**Manual de Instalación**

# La presente guía Indica los procedimientos para instalar la plataforma “Urbetrack framework 2.0”, en adelante Urbetrack.

**Se requiere que el lector posea conocimiento de:**

administración del sistema operativo Windows

instalación de programas

instalación de servicios

instalación de componentes de Windows

manejo de permisos

.net conectionStrings

formato de datos XML (para la configuración de los componentes de Urbetrack)

La instalación consta de los siguientes pasos:

1. **Verificar los Pre-Requisitos:**

* .NET Framework 3.5
* Windows Installer 3.1
* MSMQ
* Servidor Web IIS
* SQL Server
* Estructura y contenido de base de datos actualizada (ver Apartado)
* MemCache (ver Apartado)

1. **Instalar los componentes de Urbetrack:**

* Addresser
* Aplicación Web
* Scheduler
* Servicios de Gateway

***Pre-Requisitos***

Levantar Bases de Datos

Crear las bases (Urbetrack, Urbetrack\_history, Adresser) y cargar sus respectivos archivos de Back-up. Al levantar el back-up verificar que la opción “Overwrite the existing database” este tildada, creación de Logins:

* Security -> Logins -> New
* Tildar la opción “SQL Authentication”
* Des tildar la opción “Enforce password policy”
* Definir como “Default database” la base del login que se esté creando.
* Poner como Server Roles “sysadmin” y “public”
* En User Mapping tildar la base correspondiente

MemCache

Instalar el manager de memcache (el archivo msi). Una vez instalado abrir el manager y darle click al botón que dice Add Server. Como Server Name poner 127.0.0.1 y dejen el puerto default que es 11211, click en Apply

Ahora hay que genera una instancia de memcache para ese servidor, para ello click derecho en el server de la lista -> Add New Instance. Poner cualquier Instance Name y cualquier valor de configuración que desee (los pueden dejar en los valores por defecto)

Luego de eso descompriman el archivo rar y copien el contenido en las rutas de memcache y del manager (C:\Program Files\MemCacheD -> Memcache y C:\Program Files\MemCacheD Manager\BinaryFiles -> Manager, si se dejan las rutas por defecto).

Volver al manager y hacer click derecho sobre el servidor en la lista -> Redeploy / Update MemCacheD, para actualizar memcache a la última versión disponible. Si todo salió bien ya debería tener su servidor e instancia de memcache configurada correctamente

***Componentes de la plataforma:***

Aclaración: para cualquier error que ocurra o después de cada instalación, primero verificar la seguridad en todas las carpetas y archivos que se estén siendo utilizados. Para el grupo Everyone, tildar todos los permisos de seguridad (otorgar Full Control).

Addresser

El paquete de instalación del Addresser contiene los archivos ejecutables del instalador y la base de datos cartográfica que se haya adquirido con el producto. Los archivos de instalación están compuestos por un archivo Microsoft Installer (.msi) y un ejecutable (setup.exe), estos archivos se consiguen compilando el proyecto del Addresser y su instalador.

Para comenzar la instalación, ejecutar el archivo setup.exe y seguir los pasos del asistente.

Para configurar el acceso a la base de datos debemos localizar la carpeta de instalación. En nuestro caso utilizaremos como ejemplo la ruta:

C:\Program Files\Mapas y Sistemas\Addresser Service

El Addresser utiliza la herramienta *NHibernate* para resolver la conexión a diferentes motores de base de datos, por lo que deberemos modificar su configuración para poder acceder a nuestros datos. Para hacer esto debemos abrir con un editor de texto el archivo Addresser.NHibernate.config.

Dentro de este archivo en formato XML, encontraremos una serie de elementos con los valores de configuración para acceder a nuestra base de datos. A continuación se muestran la configuración correspondiente:

## SQL Server

<?xml version="1.0" encoding="utf-8" ?>

<hibernate-configuration xmlns="urn:nhibernate-configuration-2.2">

<session-factory name="NHibernate.Addresser">

<property name="**connection.driver\_class**"> *NHibernate.Driver.SqlClientDriver* </property>

<property name="**dialect**"> *NHibernate.Dialect.MsSql2000Dialect* </property>

<property name="**connection.connection\_string**"> *Data Source=(local);Initial Catalog=nomenclador;Persist Security Info=True;User ID=user;pwd=password*  
</property>

...

</session-factory>

</hibernate-configuration>

En todos los casos se deberá modificar el elemento de nombre connection.connection\_string para que tenga como valor el string de conexión a la base de datos. Esto se consigue cambiando el server, el nombre de la base de datos, el usuario y la contraseña.

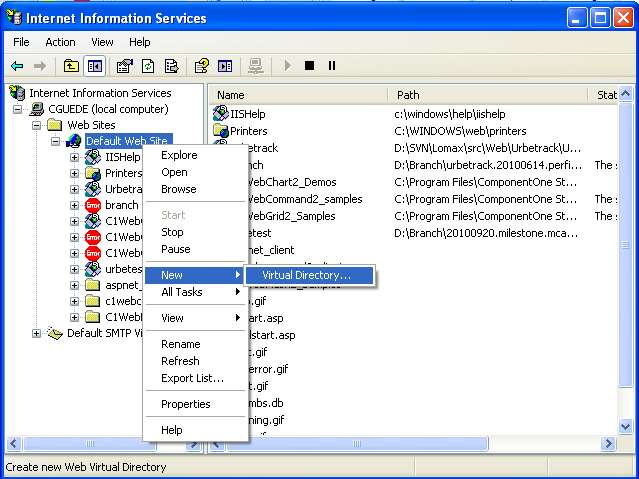
Para realizar la activación del producto hay que correr el keygen en la maquina donde está instalado y generar el archivo licence.spk (Generar, Guardar Archivo) y copiarlo a la carpeta del producto.

Una vez realizados estos pasos, el Addresser estará funcionando correctamente. Para utilizarlo configurar el acceso Remoting en el archivo .config de la aplicación cliente.

Hay dos interfaces: INomenclador, ICleaning.

Aplicación Web

Crear un nuevo directorio virtual en ISS: panel de control, herramientas administrativas, Internet Information Services y seguir la captura siguiente:



Luego seguir los pasos del asistente.

Pre compilar la aplicación Web y buscar dentro de la solución la carpeta “Precompiled Web”. Luego hay que copiar estos archivos en el servidor correspondiente. No se debe pisar el archivo app\_licencies, en caso de no tenerlo por ser una instalación nueva, o haberlo borrado por error, se debe copiar este archivo desde otro servidor que lo tenga. Tampoco modificar el archivo web.config de no ser necesario.

Verificar los logins de la base de datos en el web.config, si no están creados, se deben crear como se explicó anteriormente.

Se deben copiar crear las key de google maps y de google earth, paro eso entrar a la página de google y buscar las keys correspondientes. Copiar estas keys en el web config, deben estar 2 veces cada una, una para acceder por localhost y otra para acceder por la ip del servidor, de la siguiente manera:

<add key="GoogleMaps.localhost" value="http://maps.google.com/maps?file=api&amp;v=2&amp;key=ABQIAAAA-ZFQIp6XzCGyF\_TNWwe0HxR3WEPGHRgqNz51JZNWVhRh6GsvQxQVjFpZVcPytX-LgUzSw5pld2F-tg"/>

<add key="GoogleMaps.200.26.26.66" value="http://maps.google.com/maps?file=api&amp;v=2&amp;key=ABQIAAAA-ZFQIp6XzCGyF\_TNWwe0HxR3WEPGHRgqNz51JZNWVhRh6GsvQxQVjFpZVcPytX-LgUzSw5pld2F-tg"/>

<add key="GoogleEarth.localhost" value="http://www.google.com/jsapi?key=ABQIAAAA-ZFQIp6XzCGyF\_TNWwe0HxT2yXp\_ZAY8\_ufC3CFXhHIE1NvwkxQ381eqP7m5GB2RE14yqpj9uuCzeQ"/>

<add key="GoogleEarth.200.26.26.66" value="http://www.google.com/jsapi?key=ABQIAAAA-ZFQIp6XzCGyF\_TNWwe0HxT2yXp\_ZAY8\_ufC3CFXhHIE1NvwkxQ381eqP7m5GB2RE14yqpj9uuCzeQ"/>

Scheduler

Compilar el proyecto del Scheduler y el instalador. Seguir los pasos del instalador.

Para su configuración utilizar el archivo de configuración en la carpeta del Scheduler luego de instalarlo. Ahí se muestran ejemplos de cómo configurar los timers para las distintas tareas, a continuación se muestra un ejemplo de cómo configurar un timer para realizar la tarea de datamart a las 14:00hs, todos los dias:

<?xml version="1.0" encoding="utf-8" ?>

<Configuration xmlns="urn:Scheduler.Configuration">

<Timer Enabled="true" Name="Datamart" StartTime="14:00:00" Periodicity="Day">

<Task Class="Urbetrack.Scheduler.Tasks.Datamart.Task, Urbetrack.Scheduler.Tasks.Datamart"/>

</Timer>

</Configuration>

Servicios de Gateway

Instalación:

* Copiar la salida de la compilación del proyecto run.exe a la carpeta de instalación elegida.
* Darle al usuario “Network Service” permisos “full control” sobre esa carpeta y subcarpetas.
* De ser necesario editar los XML de los servicios que se quieran correr. (ver apartado Configuración)
* Ejecutar el programa “Run.Exe” y elegir la opción “Instalar y Correr Servicios” del menú.

Configuración:

Los archivos XML en la carpeta “Applications” describen los servicios, configuran parámetros y también sirven para la instalación y ejecución de los mismos.

La instalación crea ejemplos de:

* 1 XML de servicio de Gateway para cada tipo de dispositivo/sensor y para el servicio de Dispatcher.
* 1 XML para registrar todos los servicios en el sistema operativo (necesario durante la instalación y creación de nuevos servicios).
* 1 XML para cómodamente levantar/bajar todos los servicios.

Edición del XML para servicio de Gateway y dispatcher:

Verificar que los siguientes elementos sean validos:

* ConectionString (conexión a la base de datos)
* Nombres de las Colas MSMQ: la cola commander se debe corresponder con el valor que este en la base de datos para ese tipo de dispositivo; el resto de las colas se deben corresponder en los gateways con las colas que lee el servicio de dispatcher; por defecto todo es correcto.
* El puerto udp donde llegan los datos de los sensores, tag UDPNetworkServer propiedad Uri.

Edición del XML para el instalador de los servicios:

Verificar que exista un tag WinService para cada servicio que se desee correr con la propiedad HostApplication referenciando al XML correcto para el servicio ("Path\Nombre.xml").

Edición del XML para el manejador de los servicios:

El XML para levantar todos los servicios cómodamente contiene tres líneas, donde la propiedad ServiceNamePrefix del tag WinServiceManager tiene que contener el mismo texto que la propiedad “ServiceNamePrefix” del tag “WinServiceGroup” del XML instalador.