformacionCliente**".

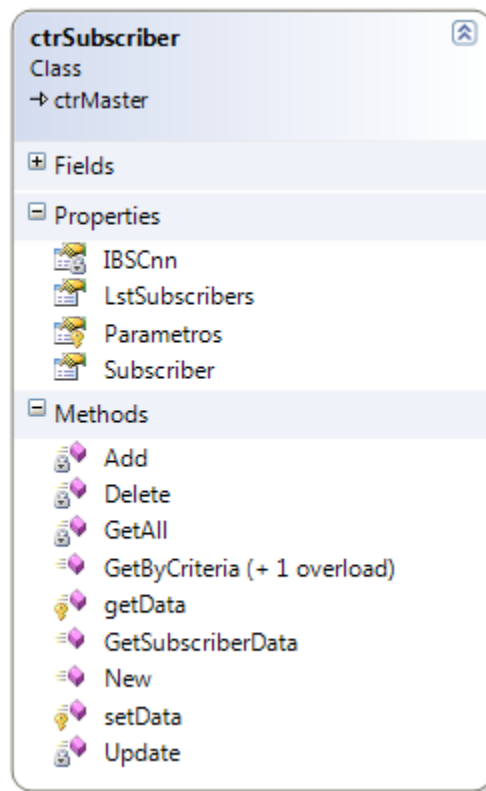
Métodos

- **GetHardwareData:** Retorna instancia de objeto de tipo **clHardware**, para ello:
 - Ejecuta el método **GetByCriteria** el cual retorna un **DataSet**.
 - A partir del **DataSet** se realiza el mapeo (cargue) del objeto de la entidad correspondiente **clHardware** para el caso.
- **GetByCriteria:** Tiene como parámetro un número (tipo de operación) para ejecutar una operación del Store Procedure, retorna **DataSet**.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0


- **setData:** Escribe sobre la base de datos. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.
- **getData:** Ejecuta store procedure con la colección de parámetros que se ha asignado. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.

3.2.1.2.2 ctrSuscriber



Constructor

- Asigna la propiedad `strIBSCnn` en valor recibido en el constructor.
- Asigna la propiedad `strNombreStp` el valor `"PCKDTVNETTST.stpQryInformacionCliente"`.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0


- Asigna la propiedad `intComProductCatInternet` el valor parametrizado en el key “`intComProductCatInternet`” del archivo de configuración web.config.
- Asigna la propiedad `intComProductCatBasico` el valor parametrizado en el key “`intComProductCatBasico`” del archivo de configuración web.config.
- Asigna la propiedad `intComProductCatHardware` el valor parametrizado en el key

Propiedades

- **IBSCnn:** Devuelve el valor asignado por el constructor.
- **Subscriber:** Es de tipo `clSubscriberCriteria`, mediante esta propiedad se realiza el mapeo a los parámetros del procedimiento almacenado “`PCKDTVNETTST.stpQryInformacionCliente`”.
- **Parametros:** Recibe Tipo de Operación, en esta propiedad se realiza el mapeo de parámetros para ejecutar el procedimiento almacenado “`PCKDTVNETTST.stpQryInformacionCliente`”. Para realizar la creación de parámetros llama al método “`CrearParametro`” de la clase `clProveedorDb`.
- **LstSubscribers:** Es de tipo `List(Of clSubscriberCriteria)`, mediante esta propiedad se retorna el resultado de ejecutar el procedimiento almacenado “`PCKDTVNETTST.stpQryInformacionCliente`” con la información ordenada.

Métodos

- **GetSubscriberData:** Retorna instancia de objeto de tipo `clSubscriber`, para ello:
 - Ejecuta el método `GetByCriteria` el cual retorna un `DataSet`.
 - A partir del `DataSet` se realiza el mapeo (cargue) del objeto de la entidad correspondiente `clSubscriber` para el caso.
- **GetByCriteria:** Tiene como parámetro un número (tipo de operación) para ejecutar una operación del Store Procedure, retorna `DataSet`.
- **setData:** Escribe sobre la base de datos. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.
- **getData:** Ejecuta store procedure con la colección de parámetros que se ha asignado. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.

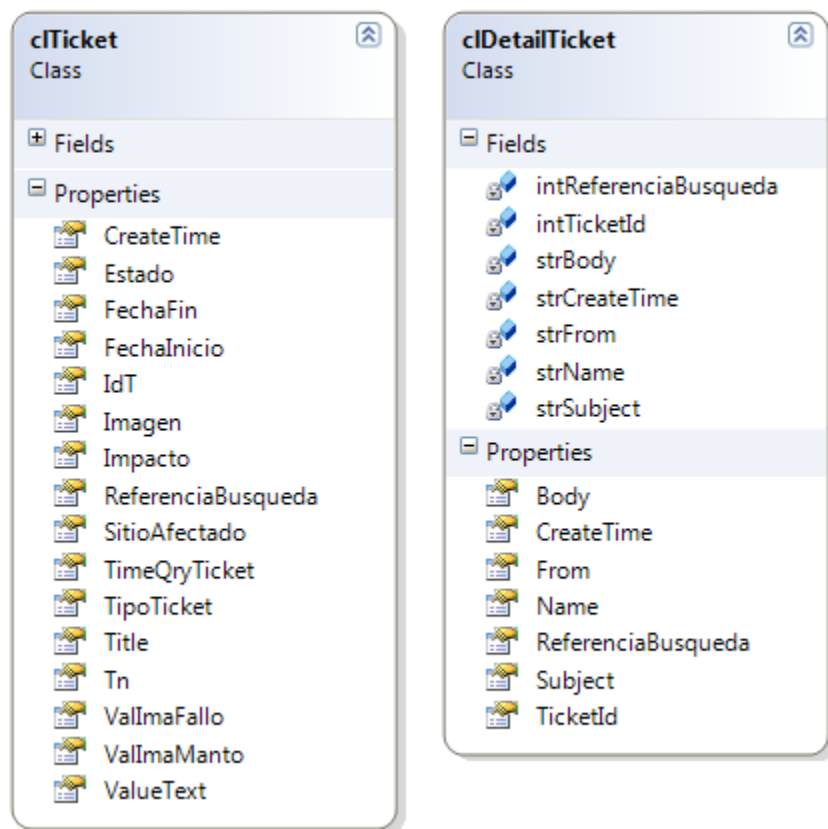
	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.2.2 *Capa de negocio OTRS*

A continuación se definen las clases entidades y las clases controladoras.

3.2.2.1 *Clases Entidades*


Las clases entidades que hacen parte de la comunicación con la base de datos OTRS, son las siguientes:



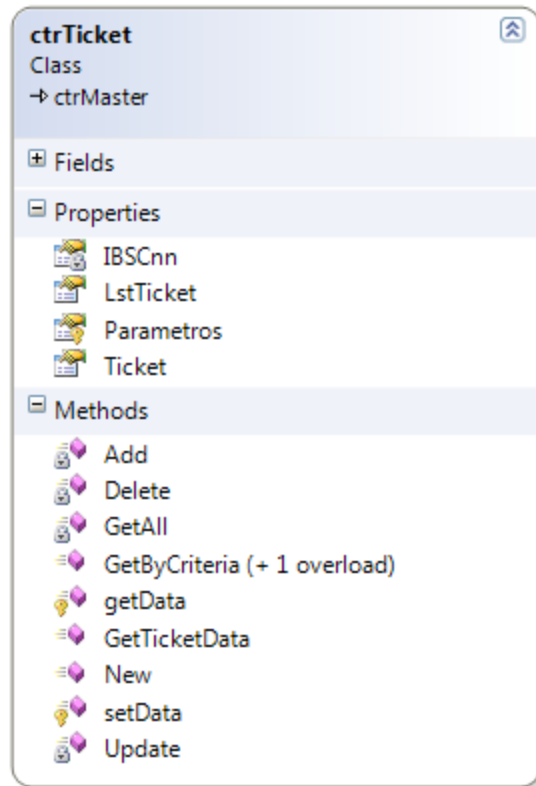
Estas clases contiene los atributos y propiedades del procedimiento almacenado desarrollado en la base de datos OTRS.

3.2.2.2 *Clases Controladoras*

Las clases controladores que hacen parte de la comunicación con el sistema OTRS, son las siguientes:

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.2.2.2.1 ctrTicket




Constructor

- Asigna la propiedad `strIBSCnn` en valor recibido en el constructor.
- Asigna la propiedad `strNombreStp` el valor `"stpQryTickets"`.

Propiedades

- **IBSCnn:** Devuelve el valor asignado por el constructor.
- **Parametros:** Recibe Tipo de Operación, en esta propiedad se realiza el mapeo de parámetros para ejecutar el procedimiento almacenado en la base de datos `"stpQryTickets"`. Para realizar la creación de parámetros llama al método `"CrearParametro"` de la clase `clProveedorDb`.
- **LstTicket:** Es de tipo `List(Of clTicket)`, mediante esta propiedad se retorna el resultado de ejecutar el procedimiento almacenado `"stpQryTickets"` con la información ordenada.

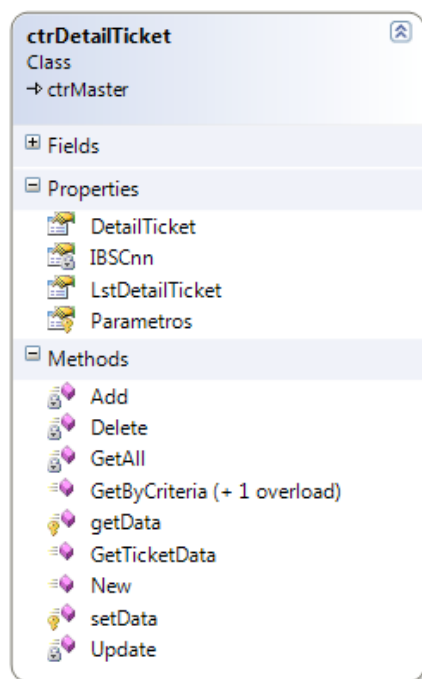
	Nombre Documento	Versión
	Net Check Manual Técnico	1.0


- **Ticket:** Es de tipo cITicket, mediante esta propiedad se realiza el mapeo a los parámetros del procedimiento almacenado “**stpQryTickets**”.

Métodos

- **GetTicketData:** Retorna instancia de objeto de tipo cITicket, para ello:
 - Ejecuta el método GetByCriteria el cual retorna un DataSet.
 - A partir del DataSet se realiza el mapeo (cargue) del objeto de la entidad correspondiente cITicket para el caso.
- **GetByCriteria:** Tiene como parámetro un número (tipo de operación) para ejecutar una operación del Store Procedure, retorna DataSet.
- **setData:** Escribe sobre la base de datos. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.
- **getData:** Ejecuta store procedure con la colección de parámetros que se ha asignado. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.

3.2.2.2.2 ctrDetailTicket



	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

Constructor


- Asigna la propiedad **strIBSCnn** en valor recibido en el constructor.
- Asigna la propiedad **strNombreStp** el valor “**stpQryTickets**”.

Propiedades

- **IBSCnn**: Devuelve el valor asignado por el constructor.
- **Parametros**: Recibe Tipo de Operación, en esta propiedad se realiza el mapeo de parámetros para ejecutar el procedimiento almacenado en la base de datos “**stpQryTickets**”. Para realizar la creación de parámetros llama al método “CrearParametro” de la clase **clProveedorDb**.
- **LstDetailTicket**: Es de tipo List(Of **clDetailTicket**), mediante esta propiedad se retorna el resultado de ejecutar el procedimiento almacenado “**stpQryTickets**” con la información ordenada.
- **DetailTicket**: Es de tipo **clDetailTicket()**, mediante esta propiedad se realiza el mapeo a los parámetros del procedimiento almacenado “**stpQryTickets**”.

Métodos

- **GetTicketData**: Retorna instancia de objeto de tipo **clDetailTicket**, para ello:
 - Ejecuta el método **GetByCriteria** el cual retorna un **DataSet**.
 - A partir del **DataSet** se realiza el mapeo (cargue) del objeto de la entidad correspondiente **clDetailTicket** para el caso.
- **GetByCriteria**: Tiene como parámetro un número (tipo de operación) para ejecutar una operación del Store Procedure, retorna **DataSet**.
- **setData**: Escribe sobre la base de datos. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.
- **getData**: Ejecuta store procedure con la colección de parámetros que se ha asignado. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.2.3 Capa de negocio ROBOT

A continuación se definen las clases entidades y las clases controladoras que permiten comunicación con el sistema Robot Net.

3.2.3.1 Clases Entidades

Las clases entidades que hacen parte de la comunicación con el sistema Robot, son las siguientes:

clHistoricoConsulta...
Class

Fields

Properties

Bsid

Bsid_name

Categoria1

Categoria2

CodDescripcion

CodError

CodHistoricoConsulta

ColorRsrp

ColorSinr

DireccionIp

Fecha

FirmwareChipsetVer...

Icc

Imsi

Ocupacion

Perfil

PingRtt

ReferenciaBusqueda

Rsrp

Rsrq

Sinr

SpeedProfileCode

TimeQry

TxPower

Usuario

ValorColorGeneral

VersionFirmware

clHistoricoProvision...
Class

Properties

CantidadIntentos

CodCliente

CodEstProcesado

CodEvento

CodEvento2

CodHistoricoProvisi...

CodigoPostal

CodRazon

CodRespuestaWS

Estado

FechaCreacion

FechaFinProcesado

FechaInicioProcesado

IMSI

IMSINuevo

Latitud

Longitud

NumeroDiasHistorico

Perfil

URLProvisioning

XMLRespuestaWS

Methods

New

clLog
Class

Properties

DateEvento

Exception

Id

Level

Logger

Message

ObjectIdentity

Thread

Methods

New

clIMapHistoricoProvi...
Class

Properties

CodigoPostal

Tecnologia

URLProvisioning


clSitiosTopologia
Class

Fields

Properties

Celda

Result

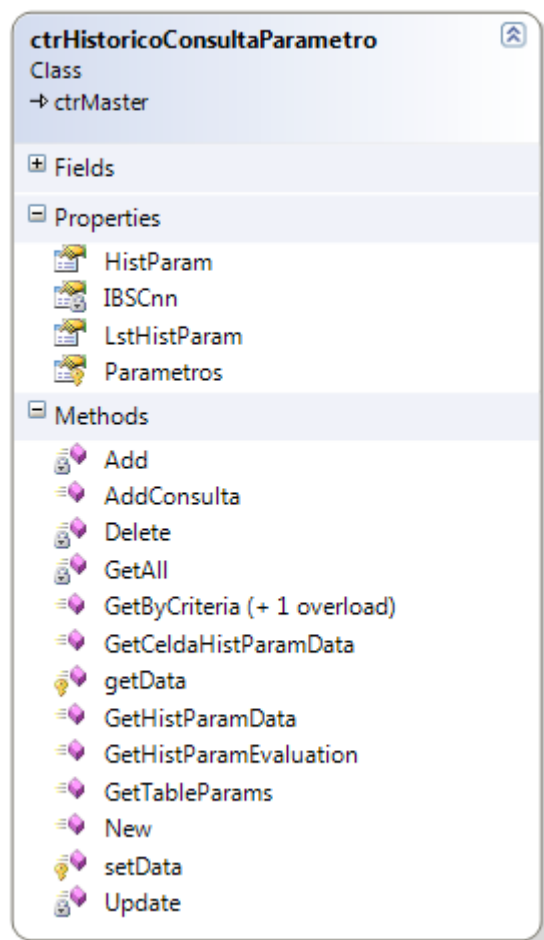
	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

Estas clases contiene los atributos y propiedades de los procedimientos almacenados desarrollados en la base de datos del sistema Robot.

3.2.3.2 Clases Controladoras


Las clases controladores que hacen parte de la comunicación con el sistema ICC, son las siguientes:

3.2.3.2.1 ctrHistoricoConsultaParametro



Constructor

- Asigna la propiedad `strIBSCnn` en valor recibido en el constructor.
- Asigna la propiedad `strNombreStp` el valor "`stpUpdHistoricoConsultaParametro`".


	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

Propiedades

- **Parametros:** Recibe Tipo de Operación, en esta propiedad se realiza el mapeo de parámetros para ejecutar el procedimiento almacenado en la base de datos “stpUpdHistoricoConsultaParametro”. Para realizar la creación de parámetros llama al método “CrearParametro” de la clase [clProveedorDb](#).
- **LstHistParam:** Es de tipo List(Of clHistoricoConsultaParametro), mediante esta propiedad se retorna el resultado de ejecutar el procedimiento almacenado “stpUpdHistoricoConsultaParametro” con la información ordenada.
- **HistParam:** Es de tipo clHistoricoConsultaParametro, mediante esta propiedad se realiza el mapeo a los parámetros del procedimiento almacenado “stpUpdHistoricoConsultaParametro”.
- **IBSCnn:** Devuelve el valor asignado por el constructor.

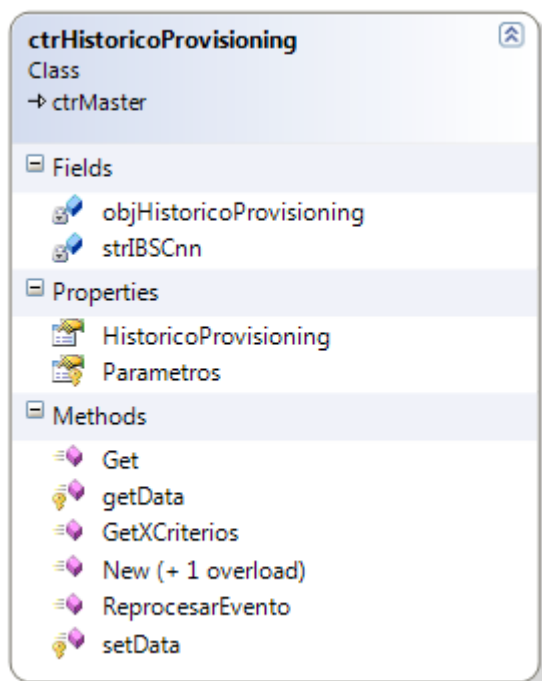
Métodos

- **GetHistParamData:** Retorna instancia de objeto de tipo clHistoricoConsultaParametro, para ello:
 - Ejecuta el método GetByCriteria el cual retorna un DataSet.
 - A partir del DataSet se realiza el mapeo (cargue) del objeto de la entidad correspondiente clHistoricoConsultaParametro para el caso.
- **GetCeldaHistParamData:** Retorna instancia de objeto de tipo clHistoricoConsultaParametro, para ello:
 - Ejecuta el método GetByCriteria el cual retorna un DataSet.
 - A partir del DataSet se realiza el mapeo (cargue) del objeto de la entidad correspondiente clHistoricoConsultaParametro para el caso.
- **GetHistParamEvaluation:** Retorna instancia de objeto de tipo clHistoricoConsultaParametro, para ello:
 - Ejecuta el método GetByCriteria el cual retorna un DataSet.
 - A partir del DataSet se realiza el mapeo (cargue) del objeto de la entidad correspondiente clHistoricoConsultaParametro para el caso.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0


- **GetByCriteria:** Tiene como parámetro un número (tipo de operación) para ejecutar una operación del Store Procedure, retorna DataSet.
- **setData:** Escribe sobre la base de datos. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.
- **getData:** Ejecuta el procedimiento almacenado con la colección de parámetros que se ha asignado. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.

3.2.3.2.2 ctrHistoricoProvisioning



Constructor

- Asigna la propiedad **strIBSCnn** en valor recibido en el constructor.
- Asigna la propiedad **strNombreStp** el valor "**stpUpdHistoricoProvisioning**".

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

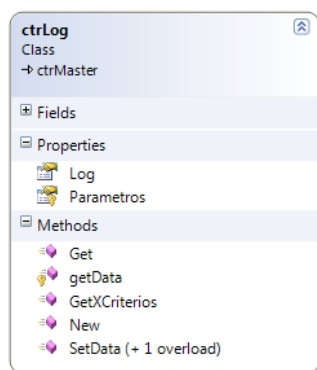
Propiedades


- **Parametros:** Recibe Tipo de Operación, en esta propiedad se realiza el mapeo de parámetros para ejecutar el procedimiento almacenado en la base de datos “**stpUpdHistoricoProvisioning**”. Para realizar la creación de parámetros llama al método “CrearParametro” de la clase **clProveedorDb**.
- **HistoricoProvisioning:** Es de tipo **clHistoricoProvisioning**, mediante esta propiedad se realiza el mapeo a los parámetros del procedimiento almacenado “**stpUpdHistorico Provisioning**”.

Métodos

- **Get:** Retorna instancia de objeto de tipo **clHistoricoProvisioning**, para ello:
 - Ejecuta el método **getData** el cual retorna un **DataSet**.
 - A partir del **DataSet** se realiza el mapeo (cargue) del objeto de la entidad correspondiente **clHistoricoProvisioning** para el caso.
- **GetXCriterios:** Tiene como parámetro un número (tipo de operación) para ejecutar una operación del Store Procedure, retorna **DataSet**.
- **setData:** Escribe sobre la base de datos. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.
- **getData:** Ejecuta el procedimiento almacenado con la colección de parámetros que se ha asignado. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.

3.2.3.2.3 ctrLog



	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

Constructor


- Asigna la propiedad `strNombreStp` el valor “`stpUpdLog`”.

Propiedades

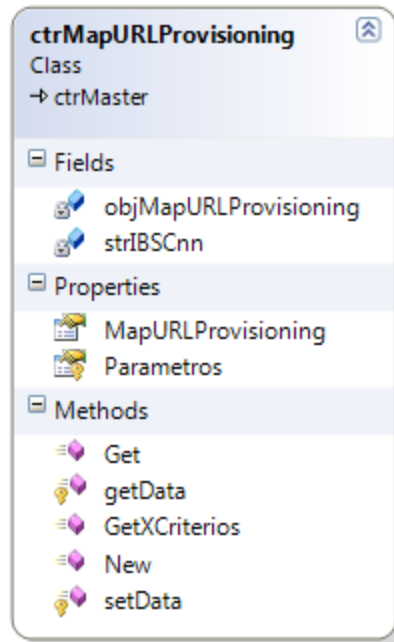
- **Parametros:** Recibe Tipo de Operación, en esta propiedad se realiza el mapeo de parámetros para ejecutar el procedimiento almacenado en la base de datos “`stpUpdLog`”. Para realizar la creación de parámetros llama al método “`CrearParametro`” de la clase `clProveedorDb`.
- **Log:** Es de tipo `clLog`, mediante esta propiedad se realiza el mapeo a los parámetros del procedimiento almacenado “`stpUpdLog`”.

Métodos

- **Get:** Retorna instancia de objeto de tipo `clLog`, para ello:
 - Ejecuta el método `getData` el cual retorna un `DataSet`.
 - A partir del `DataSet` se realiza el mapeo (cargue) del objeto de la entidad correspondiente `clLog` para el caso.
- **GetXCriterios:** Tiene como parámetro un número (tipo de operación) para ejecutar una operación del Store Procedure, retorna `DataSet`.
- **setData:** Escribe sobre la base de datos. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.
- **getData:** Ejecuta el procedimiento almacenado con la colección de parámetros que se ha asignado. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.2.3.2.4 ctrMapURLProvisioning




Constructor

- Asigna la propiedad `strNombreStp` el valor “`stpUpdMapURLProvisioning`”.
- Asigna la propiedad `strIBSCnn` en valor recibido en el constructor.

Propiedades

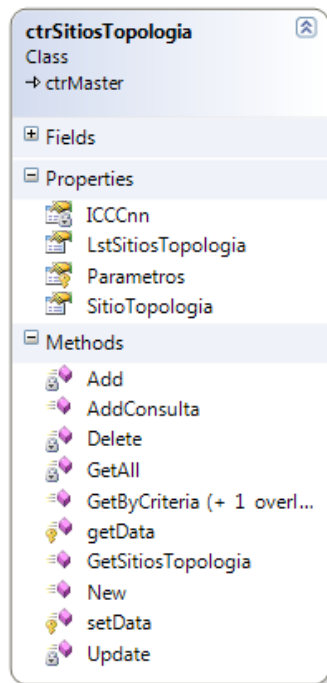
- **MapURLProvisioning:** Es de tipo `clMapHistoricoProvisioning`, mediante esta propiedad se realiza el mapeo a los parámetros del procedimiento almacenado “`stpUpdMapURLProvisioning`”.
- **Parametros:** Recibe Tipo de Operación, en esta propiedad se realiza el mapeo de parámetros para ejecutar el procedimiento almacenado en la base de datos “`stpUpdMapURLProvisioning`”. Para realizar la creación de parámetros llama al método “`CrearParametro`” de la clase `clProveedorDb`.


	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

Métodos

- **Get:** Retorna instancia de objeto de tipo `clMapHistoricoProvisioning`, para ello:
 - Ejecuta el método `getData` el cual retorna un `DataSet`.
 - A partir del `DataSet` se realiza el mapeo (cargue) del objeto de la entidad correspondiente `clMapHistoricoProvisioning` para el caso.
- **GetXCriterios:** Tiene como parámetro un número (tipo de operación) para ejecutar una operación del Store Procedure, retorna `DataSet`.
- **setData:** Escribe sobre la base de datos. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.
- **getData:** Ejecuta el procedimiento almacenado con la colección de parámetros que se ha asignado. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.

3.2.3.2.5 ctrSitiostopologia



	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

Constructor


- Asigna la propiedad `strNombreStp` el valor “`stpSitios`”.
- Asigna la propiedad `strIBSCnn` en valor recibido en el constructor.

Propiedades

- **ICCCnn:** Devuelve el valor asignado por el constructor.
- **Parametros:** Recibe Tipo de Operación, en esta propiedad se realiza el mapeo de parámetros para ejecutar el procedimiento almacenado en la base de datos “`stpSitios`”. Para realizar la creación de parámetros llama al método “CrearParametro” de la clase `clProveedorDb`.
- **LstSitiosTopologia:** Es de tipo List(Of `clSitiosTopologia`), mediante esta propiedad se retorna el resultado de ejecutar el procedimiento almacenado “`stpSitios`” con la información ordenada.
- **SitioTopologia:** Es de tipo `clSitiosTopologia`, mediante esta propiedad se realiza el mapeo a los parámetros del procedimiento almacenado “`stpSitios`”.

Métodos

- **GetSitiosTopologia:** Retorna instancia de objeto de tipo `clSitiosTopologia`, para ello:
 - Ejecuta el método `GetByCriteria` el cual retorna un `DataSet`.
 - A partir del `DataSet` se realiza el mapeo (cargue) del objeto de la entidad correspondiente `clSitiosTopologia` para el caso.
- **GetByCriteria:** Tiene como parámetro un número (tipo de operación) para ejecutar una operación del Store Procedure, retorna `DataSet`.
- **setData:** Escribe sobre la base de datos. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.
- **getData:** Ejecuta el procedimiento almacenado con la colección de parámetros que se ha asignado. Se le pasa el nombre del procedimiento almacenado y el tipo de operación.

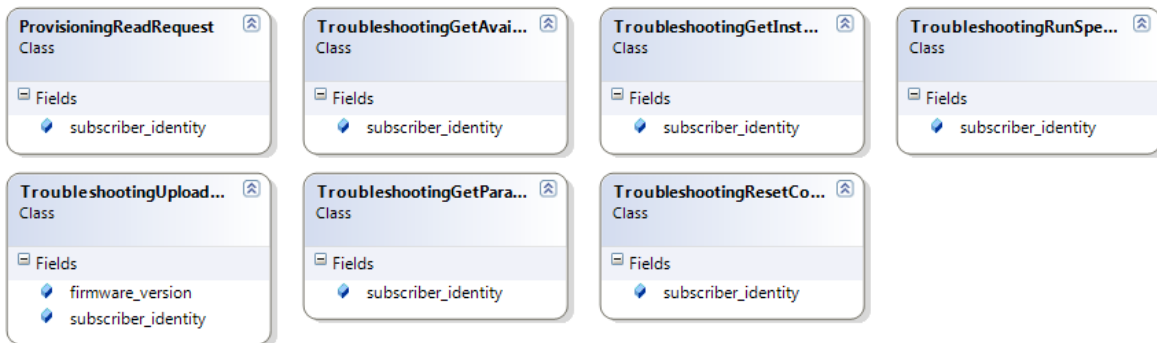
	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.2.4 Capa de negocio TROUBLESHOOTING

A continuación se definen las clases que permiten al Net Check hacer consultas hacia el sistema ProvisioningNAPI utilizando WS.


3.2.4.1 Business Objects

Estas clases contienen la información que es enviada al ProvisioningNAPI por medio de mensajes XML.

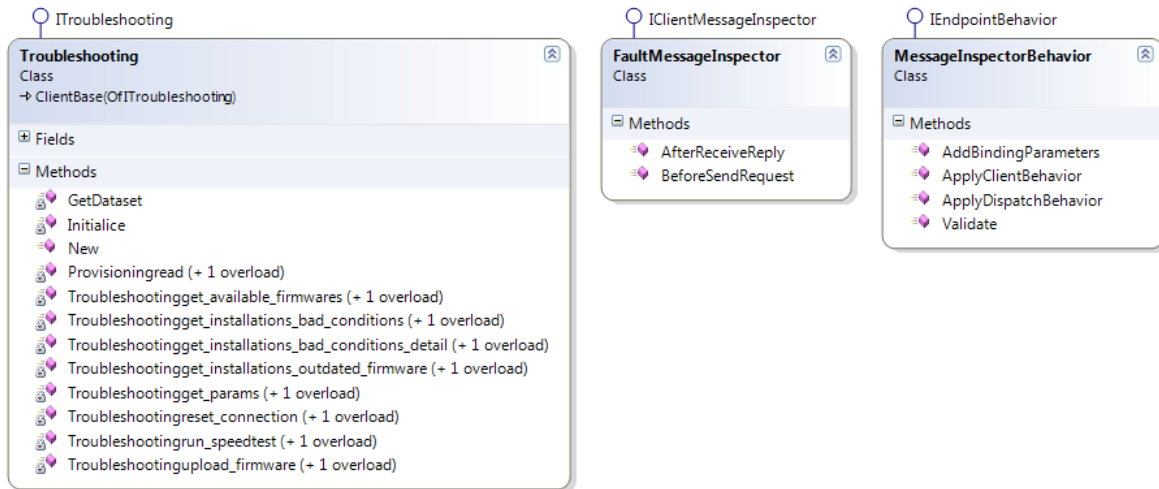


Atributos:

- **subscriber_identity:** Este campo guarda el valor de la IMSI. Este es enviado como parámetro en la consulta.
- **firmware_version:** Este campo guarda el nombre de la versión del firmware el cual se requiere actualizar. Este es enviado como parámetro en la consulta.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.2.4.2 Business Process




La clase “Troubleshooting” es la encargada de estructurar las peticiones que se envían por WS al Provisioning y las respuestas que se reciben de este mismo.

Métodos

- **GetDataset:** Recibe el mensaje en un XML, el cual lo envía al método “ConvertToDataset”, quien se encarga de retornarlo en un dataset.
- **Initialice:** Inicializa los atributos de la clase con su respectiva URL.

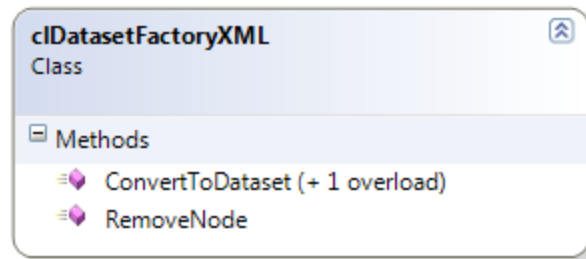
Los métodos restantes se encargan de definir un **endpoint** con la información del WS cliente. Luego se define el mensaje con la versión del Soap, el contrato y el parámetro de consulta, el cual es enviado a la clase “FaultMessageInspector”, quien se encarga de enviar la petición hacia el Provisioning.

Cuando se recibe respuesta de la petición, el método “AfterReceiveReply” de la clase “FaultMessageInspector”, se encarga de obtener el código y la descripción de errores en caso de presentarse en la respuesta del mensaje, de este modo se genera una excepción mostrando la información.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.2.5 Otras clases

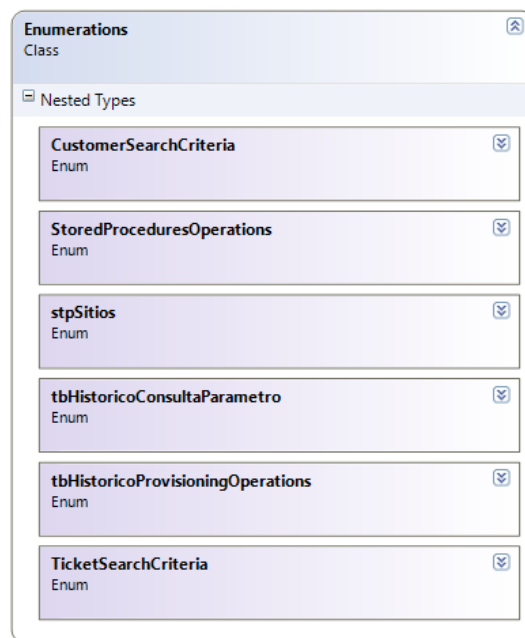
3.2.5.1 Clases *clDatasetFactory*




Esta clase se encarga de separar los parámetros contenidos en el XML recibido por el Provisioning, para retornarlos en un dataset.

3.2.5.2 Clases *Enumerations*

Esta clase contiene la configuración numérica de los parámetros que son enviados a los procedimientos almacenados como tipo de operación a realizar.



	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.3 CAPA DE PRESENTACIÓN

Esta capa del proyecto contiene definición de páginas ASP y archivos de recursos necesarios para la vista del sitio web.

3.3.1 *Páginas ASP*

Dentro de la carpeta “page” se encuentra lo siguiente:

3.3.1.1 *NetCheck.master*

Contiene el diseño base de las páginas de la aplicación.

3.3.1.2 *customer*

Contiene los controles de la página “customer”, cual se encarga de todo el proceso de consulta y verificación por parte del CSR.


3.3.1.3 *installations_firmware*

Contiene los controles de la página “installations_firmware”, cual se encarga de la consulta de firmware desactualizado.

3.3.1.4 *installations_parameters*

Contiene los controles de la página “installations_parameters”, cual se encarga de parámetros no aceptables.

Nota: Para ampliar esta información, consulte el manual de usuario [Net Check](#).

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

3.3.2 *Archivos de recursos*

Los archivos de recursos se encuentran en la carpeta Resources, la cual contiene dos subcarpetas llamadas “images” y “css”.


3.3.2.1 *Css*

Contiene los estilos para las páginas. Para la aplicación se utilizan los siguientes archivos:

- customer.css: Contiene los estilos de la página “customer”
- NETStyles.css: Contiene los estilos de [NetCheck.master](#)

3.3.2.2 *images*

Contiene todas las imágenes que utiliza la aplicación.


	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

4 CONFIGURACION DE LA APLICACIÓN


En el archivo de configuración de la aplicación web.config se realizan las siguientes configuraciones:

4.1 PARÁMETROS DE LA APLICACIÓN


NOMBRE	DESCRIPCIÓN	OBSERVACIONES
IstModelosTestLTE	Lista de modelos de Simcard con tecnología LTE.	Los valores deben ir separados por coma (,) y deben corresponder con identificadores de la tabla "model_id" de la base de datos del sistema ICC.
IstModelosTestBTB	Lista de modelos de Simcard con tecnología BTB.	Los valores deben ir separados por coma (,) y deben corresponder con identificadores de la tabla "model_id" de la base de datos del sistema ICC.
IstModelosIndoor	Lista de modelos de módem Indoor.	Los valores deben ir separados por coma (,) y deben corresponder con identificadores en el campo "model_id" de la base de datos del sistema ICC.
IstModelosOutdoor	Lista de modelos de módem Outdoor.	Los valores deben ir separados por coma (,) y deben corresponder con identificadores de la tabla "model_id" de la base de datos del sistema ICC.
IstModelosNA	No existe modelo de modem.	Cuando el sistema ICC, devuelva "0" en el campo "model_id", los campos con la información del hardware serán mostrados como "NA".
decMinRSRP	Nivel minimo aceptable de señal del módem.	Este valor se compara con el valor "rsrp" obtenido de ejecutar el método "Troubleshooting.get_params" del WS Troubleshooting para pintar un color en el respectivo campo en la pantalla de la siguiente forma: Si "rsrp" >= decRSRP: Verde Si "rsrp" < decRSRP: Rojo

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

decMinCINR	Nivel minimo aceptable de señal a ruido.	<p>Este valor se compara con el valor "cinr" obtenido de ejecutar el método "Troubleshooting.get_params" del WS Troubleshooting para pintar un color en el respectivo campo en la pantalla de la siguiente forma:</p> <p>Si "cinr" >= decCINR: Verde</p> <p>Si "cinr" < decCINR: Rojo</p>
decMinAtenuacion	Nivel minimo aceptable de atenuación.	<p>Este valor se compara con el valor "attenuation" obtenido de ejecutar el método "Troubleshooting.get_params" del WS Troubleshooting para pintar un color en el respectivo campo en la pantalla de la siguiente forma:</p> <p>Si "attenuation" >= decMinAtenuacion</p> <p>Y "attenuation" <= decMaxAtenuacion</p> <p>Entonces: "Verde"</p> <p>Si no: Rojo</p>
decMaxAtenuacion	Nivel maximo aceptable de atenuación.	<p>Este valor se compara con el valor "attenuation" obtenido de ejecutar el método "Troubleshooting.get_params" del WS Troubleshooting para pintar un color en el respectivo campo en la pantalla de la siguiente forma:</p> <p>Si "attenuation" >= decMinAtenuacion</p> <p>Y "attenuation" <= decMaxAtenuacion</p> <p>Entonces: "Verde"</p> <p>Si no: Rojo</p>
intComProductCatInternet	Id del commercial product category de Internet.	Valor numérico.
intComProductCatBasico	Id del commercial product category de Producto básico de televisión.	Valor numérico.
lstCodigosErrorExcluidos	Lista de los códigos de error devueltos por el ws	Los valores deben ir separados por coma (,) y deben corresponder con los códigos de error

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

	de Provisioning que no se mostrarán en la aplicación.	del ws Provisioning que actualmente se manejan.
intNumeroDiasHistorico	Número de días desde la fecha hacia atrás que se mostrarán eventos.	Actualmente se encuentra configurado tal cual como se muestran los eventos en IBS (7).
estadosNoValidosReprocesarEvento	Lista de estados de los eventos que no permiten activar la opción de reprocesar.	Los valores deben ir separados por coma (,) y deben corresponder con los códigos de respuesta que actualmente maneja Robot NET.
strNoAplicaIBS	Texto que indica que un valor no aplica.	
strCatg1	Definicion Categoría IM	Indica que la categoría 1 es “IM” (Instalación y mudanza)
strCatg2	Definicion Categoría TA	Indica que la categoría 1 es “TA” (Tolerancia)
timeQryHistorico	Número de días desde la fecha hacia atrás que se mostrarán las consultas de historico de parámetros.	Actualmente se encuentra configurado con 2 días.
timeQryTicket	Número de días desde la fecha hacia atrás que se mostrarán los tickets.	Actualmente se encuentra configurado con 156 días.
lbsDisable	Indica si se hace busqueda en ICC o no.	Si el valor es 0, entonces se realiza la busqueda de la información en el ICC, de lo contrario se omitirá.
strMensajeActualizacionFirmware	Cuando se esta actualizando el firmware del modem se muestra una cadena de texto que dice “Actualización de firmware en proceso...”.	
OcupacionAlta	Nivel de ocupación Alta.	Se toma el campo “high_occupancy” y si el valor de esta es igual 3, entonces aparecera una imagen circular de color rojo al lado del campo “Ocupación”.
OcupacionMedia	Nivel de ocupación Media.	Se toma el campo “high_occupancy” y si el valor de esta es igual 2, entonces aparecera una imagen circular de color amarillo al lado

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

		del campo "Ocupación".
OcupacionMedia2	Nivel de ocupación Media 2.	Se toma el campo "high_occupancy" y si el valor de esta es igual 1, entonces aparecera una imagen circular de color naranja al lado del campo "Ocupación".
OcupacionBaja	Nivel de ocupación baja.	Se toma el campo "high_occupancy" y si el valor de esta es igual 0, entonces no aparecera nada al lado del campo "Ocupación".

4.2 CONEXIONES A BASES DE DATOS TRANSACCIONALES

Información de configuración de las fuentes de datos que se utilizan en el proyecto así:

- cnStrDBIBSCnn: String de conexión a la base de datos del sistema IBS (Oracle).
- cnStrbdIBSBatchLog: String de conexión a la base de datos del sistema Robot Internet (SQL Server).
- cnStrbdOTRSCnn: String de conexión a la base de datos del sistema OTRS (MySQL Server).


Ejemplo

```
<connectionStrings>
  <add name="cnStrDBIBSCnn" connectionString="Server=****;uid=****;pwd=****;"
    providerName="System.Data.OracleClient"/>
  <add name="cnStrbdIBSBatchLog" connectionString="Data Source=****;Initial
    Catalog=Test;User ID=****; Password=****;Connect Timeout=240"
    providerName="System.Data.SqlClient"/>
  <add name="cnStrbdOTRSCnn" connectionString="server=****;User
    Id=****;password=****;database=****"
    providerName="MySql.Data.MySqlClient" />
</connectionStrings>
```

4.3 WEB SERVICE PROVISIONING

Se requiere la siguiente configuración para el correcto funcionamiento de consulta de información hacia el web service de Provisioning en la sección

</system.serviceModel>:


	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

```

<system.serviceModel>
  <diagnostics>
    <messageLogging logMalformedMessages="true" logMessagesAtServiceLevel="true"
logMessagesAtTransportLevel="true"/>
  </diagnostics>
  <extensions>
    <bindingElementExtensions>
      <add name="customTextMessageEncoding"
type="Microsoft.Samples.CustomTextMessageEncoder.CustomTextMessageEncodingElement,
CustomTextMessageEncoder, Version=4.0.0.0, Culture=neutral, PublicKeyToken=null"/>
    </bindingElementExtensions>
  </extensions>
  <bindings>
    <customBinding>
      <binding name="ISO8859Binding">
        <customTextMessageEncoding encoding="UTF-8" messageVersion="Soap11"/>
        <httpTransport maxReceivedMessageSize="402653184"
maxBufferSize="402653184"
maxBufferPoolSize="402653184"/>
      </binding>
    </customBinding>
  </bindings>
  <client>
    <endpoint name="ITroubleshootingLTE" binding="customBinding"
bindingConfiguration="ISO8859Binding"
contract="libDTVCOLNETTTroubleshooting.Troubleshooting.BusinessProcess.ITroubleshooting"
address="http://*****/*****/*****/BOG/LTE?wsdl"/>
    <endpoint name="ITroubleshootingBTB" binding="customBinding"
bindingConfiguration="ISO8859Binding"
contract="libDTVCOLNETTTroubleshooting.Troubleshooting.BusinessProcess.ITroubleshooting" address="http://
*****/*****/*****/BOG/BTB?wsdl"/>
  </client>
</system.serviceModel>

```

La URL del servicio no se especifica aquí porque es consultada en la base de datos de Robot NET y su funcionamiento es de forma dinámica.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0


5 EJECUCIÓN

Net Check es una aplicación Web, por lo tanto es requisito para su ejecución contar con alguno de los siguientes navegadores en el equipo cliente:

- Internet Explorer 8 o superior.
- Mozilla Firefox 27.0.1 o superior.

El equipo servidor debe contar con lo siguiente:

- Acceso a los web services de Troubleshooting, para probar esto consultar la tabla “tbMapURLProvisioning” de la base de datos de Robot NET y probar el acceso en el navegador algunas de las URL’s que se encuentran en el campo “URLProvisioning”.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

6 GLOSARIO

6.1 ENDPOINT

Información del Web Service Provisioning.

6.2 ENTIDAD CONTROLADORA

La arquitectura definida para la aplicación Net Check hace uso de este concepto, mediante el cual la entidad controladora realiza los llamados al procedimiento almacenado y retorna la información a la entidad que realice el llamado.

6.3 FIRMWARE

Firmware se refiere a las aplicaciones y el sistema operativo que controla el funcionamiento de un dispositivo. Se llama firmware en lugar de software para destacar que está muy estrechamente ligada a los componentes de hardware específicos de un dispositivo.

6.4 ICC

CRM que utiliza DIRECTV, el cual contiene la base de clientes activa e inactiva de DirecTV.

6.5 IMSI

Serial de la simcard. Identifica al equipo en la red, de forma que sea posible realizar diagnóstico y solución de problemas de forma inmediata (Troubleshooting).


6.6 MÓDEM

Es el dispositivo que convierte las señales digitales en analógicas y viceversa, permitiendo la comunicación entre computadoras a través del cablemódem

Sinónimos: CPE

6.7 OTRS

Sistema de tickets.

	Nombre Documento	Versión
	Net Check Manual Técnico	1.0

6.8 PROVISIONING

Sistema de aprovisionamiento de Internet.

6.9 ROBOT NET

Aplicación que se encarga de procesar todas las transacciones (eventos) relacionadas al aprovisionamiento de Internet (Activaciones, Swop's, Suspensiones, Cancelaciones, Reconexiones, Desconexiones, Upgrades/Downgrades) enviando la respectiva solicitud al Core de Internet.

6.10 SIMCARD

Tarjeta inteligente desmontable usada para los módems LTE.

6.11 XML

Siglas en inglés de eXtensible Markup Language ("lenguaje de marcas Extensible"), es un meta-lenguaje que permite definir lenguajes de marcas, utilizado para almacenar datos en forma legible.

6.12 WS

Web Service.