Démarrer avec le Kit de démarrage ODAF (Open Data Application Framework) Openturf

Solution Open Data Clé en main

Publication : Juin 2013

Version: 2.1

Auteurs : Philippe Beraud (Microsoft France), Sébastien Brasseur (Microsoft France), Rémi Olivier (Microsoft France), Frédéric Romanczuk (Microsoft France), Alexandre Fournier (Microsoft France)

Pour les dernières informations, consultez

aka.ms/OpenDataFrance

Sommaire

[Licence du Kit de démarrage ODAF Openturf 4](#_Toc358205301)

[Commentaires 5](#_Toc358205302)

[Résumé 5](#_Toc358205303)

[Prérequis 5](#_Toc358205304)

[Connaissances et attendu 6](#_Toc358205305)

[Présentation rapide du Kit de démarrage ODAF Openturf 8](#_Toc358205306)

[Présentation rapide de Windows Azure 12](#_Toc358205307)

[Configuration requise pour le Kit de démarrage ODAF Openturf 15](#_Toc358205308)

[Configuration système 15](#_Toc358205309)

[Comment installer les composants système nécessaires 17](#_Toc358205310)

[Obtention des sources du kit de démarrage ODAF Openturf 18](#_Toc358205311)

[Contenu du Kit de démarrage ODAF Openturf 19](#_Toc358205312)

[Organisation du code source du Kit de démarrage 21](#_Toc358205313)

[Dossier Database 23](#_Toc358205314)

[Scripts SQL 23](#_Toc358205315)

[Organisation projet ODAF.Data 24](#_Toc358205316)

[Eléments de configuration 24](#_Toc358205317)

[Dossier *Libraries* 24](#_Toc358205318)

[Dossier *Dependencies* 25](#_Toc358205319)

[Dossier *ODAF.Azure* 25](#_Toc358205320)

[Dossier *ODAF.Android* 26](#_Toc358205321)

[Eléments de configuration 26](#_Toc358205322)

[Dossier *ODAF.iPhoneApp* 27](#_Toc358205323)

[Eléments de configuration 27](#_Toc358205324)

[Dossier *ODAF.SilverlightApp* 27](#_Toc358205325)

[Organisation projet 28](#_Toc358205326)

[Eléments de configuration 29](#_Toc358205327)

[Dossier *ODAF.Website* 29](#_Toc358205328)

[Organisation projet 30](#_Toc358205329)

[Eléments de configuration 31](#_Toc358205330)

[Dossier *ODAF.WindowsPhone* 31](#_Toc358205331)

[Organisation projet 31](#_Toc358205332)

[Eléments de configuration 32](#_Toc358205333)

[Dépendances 33](#_Toc358205334)

[Compilation des éléments du Kit de démarrage ODAF Openturf 35](#_Toc358205335)

[Compilation des éléments en ligne de commande 35](#_Toc358205336)

[Compilation des éléments en utilisant Visual Studio 2010 35](#_Toc358205337)

[Mise en œuvre des éléments du Kit de démarrage ODAF Openturf 36](#_Toc358205338)

[Pour aller plus loin 38](#_Toc358205339)

# Licence du Kit de démarrage ODAF Openturf



Le Kit de démarrage **ODAF** (*Open Data Application Framework*) **Openturf** est publié sous le contrat de licence de logiciel libre Microsoft Public License (Ms-PL) tel que décrit à l’adresse Internet <http://opensource.org/licenses/ms-pl.html>.

# Commentaires

Votre opinion nous intéresse. N’hésitez pas à nous transmettre vos commentaires via la boîte aux lettres Open Data France (<mailto:ogdifrance@live.fr>) afin que nous puissions améliorer la [solution Open Data Clé en main](https://mspartner.microsoft.com/fr/fr/Pages/Solutions/Open-Data-Cle-en-main.aspx)[[1]](#footnote-1) disponible sur le site Microsoft Partner Network et ses différents accélérateurs.

Le [blog MSDN Open Data France](http://blogs.msdn.com/b/ogdifrance/)[[2]](#footnote-2) vous tient informé dans le même temps des dernières évolutions de la solution Open Data Clé en main, des différents accélérateurs proposés dans ce contexte, de l’actualité et des nouveautés relatives à ceux-ci.

# Résumé

Le kit de démarrage **ODAF** (*Open Data Application Framework*) **Openturf** sous license libre est une initiative menée par les équipes Microsoft en charge du [Secteur Public](http://www.microsoft.com/france/entreprises/secteur-public/solutions-pour-secteur-public.aspx)[[3]](#footnote-3). ODAF Openturf utilise les services de la plateforme de publication de données ouvertes **OGDI** (*Open Government Data Initiative*) **DataLab** hébergée sur [Windows Azure](http://www.microsoft.com/france/windows-azure/)[[4]](#footnote-4) dans le Cloud pour faciliter l’affichage sur carte et l'utilisation d’informations publiques, ou données ouvertes (*open data*) géo-localisées variées provenant d'entités publiques (gouvernement, ministères, collectivités locales et territoriales, mairies, services déconcentrés de l'Etat, etc.).

Après une courte présentation du kit de démarrage, de ses fondations et architecture, ce document décrit le contenu à proprement parler du kit, l’organisation du code source ainsi mis à disposition et ses modalités de compilation. D’autres documents accompagnant le kit décrivent les modalités de mise en œuvre du kit et de mise à disposition de données notamment dans Windows Azure.

Ce document s’adresse aux architectes, aux développeurs et à tous ceux qui souhaitent consommer des informations publiques géolocalisées via le Framework ODAF Openturf et la plateforme de publication OGDI DataLab.

# Prérequis

Ce Kit de démarrage ODAF Openturf (*Open Data Application Framework*) nécessite à minima un ordinateur sous Windows Server 2003 (R2) SP1, Windows 7, Windows Server (R2) 2008 ou ultérieur.

Le lecteur doit connaître notamment le langage C# et le Framework Silverlight afin de pouvoir facilement comprendre les éléments mis ainsi à disposition.

# Connaissances et attendu

La mise en œuvre du Kit de démarrage ODAF Openturf (*Open Data Application Framework*) requiert une expérience préalable :

* De l’environnement de développement Microsoft Visual Studio 2010,
* De la programmation avec le Microsoft Framework .NET,
* De la technologie [Silverlight](http://www.silverlight.net/)[[5]](#footnote-5),
* De la technologie Web [ASP.NET MVC](http://www.asp.net/mvc)[[6]](#footnote-6),
* De l’utilisation d’interfaces Web de style REST (*Representational State Transfer*),
* De la plateforme Windows Azure,
* Du développement avec les interfaces programmatiques (API) Twitter et Facebook pour l’intégration avec les réseaux sociaux,
* Du développement sur Android pour la partie Android,
* Du développement sur iOS pour la partie iPhone,
* De Windows Phone [Windows Phone 7.1 (nom de code Mango)](http://www.microsoft.com/windowsphone/en-us/cmpn/mango-overview.aspx)[[7]](#footnote-7).

Remarque : Microsoft met à disposition des formations gratuites téléchargeables sur ces environnements et technologies. Nous vous invitons à considérer le cas échéant les kits de formation suivants qui comprennent des présentations, des vidéos de formation, des démonstrations accompagnées de leur script ainsi que des ateliers didactiques complets :

* [Microsoft Windows Azure Training Kit](http://www.microsoft.com/downloads/en/details.aspx?FamilyID=413E88F8-5966-4A83-B309-53B7B77EDF78&displaylang=en)[[8]](#footnote-8) couvrant la plateforme Windows Azure et l’environnement de développement Microsoft Visual Studio 2010.
* [Microsoft Silverlight 4 Training Course](http://msdn.microsoft.com/en-us/silverlight4trainingcourse.aspx)[[9]](#footnote-9) couvrant les nouvelles fonctionnalités de Silverlight 4, des exercices pratiques, ainsi que 8 modules pour construire des applications métier avec Silverlight 4.
* [Windows Phone 7.1 Training Kit](http://msdn.microsoft.com/en-us/wp7trainingcourse.aspx)[[10]](#footnote-10) couvrant les Framework Silverlight et XNA sous forme d’exercices pratiques et de vidéos. Vous y apprendrez les bases de la navigation dans Windows Phone, les différents contrôles, le lancement de tâches, la gestion du cycle de vie de votre application, etc.

Remarque : Le [blog du groupe Produit de la plateforme Windows Azure](http://blogs.msdn.com/b/windowsazure/)[[11]](#footnote-11) propose également de nombreuses ressources en termes de formation sur cet environnement d’exécution et d’hébergement.

Celui-ci constitue également un excellent moyen de se tenir informé des évolutions proposées par la plateforme.

Remarque :

L’intégration programmatique avec les principaux réseaux sociaux est notamment couverte dans l’ouvrage [Developer's Guide to Social Programming: Building Social Context Using Facebook, Google Friend Connect, and the Twitter API](http://www.amazon.fr/Developers-Guide-Social-Programming-Building/dp/0321680774/ref=sr_1_1?ie=UTF8&qid=1313055760&sr=8-1)[[12]](#footnote-12) aux éditions Addison-Wesley Professional.

[](http://www.wrox.com/WileyCDA/WroxTitle/Professional-Twitter-Development-With-Examples-in-NET-3-5.productCd-0470531320.html)

Vous pouvez également consulter l’ouvrage l’ouvrage [Professional Twitter Development: With Examples in .NET 3.5](http://www.wrox.com/WileyCDA/WroxTitle/Professional-Twitter-Development-With-Examples-in-NET-3-5.productCd-0470531320.html)[[13]](#footnote-13) aux éditions Wrox dont certains exemples de code sont repris dans le Framework ODAF.

# Présentation rapide du Kit de démarrage ODAF Openturf

Les informations publiques, ou données ouvertes (*open data*), sont aujourd’hui en termes de potentiel, d’opportunités et d’enjeu au cœur du développement de multiples innovations sociales et économiques grâce à leur mise en ligne par l’administration et les collectivités territoriales.

Portées à la connaissance des citoyens, des chercheurs, des associations, des entreprises, ces informations statistiques, géographiques, urbanistiques, etc. peuvent devenir des mines d’or dès lors qu’elles sont retravaillées, croisées entre elles (*mash-up*) et mises en scène et ainsi contribuer à la création de nouveaux services tout en favorisant la transparence de l’action publique.

 L’[édition du premier trimestre 2011](http://www.rslnmag.fr/Content/Files/ff/UERGL1YyUkN6ZHdFaTZNSFJOQS5wZGY=)[[14]](#footnote-14) du magazine RSLN - Regards sur le numérique propose une enquête complète sur les informations publiques :

*« L’information détenue par le gouvernement fédéral est un bien national ! » Dès le lendemain de son investiture, le 20 janvier 2009, Barack Obama hissait l’Open data au premier rang de son mandat. Quatre mois plus tard, le site data.gov était lancé, qui rendait accessibles des informations publiques de tout ordre : consommation d’énergie dans les foyers, rapports d’inspection des centrales nucléaires, dépenses de l’Etat, etc. Une petite révolution, montrant qu’un état peut, sans perdre son pouvoir, jouer la carte de la transparence avec ses concitoyens.*

*Aujourd’hui personne ne mesure ni l’étendue ni les conséquences de cette philosophie, qui parie sur l’intelligence collective, la créativité fertilisée par le croisement d’autant de données et d’usages à inventer.*

*Nul ne sait, mais le rêve est permis. Car des possibilités immenses se dessinent pour créer les services et les applications qui vont faciliter la vie quotidienne des citoyens, injecter une forte dose de transparence dans la vie démocratique, entraîner de spectaculaires avancées de la connaissance…*

*À bien y regarder, l’Open data n’est ni plus ni moins que l’opportunité, pour nous tous, de produire du sens, du bien-être et de l’intelligence à partir des tonnes d’informations amassées, classées, archivées et produites par nous-mêmes. Ne manquent plus que les idées pour leur donner vie… une seconde fois !*

Editorial par Eric Boustouller, Président de Microsoft France

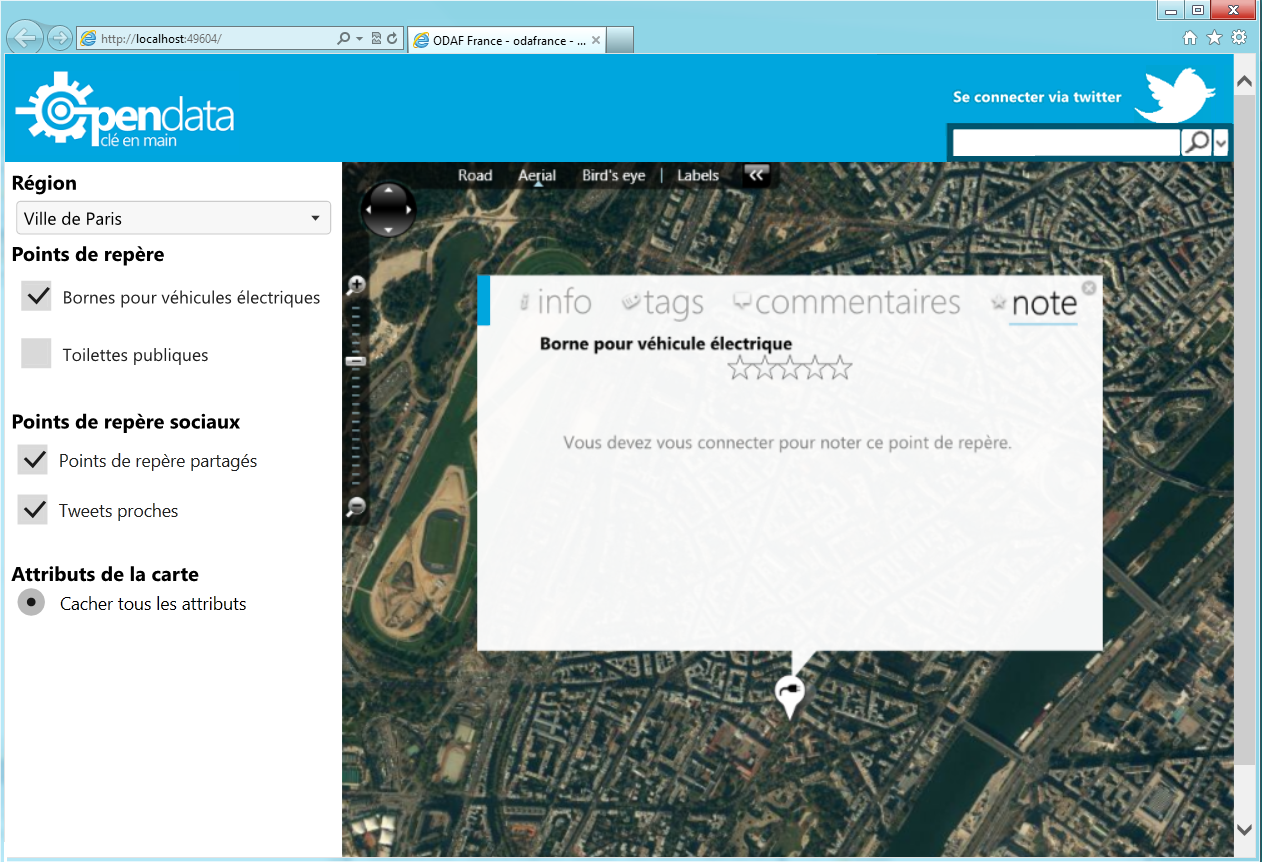
Remarque : Le site Web du magazine RSLN propose un [dossier complet](http://www.rslnmag.fr/blog/dossier/l-open-data_et-nous_et-nous_et-nous_/2011/4/19/quel-sens-donner-au-monde-des-donnees-ouvertes_/)[[15]](#footnote-15) sur le sens à donner au monde des données ouvertes avec des éclairages complémentaires comme notamment au travers de la publication des vidéos tirées de la conférence [L’Open Data, et nous, et nous et nous ?](http://www.rslnmag.fr/blog/theme/les-rencontres-rsln/2011/2/10/l-open-data_et-nous_et-nous_et-nous_venez-imaginer-le-monde-de-l-abondance-des-donnees_/)[[16]](#footnote-16) organisée le 17 mars dernier avec le [World e.gov Forum](http://wegf.org/fr/)[[17]](#footnote-17) au campus Microsoft.

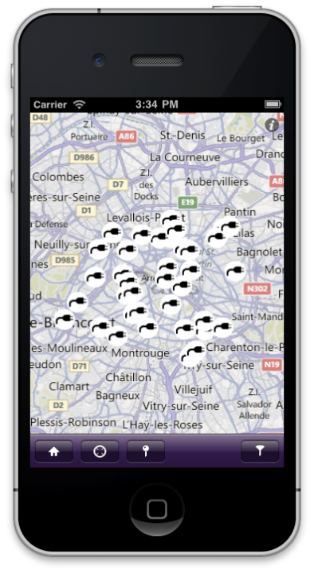
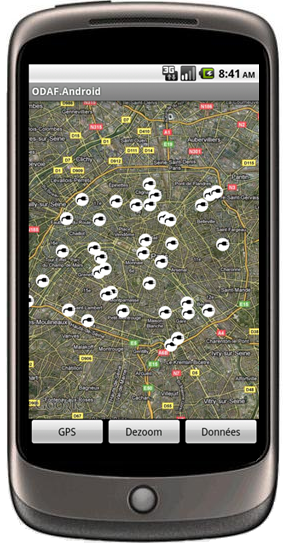
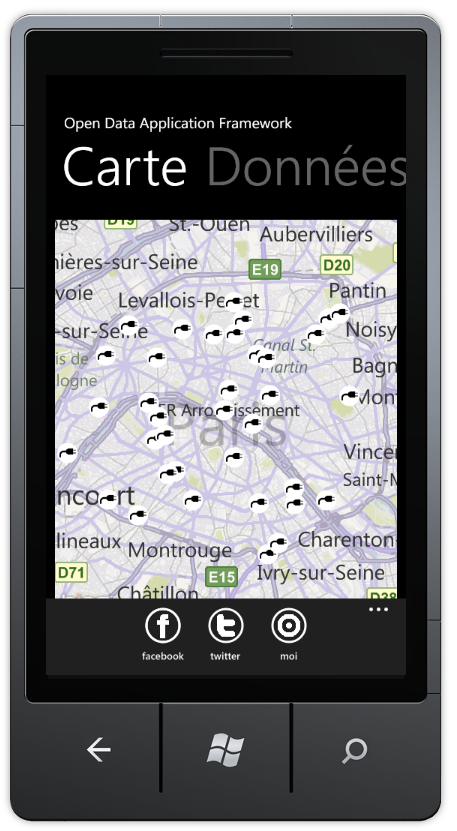
Dans un tel contexte, la publication et la consommation d’informations publiques sur le Web sous une forme facilement accessible par les personnes et les applications constituent une dimension essentielle. Afin justement de faciliter la mise à disposition et l’utilisation d’informations publiques, Microsoft France souhaite contribuer à cette dynamique en marche au travers du kit de démarrage ODAF Openturf (*Open Data Application Framework*), un Framework applicatif sous licence libre qui permet aux développeurs d’applications, à l’administration et aux collectivités territoriales d’étendre leur écosystème applicatif en servant de base pour exposer des données publiques et civiques et aux citoyens d’accéder à ces données, d’apporter leurs suggestions, de les noter et de les partager via leur réseau social comme Twitter ou Facebook. Les fonctionnalités ainsi proposées au travers de ce Framework permettent à la fois aux citoyens de bénéficier d’un accès plus facile aux données publiques et ainsi de bénéficier d’une meilleure visibilité sur les services mis à disposition par l’administration et les collectivités territoriales ainsi que de la plus grande transparence de ces dernières.

Le Kit de démarrage ODAF Openturf correspond à un ensemble d’évolutions du kit de développement logiciel éponyme disponible en téléchargement sous la même licence libre sur la [forge CodePlex](http://ogdi.codeplex.com/)[[18]](#footnote-18). Il s’inspire également librement du projet [VanGuide for Windows Phone 7](http://vanguidewp7.codeplex.com/)[[19]](#footnote-19) également disponible sur la forge CodePlex.

Il se présente sous la forme d’un modèle d’application de cartographie qui affiche sur une carte [Bing Cartes](http://www.bing.com/maps/?cc=fr) les données publiques géo-localisées. L’application client est disponible sous 4 formes différentes :

1. Une application Web riche s’exécutant dans un navigateur via le plugin Silverlight de Microsoft ;
2. Une application client mobile pour Android ;
3. Une application client mobile pour iPhone ;
4. Une application client mobile pour Windows Phone.





Pour afficher ces données géo-localisées, la solution consomme des flux de données au format KML (*Keyhole Markup Language*). [Wikipédia](http://en.wikipedia.org/wiki/Keyhole_Markup_Language) définit le format KML de la façon suivante :

*« Keyhole Markup Language (KML) is an XML notation for expressing geographic annotation and visualization within Internet-based, two-dimensional maps and three-dimensional Earth browsers. […]*

*The KML file specifies a set of features (place marks, images, polygons, 3D models, textual descriptions, etc.) for display in Google Earth, Maps and Mobile, or any other 3D Earth browser (geobrowser) implementing the KML encoding. Each place always has a longitude and latitude. Other data can make the view more specific, such as tilt, heading, altitude, which together define a "camera view". KML shares some of the same structural grammar as GML. Some KML information cannot be viewed in Google Maps or Mobile.* »

Pour alimenter une instance client ODAF Openturf en données au format KML, le Framework utilise les catalogues d’informations publiques hébergés par une instance dans le Cloud Windows Azure du kit de démarrage OGDI (*Open Government Data initiative*) DataLab sous licence libre, comme, en l’occurrence, l’instance de test OGDI DataLab France déjà disponible à l’adresse <http://ogdifrance.cloudapp.net/>.

Le Framework ODAF Openturf se présente comme un client naturel du kit de démarrage OGDI DataLab qui permet d’exposer des flux de données en KML. Le kit de démarrage OGDI DataLab rend possible la mise à disposition facilitée par l’Administration et les Collectivités Territoriales de services de données performants qu’elles exploitent, les informations publiques étant hébergées dans le Cloud Windows Azure. Les citoyens et les développeurs d’application peuvent accéder aux données ainsi publiées depuis toute plateforme ou navigateur.

Au-delà des modèles d’application évoqués ci-dessus, le Framework ODAF Openturf repose au niveau « back-end » sur une base de données Microsoft SQL Azure dans le cloud Windows Azure ou bien Microsoft SQL Server 2008 R2 en entreprise, avec possibilité de portage sur une base de données MySQL ou SQLite.

Un site Web en ASP.NET MVC communique avec la solution OGDI DataLab pour récupérer les flux de données KML. Les modèles d’application qui utilisent Bing Cartes s’interfacent avec le site Web pour récupérer et consommer les données géo-localisées au format [JSON (*JavaScript Object Notation*)](http://tools.ietf.org/html/rfc4627)[[20]](#footnote-20).

Le site Web joue alors le rôle de service RESTful qui convertir les flux KML en données au format JSON. [Wikipedia](http://en.wikipedia.org/wiki/Representational_State_Transfer)[[21]](#footnote-21) définit REST (*Representational State Transfer*) et les services Web RESTful comme suit:

« *Representational State Transfer (REST) is a style of software architecture for distributed hypermedia systems such as the World Wide Web. The term Representational State Transfer was introduced and defined in 2000 by Roy Fielding in his doctoral dissertation. Fielding is one of the principal authors of the Hypertext Transfer Protocol (HTTP) specification versions 1.0 and 1.1.*

*Conforming to the REST constraints is referred to as being ‘RESTful’. …*

*A RESTful web service (also called a RESTful web API) is a simple web service implemented using HTTP and the principles of REST. It is a collection of resources, with three defined aspects:*

* *The base URI for the web service, such as http://example.com/resources/*
* *The MIME type of the data supported by the web service. This is often JSON, XML or YAML but can be any other valid MIME type.*
* *The set of operations supported by the web service using HTTP methods (e.g., POST, GET, PUT or DELETE).* »

Dans la pratique, les services Web RESTful sont accessibles par une très large variété de technologies utilisées aujourd’hui sur Internet, telles qu’ASP.NET, JavaScript, Flash, Flex, PHP, Python, Ruby, Silverlight, Objective-C (iPhone iOS), Android, etc. Ceci revêt une importance certaine dans le contexte d’un Framework applicatif comme ODAF Openturf.

# Présentation rapide de Windows Azure



Windows Azure est une plateforme Cloud ouverte et flexible qui vous permet de générer, déployer et gérer rapidement des applications à travers un réseau global de centres de données gérés par Microsoft. Vous pouvez développer des applications à l'aide de n'importe quel langage, n'importe quel outil ou n'importe quelle infrastructure.

Ainsi, Windows Azure propose différents services et technologies pour l'intégration, le déploiement, la mise à l'échelle et la gestion d’applications ou de composantes de celles-ci dans le Cloud en s’inspirant du modèle qu’offre la plateforme Windows Server pour l’entreprise.

Cette plateforme comprend un socle d’exécution, Windows Azure, que l’on peut considérer comme l’équivalent d’un système d'exploitation dans le Cloud. De la même manière que l’on développe des applications pour Windows Server, on peut désormais développer des applications Cloud pour Windows Azure qui se charge de les exécuter.

Dans sa dimension Plateforme à la demande ou PaaS (Platform as-a-Service), Windows Azure est un système d'exploitation pour les services Cloud qui fait office d'environnement de développement, d'hébergement de services et de gestion de services pour la plateforme Windows Azure. Windows Azure vous fournit ainsi une plateforme Cloud flexible qui satisfera les besoins applicatifs. Cela vous permet d'héberger et de faire évoluer de façon fiable le code d'application dans les rôles de calcul.

Vous pouvez stocker les données à l'aide de bases de données relationnelles SQL, de magasins de tables NoSQL et de magasins de blob non structurés, et éventuellement utiliser les services Hadoop et Business Intelligence (BI) pour les explorer. Ainsi par exemple, SQL Azure est une base de données relationnelle hébergée dans les centres de données Microsoft. Sa maintenance et sa redondance sont assurées directement par Microsoft. Utilisant les protocoles normalisés de SQL Server, vous pouvez vous appuyer dessus indifféremment depuis une application située dans l’entreprise ou dans le Cloud. Largement compatible avec Microsoft SQL Server 2008 (R2), SQL Azure permet de répliquer et d’enrichir les bases de données d’entreprise de façon simple.

Vous pouvez tirer parti des fonctions de messagerie fiables de Windows Azure pour activer les applications distribuées évolutives, et fournir des solutions hybrides exécutées sur un Cloud et dans l'environnement d'entreprise local. Les services de mise en cache en mémoire distribuée et du réseau de distribution de contenu de Windows Azure vous permettent de réduire la latence et de garantir des performances d'application optimales n'importe où dans le monde.

Windows Azure vous permet d'utiliser n'importe quel langage, n'importe quelle infrastructure ou n'importe quel outil pour développer des applications. Les fonctionnalités et services sont exposés à l'aide de protocoles REST ouverts. Les bibliothèques clientes de Windows Azure sont disponibles pour plusieurs langages de programmation et sont publiées suivant les termes d'une licence source ouverte et hébergée sur GitHub.

Windows Azure fournit une infrastructure à la demande ou IaaS (Infrastructure as-a-Service) qui évolue et s'adapte aux nouveaux besoins d’une organisation. Windows Azure permet d'ajouter des réseaux virtuels et des ordinateurs virtuels Windows Server et Linux en quelques minutes seulement et d'adapter votre utilisation au fur et à mesure de l'évolution de vos besoins.

Pour de plus amples détails sur les différentes possibilités que vous offre Windows azure, vous pouvez consulter la vue d’ensemble proposée à l’adresse <http://www.windowsazure.com/fr-fr/solutions/>.

Windows Azure est actuellement commercialisé dans 41 pays. Vous pouvez accéder à Windows Azure avec un paiement à l’usage, sans engagement, ou avec des offres d’abonnements comme décrit à l’adresse <http://www.microsoft.com/france/windows-azure/Offres.aspx>, ou encore dans le cadre d’un accord d’entreprise Microsoft.

Au-delà de ces éléments, vous pouvez vous appuyer également sur les ressources mises à disposition par Microsoft Platform Ready (MPR) à l’adresse <http://microsoftplatformready.com/fr>. MPR a été conçu pour vous fournir les éléments dont vous avez besoin pour planifier, développer, tester et commercialiser vos solutions sur la plateforme Microsoft en entreprise et/ou dans le Cloud.

Les données ouvertes supposent une plateforme capable d’héberger un nombre croissant de données (et d’applications), totalement sécurisée, accessible à tous et tout type de plateforme ou environnement, et en mesure de supporter de forts pics de trafic.

Dans un tel contexte, l’utilisation de la plateforme Windows Azure pour ce kit de démarrage présente de multiples bénéfices :

* Comme évoqué ci-avant, la plateforme prend en charge une variété de normes et standards, de langages et de protocoles offrant ainsi une réelle interopérabilité. Jean Paoli en charge de la stratégie en matière d’interopérabilité chez Microsoft revient sur les éléments d’interopérabilité d’une plateforme Cloud dans le billet [Interoperability Elements of a Cloud Platform Outlined at OSCON](http://blogs.msdn.com/b/interoperability/archive/2010/07/22/interoperability-elements-of-a-cloud-platform-outlined-at-oscon.aspx)[[22]](#footnote-22) ;
* Comme fournisseur de services, Microsoft doit se conformer aux exigences réglementaires des entités gouvernementales pour les juridictions où la plateforme Windows Azure est opérée pour en assurer la conformité et gérer les risques en matière de sécurité. Windows Azure, qui fonctionne dans l'infrastructure de [Microsoft Global Foundation Services (GFS)](http://www.globalfoundationservices.com/)[[23]](#footnote-23) [certifiée ISO 27001](http://www.bsiamerica.com/en-us/Assessment-and-Certification-services/Management-systems/Certificate-and-Client-Directory-search/Search/Search-Results/?pg=1&licencenumber=IS+533913&searchkey=companyXeqXmicrosoft)[[24]](#footnote-24), est également [certifiée ISO 27001](http://www.bsiamerica.com/en-us/Assessment-and-Certification-services/Management-systems/Certificate-and-Client-Directory-search/Search/Search-Results/?pg=1&licencenumber=IS+577753&searchkey=licenceXeqX577753)[[25]](#footnote-25).

La norme ISO 27001 est reconnue dans le monde entier comme l'une des principales normes internationales en matière de gestion de la sécurité et des exigences associées. Dans le même temps, il est important de noter que Windows Azure va au-delà d’ISO 27001 avec l’application de [contrôles de sécurité additionnels](http://blogs.technet.com/b/gfs/archive/2009/06/16/response-to-question-about-sas-70-objectives.aspx)[[26]](#footnote-26).

Par ailleurs, Microsoft est signataire du [Safe Harbor](http://www.export.gov/safeharbor/eu/eg_main_018365.asp)[[27]](#footnote-27) et s'engage à remplir toutes ses obligations en vertu du Framework Safe Harbor.

D’autres certifications de l'industrie ont été obtenues ou sont en cours d’évaluation pour la plateforme. Pour de plus amples informations, nous vous invitons à consulter les ressources du [Centre de gestion de la confidentialité Windows Azure](http://www.windowsazure.com/fr-fr/support/trust-center/)[[28]](#footnote-28) et notamment le document [Réponse standard pour les demandes d'information, Sécurité et Respect de la vie privée](http://www.microsoft.com/en-us/download/details.aspx?id=26647)[[29]](#footnote-29) ;

* Une quantité illimitée d’informations publiques (données ouvertes) peut être hébergée dans le Cloud sans matériel serveur supplémentaire et gestion additionnelle. Les informations publiques deviennent encore plus précieuses avec une plus grande pertinence locale. Par exemple, les villes et les collectivités (région, conseil général, communauté urbaine, etc.) génèrent des données riches sur l’espace public (voirie, assainissement), le transport et ses perturbations, les espaces naturels (parcs, arbres), les projets de construction, etc. ;
* Le Cloud est hautement évolutif (*scalable*) et vous ne payez seulement que pour ce qui est utilisé. Ceci le rend idéal pour l'hébergement d’informations publiques qui peuvent susciter des volumes de transactions variés en fonction de l’intérêt porté. Windows Azure propose une approche économique des plus attractives pour démarrer rapidement et mettre à disposition des informations publiques ;
* [Windows Azure Marketplace](https://datamarket.azure.com/)[[30]](#footnote-30) permet aux entités publiques (gouvernement, ministères, collectivités locales et territoriales, mairies, services déconcentrés de l'Etat, etc.) de facturer les éditeurs d’applications pour la réutilisation et la consommation des informations publiques mises à disposition et aux éditeurs d’applications de facturer leurs utilisateurs pour les services à valeur ajoutée offerts au-dessus de ces données.

# Configuration requise pour le Kit de démarrage ODAF Openturf

Cette section s’intéresse aux différents prérequis du kit de démarrage ODAF Openturf vis-à-vis de la mise en œuvre de son socle « back-end » et de l’application client Web riche.

Les prérequis spécifiques aux environnements mobiles Android, iPhone et Windows Phone sont abordés dans les documents Déployer le kit de démarrage Android, Déployer le kit de démarrage ODAF Openturf pour iPhone et Déployer le kit de démarrage ODAF Openturf pour Windows Phone qui accompagnent le kit de démarrage. Vous pouvez vous reporter à ces trois documents pour une configuration complète de vos environnements de développement.

## Configuration système

Le Kit de démarrage ODAF Openturf a été conçu pour pouvoir être déployé notamment sur la plateforme Windows Azure. Les pré-requis exigés pour le déploiement de ce Kit de démarrage dans Windows Azure sont donc ceux de cet environnement.

Remarque : Ces prérequis avec les correctifs logiciels que cela suppose, peuvent bénéficier d’une installation automatisée de l’ensemble via le programme d’installation de la plateforme Web Microsoft 3.0. Celui-ci simplifie le téléchargement, l'installation et la tenue à jour des composants de la plate-forme Web Microsoft, notamment Internet Information Services (IIS) et ses extensions, SQL Server Express, .NET Framework, Visual Web Developer, etc. Il permet également d'installer des applications Web ASP.NET et PHP open source populaires.

[](http://www.microsoft.com/web/downloads/platform.aspx) Le programme d’installation de la plateforme Web Microsoft doit être installé préalablement depuis <http://www.microsoft.com/web/downloads/platform.aspx>.

En termes de systèmes d’exploitation, il s’agit en particulier de  :

* Microsoft Windows Vista SP2 (32 bits ou 64 bits) ou Windows Server 2008 SP2 (32 bits ou 64 bits) avec le [correctif logiciel KB971842 Correction des métadonnées WCF lors de l'hébergement derrière un programme d'équilibrage de charge](http://support.microsoft.com/kb/971842)[[31]](#footnote-31),
* Microsoft Windows 7 (32 bits ou 64 bits) ou Windows Server 2008 R2 (64 bits) avec le [correctif logiciel cumulatif KB977420 pour Windows Communication Foundation](http://support.microsoft.com/kb/977420)[[32]](#footnote-32).

Remarque : Certaines fonctionnalités de Windows Server AppFabric, tels que le démarrage automatique, fonctionnent uniquement avec Windows 7 et Windows Server 2008 R2.

Les éléments suivants doivent être installés :

* Microsoft Internet Information Services (IIS) 7.0 ou 7.5 (avec ASP.NET, Activation WCF http, Contenu statique, Console de gestion IIS et éventuellement CGI).
* Microsoft .NET Framework 4.0.

Remarque : Windows Azure requiert Microsoft .NET Framework 3.5 (ou ultérieur).

* Microsoft Visual Studio 2010 (ou Microsoft Visual C# 2010 Express Edition) avec :
* [Correctif logiciel KB983301 Activation de Windows Azure IntelliTrace sur les systèmes d'exploitation 32 bits](http://archive.msdn.microsoft.com/KB983301)[[33]](#footnote-33) (non requis pour les systèmes x64) ;
* [Module de réécriture d'URL d'IIS 7.0](http://www.iis.net/download/URLRewrite)[[34]](#footnote-34).
* [Windows PowerShell 2.0](http://go.microsoft.com/fwlink/?LinkId=192992)[[35]](#footnote-35) (non requis pour Windows 7 et Windows Server 2008 R2).
* [Kit de développement Windows Azure 1.4 (Refresh) et Windows Azure Tools pour Microsoft Visual Studio v1.6 (novembre 2011)](http://www.microsoft.com/windowsazure/sdk/)[[36]](#footnote-36).
* [Kit de développement Silverlight 4](http://www.silverlight.net/getstarted/)[[37]](#footnote-37).



Le Microsoft .NET Framework 4.0 est le modèle de programmation de code managé pour la plateforme Windows. Il combine la puissance du .NET Framework 2.0 avec de nouvelles techniques pour le développement d’applications qui communiquent très facilement au-delà des frontières technologiques, et qui sont capables de prendre en charge un large éventail de processus métiers. Parmi ces innovations se trouve notamment WCF (Windows Communication Foundation).

Pour des informations complémentaires sur le Microsoft .NET Framework, veuillez consulter le Centre de développement MSDN éponyme dédié à l’adresse <http://msdn.microsoft.com/fr-fr/netframework/aa663324.aspx>.

Pour des informations complémentaires sur la technologie Silverlight, veuillez consulter le Centre de développement MSDN dédié à l’adresse <http://msdn.microsoft.com/fr-fr/silverlight/bb187358>.

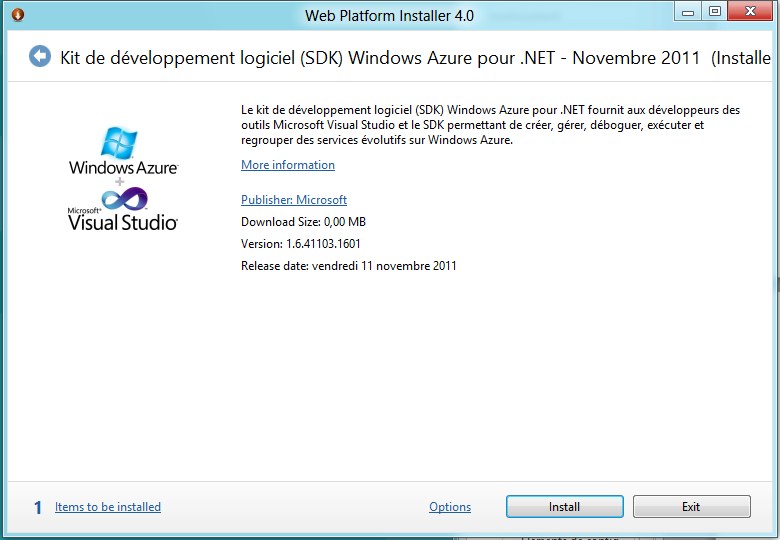
Le Kit de démarrage repose sur la version 4.0 du Microsoft .NET Framework. Le package redistribuable Microsoft .NET Framework 4.0 est disponible par téléchargement gratuit à l’adresse <http://www.microsoft.com/downloads/fr-fr/details.aspx?FamilyID=9CFB2D51-5FF4-4491-B0E5-B386F32C0992>.

Vous pouvez télécharger Visual C# 2010 Express Edition gratuitement sur  
<http://www.microsoft.com/express/Downloads/#2010-Visual-CS>.

Le package Windows Azure Tools pour Microsoft Visual Studio, qui comprend le kit de développement Windows Azure, étend Microsoft Visual Studio 2010 pour permettre la création, la configuration, la génération, le débogage, l'exécution, le packaging et le déploiement d'applications et de services Web évolutifs sur Windows Azure.

Le Kit de développement Windows Azure 1.6 est disponible en téléchargement à l’adresse <http://www.windowsazure.com/fr-fr/develop/net/>.

Au-delà d’une installation manuelle (<http://www.microsoft.com/downloads/fr-fr/details.aspx?FamilyID=E4635D95-61CE-415E-AE92-3004856C08E2&amp;amp;displaylang=fr>) avec les prérequis et correctifs logiciels que cela suppose, est proposée une installation automatisée de l’ensemble via le programme d’installation de la plateforme Web Microsoft 4.0 depuis l’adresse <http://www.microsoft.com/web/gallery/install.aspx?appid=WindowsAzureToolsVS2010>. Le Kit de développement Windows Azure 1.6 pour .NET peut être ajouté dans la liste des produits et installés.



La documentation du Kit de développement Windows Azure 1.6 pour .NET sous forme de fichier .chm est disponible à l’adresse <http://download.microsoft.com/download/3/3/2/3321A9FA-64C3-463F-981A-4E17FC29B15B/WindowsAzureSDK.chm>.

## Comment installer les composants système nécessaires

Vous trouverez ci-dessous la procédure d’installation succinte pour la mise en place d’un environnement de développement et de test pour le Kit de démarrage sur une plateforme Windows 7 ou Windows Server 2008 R2 :

1. Installer Internet Information Services (IIS) 7.5,
2. Installer le Microsoft Framework .NET 4.0,
3. Installer Microsoft Visual Studio 2010,
4. Enfin utiliser le programme d’installation de la plateforme Web Microsoft 4.0 pour installer le Kit de développement Windows Azure 1.6 pour .NET (Novembre 2011), avec leurs prérequis et correctifs logiciels associés, ainsi que le kit de développement Silverlight 4.

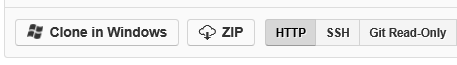
# Obtention des sources du kit de démarrage ODAF Openturf

La solution communautaire ODAF Openturf est disponible sur la forge GitHub à l’adresse <https://github.com/openlab/Openturf>.

Les solutions communautaires évoluant avec l’investissement de la communauté, il est souhaitable de vos assurer que vous disposez de la dernière version, ne serait-ce que pour bénéficier, le cas échéant, de l’ensemble des correctifs.

[image](https://github.com/openlab/DataLab)

Pour télécharger les sources depuis la forge GitHub, le moyen le plus simple et rapide est de cliquer directement sur le bouton ZIP.



S’il constitue le moyen le plus simple, les autres options se révèlent plus adaptées pour mettre en place une gestion de versions.



A ce titre, nous vous invitons à consulter les billets [OGDI DataLab sur GitHub ! Impliquez-vous dans la communauté](http://blogs.msdn.com/b/ogdifrance/archive/2012/05/14/ogdi-datalab-sur-github-impliquez-vous-dans-la-communaut-233.aspx)[[38]](#footnote-38) et [GitHub for Windows](http://blogs.msdn.com/b/ogdifrance/archive/2012/06/14/github-for-windows.aspx)[[39]](#footnote-39), tous deux présents sur le blog MSDN [Open Data France](http://aka.ms/OpenDataFrance)[[40]](#footnote-40).

Par ailleurs, L’environnement Visual Studio 2012 propose depuis l’[Update 2](http://www.microsoft.com/en-us/download/details.aspx?id=38188)[[41]](#footnote-41) avec les [outils Visual Studio pour Git](http://visualstudiogallery.msdn.microsoft.com/abafc7d6-dcaa-40f4-8a5e-d6724bdb980c)[[42]](#footnote-42) une intégration directe avec GitHub. L’article [Integrating and Using Github in Visual Studio 2012](http://www.codeproject.com/Articles/581907/IntegratingplusandplusUsingplusGithubplusinplusVis)[[43]](#footnote-43) vous propose un guide pas-à-pas dans ce contexte.

Quelle que soit l’approche retenue, et une fois, les sources téléchargées, nous vous invitons à passer à la section suivante.

# Contenu du Kit de démarrage ODAF Openturf

Le Kit de démarrage ODAF Openturf s’articule autour de cinq composants :

1. **Base de données**. Le Kit de démarrage ODAF Openturf utilise une base de données Microsoft SQL Server 2008, Microsoft SQL Server 2008 R2 ou Microsoft SQL Azure afin de stocker les données nécessaires à son exécution, à savoir les commentaires, les données et les comptes des utilisateurs. Les scripts de création de la base de données, de réinitialisation des données et de suppression de la base de données sont fournis avec le kit.
2. **Site Web ASP.NET MVC**. Le site Web expose une couche de services RESTful consommés par les applications modèle de la solution pour lire les flux de données KML, et interagir avec les données utilisateurs stockées dans la base de données (tags, commentaires, notes, profils, etc.).

Le site Web suit l’architecture Modèle-Vue-Contrôleur (MVC). L’architecture MVC est une façon d'organiser une interface graphique, mais également et surtout une façon d’architecturer un code source en respectant le principe de responsabilité unique, le principe de séparation des préoccupations et répondre aux problématiques de testabilité du code. Ce paradigme divise l’interface de l’application en un modèle (modèle de données), une vue (présentation, interface utilisateur) et un contrôleur (logique de contrôle, gestion des événements, synchronisation), chacun ayant un rôle précis dans l'interface.

Une telle architecture s’adapte très bien à la création de services RESTful. En effet, une URL permet d’accéder à une ressource, c’est-à-dire, dans le cas, présent un contrôleur qui va invoquer une action côté serveur et retourner en guise de vue des données au format JSON aux consommateurs du service, en l’occurrence ici les modèles suivants d’application.

1. **Application Silverlight.** Le cœur de la solution est l’application modèle Silverlight centrée autour d’une carte Bing Cartes sur laquelle les points de données vont être affichés.

L’application est hébergée sur le site Web ASP.NET MVC et est téléchargée par le navigateur côté client pour s’exécuter dans la machine virtuelle Silverlight. Le composant dialogue avec le site Web ASP.NET MVC via les services RESTful précédents afin de lire les données des flux KML configurés et restituer les points correspondants sur la carte Bing Cartes.

[](http://twitter.com) L’utilisateur a la possibilité de filtrer les sources de données qu’il souhaite afficher sur sa carte et de s’authentifier via son [compte Twitter](https://twitter.com/signup)[[44]](#footnote-44) dans le cas où il voudrait commenter ou noter des données, où même créer ses propres points de données. Il convient de noter que les commentaires sont restitués sous Twitter sous forme de tweets géo-localisés.

1. **Application Android.** Une application client mobile pour Android est également fournie dans le cadre du Framework ODAF Openturf. Celle-ci dialogue également avec le site Web ASP.NET MVC via les services RESTful qu’il expose pour consommer les données des flux de données KML.

L’application utilise Google Maps comme technologie de cartographie et fournit la possibilité de se connecter à Twitter et de commenter les données.

1. **Application iPhone.** Une application client mobile pour iPhone est également fournie dans le cadre du Framework ODAF Openturf. Celle-ci dialogue également avec le site Web ASP.NET MVC via les services RESTful qu’il expose pour consommer les données des flux de données KML.

L’application utilise Bing Cartes comme technologie de cartographie et fournit la possibilité de se connecter à Twitter et de commenter les données.

1. **Application Windows Phone.** Une application client mobile pour Windows Phone est également fournie dans le cadre du Framework ODAF Openturf. A l’instar des applications Silverlight, Android et iPhone, cette application dialogue avec le site ASP.NET MVC pour consommer les données des flux KML.

[](http://twitter.com) [](http://www.facebook.com) Cette dernière utilise Bing Cartes comme technologie de cartographie et fournit la possibilité de se connecter à Twitter ou Facebook pour commenter un point de donnée. Cette version autorise également la prise de photos lorsque l’on se trouve à proximité d’un point de donnée au lieu de simples commentaires.

[](http://twitpic.com/) Ainsi, le contenu partagé est plus riche et plus concret aussi. Ces photos sont soient partagés dans son album Facebook, soit téléchargées sur TwitPic dans le cas où l’utilisateur choisit de partager via Twitter.

# Organisation du code source du Kit de démarrage

L’intégralité du code source des sites Web, services et composants ODAF Openturf est située dans le sous-répertoire *Source* du répertoire d’extraction du Kit de démarrage ODAF Openturf.

Compte tenu de ce qui précède, ce répertoire contient 8 sous-dossiers principaux :

1. Le dossier *Database* qui correspond à la couche de données de la solution ODAF Openturf. Il contient 2 sous-dossiers :
   1. Le sous-dossier *ODAF.Data* contient le projet éponyme qui est la couche d’accès aux données de la solution.

Remarque : Ce projet utilise la bibliothèque [SubSonic](http://subsonicproject.com/)[[45]](#footnote-45), qui est un générateur de code basé sur les modèles (*templates*) [T4 (*Text Template Transformation Toolkit*)](http://www.olegsych.com/2007/12/text-template-transformation-toolkit/)[[46]](#footnote-46) pour l’interfaçage avec une base de données.

Les modèles T4 sont des fichiers qui mélangent des blocs de texte avec une logique de contrôle (en C# ou Visual Basic) afin de générer des fichiers de texte en sortie. Ils sont utilisés par SubSonic pour aller lire le schéma d’une base de données et générer les fichiers .cs qui contiennent les classes en C# pour se connecter et interagir avec celle-ci.

La bibliothèque SubSonic permet de s’interfacer avec des bases de données SQL Server, Oracle, MySql ou SQLite. Elle tire pleinement parti de la syntaxe [LINQ (*Language-INtegrated Query*)](http://msdn.microsoft.com/en-us/netframework/aa904594)[[47]](#footnote-47) et se fonde sur le patron de conception (*pattern*) « [Active Record](http://en.wikipedia.org/wiki/Active_record_pattern)[[48]](#footnote-48) ».

* 1. Le sous-dossier *scripts* contient quant à lui les scripts SQL de création de la base de données

1. Le dossier *Libraries* comportant les différents projets d’appoints référencés en particulier par le projet *ODAF.WebSite.Mvc* de la solution notamment pour se connecter à diverses API REST utilisées par la solution. Il s’agit en l’occurrence des projets suivants :
   1. Projet *BitlyDotNet* : « wrapper » d’interfaces programmatiques (API) écrit en C# pour le service bit.ly pour raccourcir les URLs. Vous pouvez suivre l’actualité de ce [projet bitly-dot-net](http://code.google.com/p/bitly-dot-net/)[[49]](#footnote-49) sous [licence libre MIT](http://www.opensource.org/licenses/mit-license.php)[[50]](#footnote-50) sur le blog de son [auteur](http://blog.mikecouturier.com/search/label/bitly-dot-net)[[51]](#footnote-51) ;

Remarque : Une clé d’API obtenue sur le [site bit.ly](http://bit.ly/account/)[[52]](#footnote-52) est nécessaire.

* 1. Projet *NUrl* : code exemple de l’ouvrage [Professional Twitter Development: With Examples in .NET 3.5](http://www.wrox.com/WileyCDA/WroxTitle/Professional-Twitter-Development-With-Examples-in-NET-3-5.productCd-0470531320.html)[[53]](#footnote-53) aux éditions Wrox référencé dans le projet suivant *OAuthLibrary* ;
     + 1. [](http://www.wrox.com/WileyCDA/WroxTitle/Professional-Twitter-Development-With-Examples-in-NET-3-5.productCd-0470531320.html)
  2. Projet *OAuthLibrary* : code exemple issu du chapitre § 6 Basic Authentication and OAuth de l’ouvrage Professional Twitter Development: With Examples in .NET 3.5 pour la mise en œuvre de la spécification [OAuth](http://oauth.net)[[54]](#footnote-54).

Remarque : A titre de complement, vous pouvez consulter les articles Wrox [Twitter Development: The OAuth Specification](http://p2p.wrox.com/content/articles/twitter-development-oauth-specification)[[55]](#footnote-55), [Twitter Development: The OAuth Workflow](http://p2p.wrox.com/content/articles/twitter-development-oauth-workflow)[[56]](#footnote-56) et [Twitter Development: Using OAuth to Authenticate Against the Twitter API: Walkthroughs](http://p2p.wrox.com/content/articles/twitter-development-using-oauth-authenticate-against-twitter-api-walkthroughs)[[57]](#footnote-57).

* 1. Projets *NET.SyntaxC4.RedBitSoftware.ODAF.\** implémentant le célèbre moteur de recherche [Lucène](http://lucene.apache.org/) pour indexer les flux KML et rechercher les données.
  2. Projet *Dependencies* permettant de faciliter la résolution des dépendances en incorporant un ensemble de fichiers .dll

1. Le dossier *ODAF.Azure* correspondant au projet de déploiement de la solution ODAF Openturf dans Windows Azure, il y a un seul rôle Web qui correspond au projet *ODAF.Website.Mvc* ;
2. Le dossier *ODAF.Android* correspondant au projet Android du Framework ODAF Openturf ;
3. Le dossier *ODAF.iPhoneApp* correspondant au projet iPhone du Framework ODAF Openturf ;
4. Le dossier *ODAF.SilverlightApp* correspondant au projet Silverlight éponyme du Framework ODAF Openturf, et donc à l’application principale permettant d’afficher la carte Bing Cartes avec les points de données affichés;
5. Le dossier *ODAF.Website* correspondant au projet ASP.NET MVC *ODAF.Website.Mvc* qui expose la couche de services RESTful consommés par les clients pour lire les flux de données KML et interagir avec les données stockées dans la base de données ;
6. Le dossier *ODAF.WindowsPhone* correspondant au projet éponyme Windows Phone de la solution ODAF Openturf.

La suite de cette section décrit les projets/éléments de configuration constituants de ces différents sous-dossiers de façon à mieux cerner la solution ainsi proposée par le kit de démarrage ODAF Openturf et ainsi vous l’approprier, nous l‘espérons, plus facilement en vue par exemple de sa personnalisation (légère ou avancée).

Tous les sites, services et composants afférents sont écrits en C#. Ils sont disponibles pour la version v4.0.30319 du Microsoft Framework .NET. Les fichiers de solution associés et présents dans ces dossiers sont destinés à l’environnement de développement Visual Studio 2010 (à l’exception des projets Android et iPhone) et sont regroupés dans la solution *ODAF.Website.sln* (à l’exception des projets Android et iPhone).

## Dossier Database

Cette première composante de la solution correspond à la couche d’accès aux données du Framework ODAF Openturf. Comme évoqué précédemment, ce dossier est composé de 3 sous-dossiers correspondants aux scripts SQL de création de la base de données de la solution, à la bibliothèque *SubSonic* et au projet *ODAF.Data* qui contient le code d’accès à la base de données.

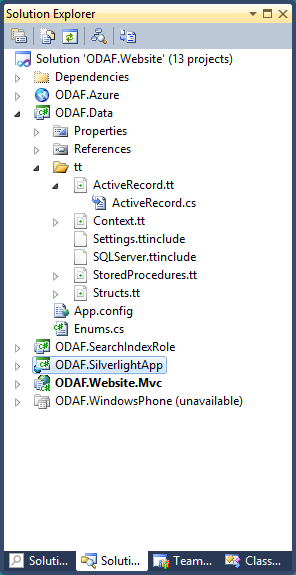
### Scripts SQL

Le dossier *scripts* contient les scripts SQL suivants :

| **fichier** | **Description** |
| --- | --- |
| ***odaf.sql*** | Ce script de création de la base de données permet de créer toutes les tables nécessaires et d’insérer des données par défaut (dans les tables *OAuthClientApp* et *PointDataLayer*) ainsi que des données qui n’ont pas vocation à changer à priori (tables *UserRole* et *UserAccess*) |
| ***odaf.wipe.tables.sql*** | Ce script permet d’effacer les données qui ont été partagées par les utilisateurs, à savoir les commentaires, les notes, les tags et les profils utilisateurs. |
| ***odaf.drop.tables.sql*** | Ce script supprime toutes les tables de la base de données. |

Le sous-dossier *tt* contient quant à lui les modèles (*templates*) T4 utilisés par le projet *ODAF.Data* pour générer le code de la couche d’accès aux données.

### Organisation projet ODAF.Data

* + - * 1. Le projet *ODAF.Data* est organisé de la façon suivante :
        2. 

| **Elément** | **Description** |
| --- | --- |
| ***Dossier tt*** | Ce dossier contient tous les modèles T4 qui vous permettent de générer le code d’accès à la base de données. Nous abordons plus spécifiquement la procédure de génération du code dans le document Déployer le kit de démarrage ODAF Openturf dans Windows Azure. |
| ***App.config*** | Ce fichier de configuration du projet contient uniquement la chaine de connexion à la base de données de la solution ODAF Openturf. Il est nécessaire que cette chaine de connexion pointe vers une base de données valide afin de pouvoir générer le code d’accès à la base de données. |
| ***Enum.cs*** | Ce fichier constitue une énumération en C# correspondants aux rôles et aux permissions d’accès des utilisateurs. |

### Eléments de configuration

Le seul paramètre à configurer dans le projet *ODAF.Data* est la chaîne de connexion à la base de données qui est située dans le fichier *App.config*.

Remarque importante : Tous les paramètres de la solution qui doivent être configurés sont précédés de la chaîne de caractères *CHANGEME.* Il devient alors très facile de rechercher toutes les occurrences de cette chaîne de caractères dans Visual Studio afin de compléter tous les paramètres.

| **Paramètre** | **Description** |
| --- | --- |
| ***ODAF*** | Chaine de connexion pointant vers la base de données de la solution ODAF Openturf.  Pour vous aider, il existe en ligne le site Web <http://www.connectionstrings.com/> qui vous propose les squelettes des différentes chaines de connexion ADO.NET. |

## Dossier *Libraries*

Le dossier *Libraries* contient les projets *BitlyDotNET*, *MvcFutures*, *NUrl* et *OAuthLibrary*.

Ces projets ne nécessitent aucune configuration particulière.

| **Projet** | **Description** |
| --- | --- |
| ***BitlyDotNET*** | Ce projet permet de s’interfacer avec l’API du service Bit.ly (raccourcisseur d’URL). Ce projet est notamment utilisé quand on commente des données sur Twitter pour pouvoir économiser des caractères dans le tweet. |
| ***NUrl*** | Ce projet fournit des classes permettant de simplifier la gestion du protocole http notamment dans l’envoi requêtes de façon asynchrone. |
| ***OAuthLibrary*** | Ce projet permet de s’authentifier à l’aide du protocole OAuth auprès d’un service distant. Ce projet est notamment utilisé dans le cadre de l’authentification à Twitter par le projet *ODAF.Website.Mvc*. |
| ***Net.SyntaxC4.RedBitSoftware.ODAF.Search*** | Ce projet implémente le célèbre moteur de recherche Lucène mais basé sur Azure en l’occurrence sur un compte de stockage. |

## Dossier *Dependencies*

Le dossier *Dependencies* contient un ensemble d’assemblies (dll) permettant une résolution simplifiée des références de la solution ODAF Openturf.

Ce projet ne nécessite aucune configuration particulière.

## Dossier *ODAF.Azure*

Ce projet standard Windows Azure est configuré pour supporter le rôle *ODAF.Website.Mvc* (Cf. ci-dessous). Ce projet contient la configuration de ce rôle unique.

Il inclut le fichier de définition de service *ServiceDefinition.csdef* qui contient les métadonnées nécessaires à Windows Azure pour comprendre les exigences de l’application, tels que les deux rôles utilisés, leur niveau de confiance, les points de terminaison exposés par chaque rôle, les exigences de stockage local et les éventuels certificats utilisés par les rôles. La définition de service établit également les paramètres de configuration spécifiques à l'application.

Le fichier de configuration de service *ServiceConfiguration.cscfg* spécifie le nombre d'instances à exécuter pour le ou les rôles configurés dans le projet (ici un seul) et définit la valeur des paramètres de configuration définis dans le fichier de définition de service. Cette séparation entre la définition et la configuration de service vous permet de mettre à jour les paramètres d'une application en cours d'exécution en téléchargeant un nouveau fichier de configuration de service.

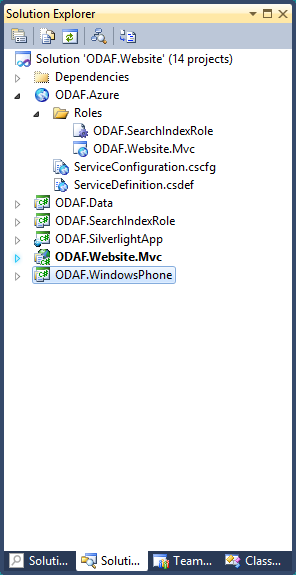
Le nœud **Roles** dans le projet de service Cloud vous permet de configurer quels rôles le service inclut (web, travailleur (*worker*) ou les deux) ainsi que les projets à associer à ces rôles. L’ajout et la configuration des rôles au travers de ce nœud **Roles** mettra à jour les fichiers *ServiceDefinition.csdef* et *ServiceConfiguration.cscfg* précédents.

Ce projet sert directement pour publier dans Windows Azure la solution complète ODAF Openturf.

Il n’y a pas de paramètres à configurer dans ce projet.

Remarque importante : La publication requiert un compte et une souscription active avec Windows Azure. Vous pouvez accéder à Windows Azure avec un paiement à l’usage, sans engagement, ou bien des forfaits comme décrit à l’adresse <http://www.microsoft.com/france/windows-azure/Offres.aspx>.

Ce projet est composé de la façon suivante :

* + - * 1. 

## Dossier *ODAF.Android*

Ce dossier contient le projet de l’application Android. Il s’agit donc d’une application écrite en Java et reposant sur le kit de développement Android 12. Il y a différents paramètres situés dans différents fichiers à configurer dans ce projet.

### Eléments de configuration

| **Fichier** | **Paramètres** |
| --- | --- |
| ***res/values/strings.xml*** | * ***app\_name***: nom de l’application affiché sur l’écran d’accueil. * ***odafWebsitebaseUrl***: URL de base de votre serveur ODAF Openturf, n’oubliez pas le « / » à la fin. * ***odafAppId*** : GUID de votre application enregistré dans la base de données SQL Azure d’ODAF Openturf. * ***latitude***: latitude initiale du point central de l’écran d’accueil. * ***longitude*** : longitude initiale du point central de l’écran d’accueil. * ***zoomLevel***:niveau de zoom initial du point central de l’écran d’accueil. * ***consumerkey***: (votre) clé de consommateur Twitter. * ***consumerSecret***: (votre) secret de consommateur Twitter. * ***twitterCommentSent*** : message qui est affiché à l’écran après l’envoi d’un tweet. |
| ***res/drawable- [hdpi|ldpi|mdpi]/icon.png*** | Icône de l’application. |

## Dossier *ODAF.iPhoneApp*

Ce dossier contient le projet de l’application iPhone. Il s’agit donc d’une application écrite en Objective-C et reposant sur le kit de développement iPhone 3.x. Il y a différents paramètres situés dans différents fichiers à configurer dans ce projet.

### Eléments de configuration

| **Fichier** | **Paramètres** |
| --- | --- |
| ***Classes/Constants.h*** | * ***kTwitterAppConsumerKey***: (votre) clé de consommateur Twitter. * ***kTwitterAppConsumerSecret***: (votre) secret de consommateur Twitter. * ***BING\_MAPS\_DEV\_KEY***:(votre) clé de développeur Bing Cartes. |
| ***Config/Configuration.plist*** | * ***TintColourRGB***: couleur des barres d’outils et de navigation de l’application au format RRGGBB hexadécimal. * ***ButtonColourGradientHighRGB***: couleur du gradient haut des boutons de l’application au format RRGGBB hexadécimal. * ***ButtonColourGradientLowRGB***: couleur du gradient bas des boutons de l’application au format RRGGBB hexadécimal. * ***WebAppUrl***: lien vers votre application Web. * ***ClientAppId*** : valeur du champ Guid de votre application dans la table *OAuthClientApp*. * ***ReachableUrl***:nom de domaine (sans le schéma url) du serveur pour tester la disponibilité. * ***MapInitialLatitude***:latitude du point central de l’écran d’accueil. * ***MapInitialLongitude***: longitude du point central de l’écran d’accueil. * ***MapInitialZoom***: niveau de zoom initial du point central de l’écran d’accueil. * ***MapCurrentLocationlZoom***:niveau de zoom quand l’application zoom pour montrer votre localisation courante*.* * ***MapViewCommentZoom***:niveau de zoom quand l’application zoom pour vous montrer le marqueur d’un point de données dont vous regardez les commentaires. |
| ***ODAF.iPhoneApp-Info.plist*** | * ***Bundle Identifier***:remplacez « *CHANGEME* » par le nom de votre entreprise --  *com.CHANGEME.${PRODUCT\_NAME:rfc1034identifier}*. * ***URL types/URL identifier***: ajoutez votre identifiant unique sous forme de chaine de caractère, généralement de la forme « *com.entreprise.produit* ». * ***URL types/URL Schemes***:paramètre doit être égal à la valeur du paramètre *AppName* dans le fichier *Web.config* du projet *ODAF.Website.Mvc* mais tout en minuscules. |
| ***Default.png*** | Ecran de chargement. Il s’agit d’un fichier .png de dimension 320x480 pixels. |
| ***icon.png*** | Icône de l’application. Il s’agit d’un fichier .png de dimension 57x57 pixels. |

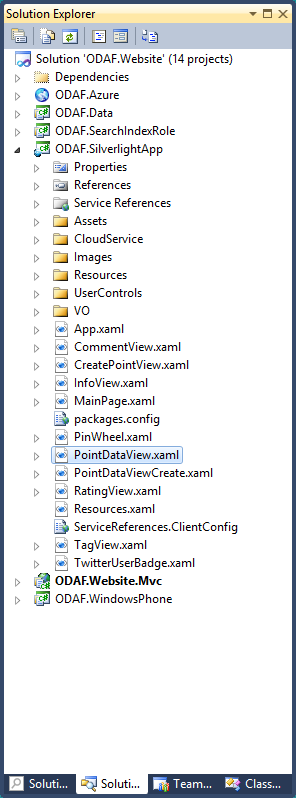
## Dossier *ODAF.SilverlightApp*

Le dossier *ODAF.SilverlightApp* contient le projet Silverlight du Framework ODAF Openturf.

Il s’agit d’un projet fondé sur la version 4 de Silverlight qui utilise donc les dernières technologies de Microsoft, notamment le [kit de développement (SDK) Bing Cartes pour Silverlight](http://www.microsoft.com/downloads/en/details.aspx?displaylang=en&FamilyID=beb29d27-6f0c-494f-b028-1e0e3187e830)[[58]](#footnote-58), le [mode étendu du contrôle Bing Cartes pour Silverlight](http://connect.microsoft.com/SilverlightMapModesBeta)[[59]](#footnote-59) (pour ajouter des modes de vues supplémentaires, tels que la vue « oiseau » et la vue « StreetSide »), ou encore la [boite à outils de Silverlight 4](http://silverlight.codeplex.com/)[[60]](#footnote-60) afin que vous puissiez disposer de nouveaux composants et nouvelles fonctionnalités pour vos développements et votre design.

### Organisation projet

Le projet est organisé de la façon suivante :



| **Dossier** | **Description** |
| --- | --- |
| ***Racine*** | A la racine du projet, vous retrouverez la page principale, c’est-à-dire *MainPage.xaml* ainsi que tous les contrôles utilisateurs Silverlight qui sont utilisés avec la carte Bing Cartes de la page principale pour interagir avec les points de données. |
| ***CloudService*** | Ce dossier contient toute la logique nécessaire pour se connecter aux services RESTful du projet *ODAF.Website.Mvc* avec notamment des classes pour interroger côté serveur les contrôleurs qui vont retourner les données au format JSON. |
| ***Images*** | Ce dossier contient comme son nom l’indique les images utilisées dans le reste de l’application pour styliser les contrôles. |
| ***UserControls*** | Ce dossier contient deux contrôles Silverlight de type fenêtre filles, qui jouent le rôle de boite de dialogues pour tweeter depuis la solution ODAF Openturf. Il contient également la classe *MapRegionControl* qui permet de représenter des polygones sur une carte Bing Cartes. |
| ***VO*** | Ce dossier contient les modèles de données utilisés par la logique applicative du projet. Il contient ainsi des classes pour représenter des flux KML, des points de données, des tweets ou des commentaires. |

### Eléments de configuration

Plusieurs paramètres sont à configurer dans ce projet. Comme évoqué précédemment une simple recherche dans Visual Studio de l’occurrence « *CHANGEME* » permet de les localiser.

| **Fichier** | **Paramètre** |
| --- | --- |
| ***CloudService\UserController*** | ***AppId*** (ligne 54) : valeur du champ *Guid* de votre application dans la table *OAuthClientApp*. |
| ***MainPage.xaml.cs*** | *Location loc = new Location(****latitude****,* ***longitude***) (ligne 62) : remplacez la latitude et la longitude actuellement renseignées par la latitude et la longitude de votre ville.  Une simple recherche d’une ville sur [Bing Cartes](http://www.bing.com/maps/) vous permet d’obtenir ses coordonnées juste en dessous de sa description dans le panneau de gauche. |
| ***MainPage.xaml.cs*** | *Location loc = new Location(****latitude****,* ***longitude****)* (ligne 212) : remplacez la latitude et la longitude actuellement renseignées par la latitude et la longitude de votre ville. |
| ***App.xaml*** | ***ApplicationId*** (ligne 11) : valeur de votre clé de compte développeur Bing Cartes. |
| ***Resources*** | *Localise l’ensemble du projet au travers de fichiers de ressources sélectionnés en fonction de la langue définie dans le navigateur.* |

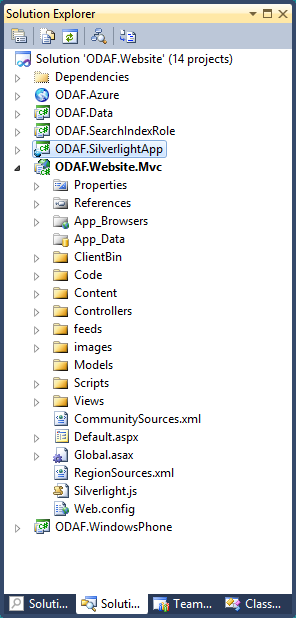
## Dossier *ODAF.Website*

Le dossier *ODAF.Website* correspond au projet *ODAF.Website.Mvc* qui expose une couche de services RESTful consommés notamment par le composant Silverlight précédemment évoqué.

Le projet expose notamment un service permettant de consommer à la base des flux KML configurés dans le projet sous forme de données JSON pour les clients du service, un service pour publier des commentaires dans la base de données, un service pour poster des tweets sur Twitter, un service pour ajouter des notes, etc.

### Organisation projet

Le projet est organisé de la façon suivante :



| **Dossier** | **Description** |
| --- | --- |
| ***ClientBin*** | Ce dossier contient le fichier XAP du projet Silverlight précédemment évoqué ; ce fichier est référencé par la vue par défaut du site pour afficher le composant Silverlight. |
| ***Code*** | Ce dossier contient 2 classes qui permettent pour l’une de représenter un commentaire utilisateur et pour l’autre de définir les prérequis en terme de d’autorisations pour qu’un utilisateur puisse effectuer une action (c’est une classe qui hérite de la classe *Attribute* du Framework .NET, elle est placée dans ce projet juste au-dessus de certains méthodes des contrôleurs). |
| ***Content*** | Ce dossier contient le fichier de style CSS du site. |
| ***Controllers*** | Ce dossier contient tous les contrôleurs du site, ce sont eux qui sont interrogés par les applications clientes et qui retournent les données demandés au format JSON. Ils jouent le rôle de services RESTful. |
| ***feeds*** | Ce dossier contient 2 flux KML de tests permettant de tester son service sans forcément avoir de flux KML à l’origine. |
| ***images*** | Ce dossier contient toutes les images du projet, notamment les images qui vont servir à représenter des points de données sur la carte Bing Cartes. Il faut donc prévoir une image par flux de données KML pour pouvoir les différencier. |
| ***Models*** | Ce dossier sert à contenir les modèles de données utilisés par le site Web (architecture MVC). Comme dans le cas présent, la couche d’accès aux données est externalisée dans le projet *ODAF.Data* qui est référencé par ce projet, ce dossier est vide. |
| ***Scripts*** | Ce dossier contient les scripts JavaScript utilisés par le site Web (JQuery et Ajax essentiellement). |
| ***Views*** | Ce dossier contient toutes les vues du site Web, notamment la vue */Home/Index.aspx* qui est la vue par défaut permettant d’afficher le composant Silverlight en plein écran. |

### Eléments de configuration

Ce projet contient différents éléments de configuration contenus pour la plupart dans le fichier de configuration *Web.config* à la racine du site Web.

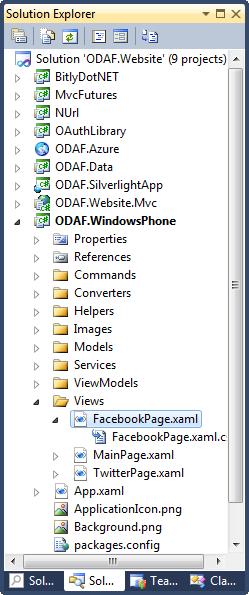
| **Fichier** | **Paramètre** |
| --- | --- |
| ***Web.config*** | * **AppName** (ligne 13) : nom de votre application dans ce paramètre. * **BitlyLogin** (ligne 16) : (votre) login pour l’API Bit.ly dans ce paramètre. * **BitlyAPIKey** (ligne 18) : (votre) clé pour l’API Bit.ly dans ce paramètre. * **DevMode** (ligne 25) : indique si vous êtes en mode développement (retire tous les contrôles d’authentification et d’autorisations si ce mode est effectif). * **AdminTwitterUser** (ligne 29) : Nom du compte twitter administrateur par défaut. * **TwitterAppId** (ligne 31) : Clé unique stockée en base de données permettant d’utiliser l’API twitter. * **ODAF** (ligne 35) : chaine de connexion à la base de données de votre solution ODAF Openturf. |
| ***Views\UserControls***  ***\AnalyticsUserControl.ascx*** | Ce contrôle utilisateur actuellement vide vous permet d’injecter du code qui sera présent dans toutes les pages du site Web. Ce qui peut être utile pour relever des statistiques ou des compteurs en termes d’utilisation. |

## Dossier *ODAF.WindowsPhone*

Le dossier *ODAF. WindowsPhone* correspond au projet *ODAF.WindowsPhone* qui est le projet mobile sur Windows Phone qui consomme (au même titre que l’application Silverlight, l’application Android ou l’application iPhone) les données exposées par la couche de services RESTful du site Web ODAF*.*

### Organisation projet

Le projet est organisé de la façon suivante :



| **Dossier** | **Description** |
| --- | --- |
| ***Commands*** | Le projet *ODAF.WindowsPhone* respectant l’architecture MVVM, le déclenchement d’un évènement par les interfaces utilisateurs est découplé de l’exécution de code métier associé par le biais de commandes. Une commande sert à faire le lien entre un évènement déclenché sur la vue et le code métier associé qui sera exécuté par le modèle logique de la vue. Le dossier *Commands* fournit la classe *DelegateCommand* qui est une implémentation de l’interface *ICommand* fournie par le Framework Silverlight. |
| ***Converters*** | Ce dossier contient des convertisseurs utilisés lors de liaisons de données entre le modèle logique de la vue et la vue. Ces convertisseurs servent par exemple à convertir un booléen en une valeur de l’énumération *Visibility* utilisée par les contrôles Silverlight. Ce qui encore une fois permet de découpler le modèle logique de la vue, de la technologie spécifique associé à la vue et augmente le potentiel de réutilisabilité du modèle logique de vue. |
| ***Helpers*** | Ce dossier sert à contenir des « helpers » divers et variés. Comme la classe *GeoMathHelper* par exemple. |
| ***Images*** | Ce dossier contient les icônes et images utilisées par le projet. |
| ***Models*** | Ce dossier contient les différents modèles de données représentant les données manipulées par l’application, par exemple les différents points sur la carte, les collections de points et les commentaires associés aux points. |
| ***Services*** | Ce dossier contient les différentes classes qui vont permettre de consommer les services utilisés par la solution, que ce soient les services exposés par le site ODAF Openturf notamment pour la gestion des commentaires ou bien le service Twitpic pour le chargement (*upload*) de photos depuis le smartphone. |
| ***ViewModels*** | Ce dossier contient les classes représentant les modèles logiques des vues associées. La logique applicative de ces classes manipule les modèles de données (dossier *Models*) et les restitue aux vues qui se chargeront de mettre en forme l’affichage. |
| ***Views*** | Ce dossier contient les vues de l’application (vue principale, vue Facebook et vue Twitter). Ces vues ne contiennent presque aucun code dans leur fichier « *code-behind* » associé, et pour cause, la logique applicative étant dans les classes contenant le modèle logique des vues (dossier *ViewModels*). A l’exception des contrôles de type « *ApplicationBar* » qui sont des contrôles un peu spécifiques, puisque ne dérivant pas de la classe *FrameworkElement* du Framework Silverlight. (et ne disposant donc pas de mécanisme de liaison de données par conséquent). |

### Eléments de configuration

Ce projet contient différents éléments de configuration contenus dans le fichier *App.xaml.cs* à la racine du projet.

| **Fichier** | **Paramètre** |
| --- | --- |
| ***App.xaml.cs*** | * ***AppId*** (ligne 23) : valeur du champ *Guid* de votre application dans la table *OAuthClientApp*. * ***BingMapsId*** (ligne 24) : (votre) clé de développeur Bing Cartes. * ***CenterLocation*** (ligne 25) : position initiale de votre carte. * ***FacebookAlbumName*** (ligne 26) : nom de l’album photo dans lequel seront uploadées les photos des utilisateurs lorsqu’ils les partageront sur Facebook. * ***FacebookAlbumDescription*** (ligne 27) : description de l’album photo dans lequel seront uploadées les photos des utilisateurs lorsqu’ils les partageront sur Facebook. * ***FacebookAppId*** (ligne 28) : identifiant de votre application sur votre compte développeur Facebook. * ***FacebookExtendedPermissions*** (ligne 29) : tableau contenant la liste des permissions d’accès demandées par votre application à l’utilisateur Facebook. * ***OdafWebsiteUrl*** (ligne 30) : URL de votre site Web ODAF Openturf. * ***OdafWebsiteUrl*** (ligne 30) : URL de votre site Web ODAF Openturf. * ***OdafWebsiteUrl*** (ligne 30) : URL de votre site Web ODAF Openturf. * ***TwitterConsumerKey*** (ligne 32) : (votre) clé de consommateur Twitter. * ***TwitterConsumerSecret*** (ligne 33) : (votre) secret de consommateur Twitter. |

## Dépendances

En termes de dépendances et au-delà des projets présents dans le dossier *Libraries*, le kit de démarrage ODAF Openturf nécessite les éléments suivants :

* Le kit de développement Silverlight 4 disponible à l’adresse <http://www.silverlight.net/getstarted/> ;
* La boite à outils de Silverlight 4 disponible à l’adresse <http://silverlight.codeplex.com/> ;
* Le kit de développement Bing Cartes pour Silverlight disponible à l’adresse <http://www.microsoft.com/downloads/en/details.aspx?displaylang=en&FamilyID=beb29d27-6f0c-494f-b028-1e0e3187e830> ;
* Le mode étendu du contrôle Bing Cartes pour Silverlight téléchargeable à l’adresse <http://connect.microsoft.com/SilverlightMapModesBeta>. Le téléchargement nécessite au préalable une inscription sur le site avec votre Windows Live ID ;
* Les outils de développement pour Windows Azure disponibles à l’adresse <http://www.microsoft.com/windowsazure/sdk/> ;

Par ailleurs, l’application client mobile pour Android référence la bibliothèque JTwitter. Cette dernière qui constitue un « wrapper » de l’API Twitter est disponible sous licence libre LGPL à l’adresse <http://www.winterwell.com/software/jtwitter.php>. Les deux archives Java  nécessaires sont :

* jtwitter.rar : <http://www.winterwell.com/software/jtwitter/jtwitter.jar> ;
* signpost.rar (Simple OAuth message signing for Java) : <http://oauth-signpost.googlecode.com/files/signpost-core-1.2.1.1.jar>.

Enfin, l’application client mobile pour Windows Phone référence les bibliothèques suivantes disponibles sous licence libre :

* La bibliothèque TweetSharp pour Windows Phone qui constitue un « wrapper » de l’API Twitter est disponible sous licence libre MIT à l’adresse <http://www.softpedia.com/dyn-postdownload.php?p=166939&t=0&i=1>. Le code source afférent est disponible à l’adresse <https://github.com/danielcrenna/tweetsharp>.

Cette bibliothèque utilise les modèles T4 de Visual Studio et référence :

* 1. La bibilothèque Hammock pour Windows Phone, une API client HTTP écrite en C# pour consommer des services RESful, et dont le code source est disponible à l’adresse <https://github.com/danielcrenna/hammock> ;
  2. La bibliothèque JSon.NET pour Windows Phone disponible sur la forge CodePlex à l’adresse <http://json.codeplex.com> ;
  3. La bibliothèque SharpZipLib pour Windows Phone dont le code source et l’assemblage sont disponibles à l’adresse <http://www.icsharpcode.net/OpenSource/SharpZipLib>.
* La bibliothèque Facebook C# SDK pour Windows Phone (*dossier sl3-wp*) disponible à l’adresse <http://facebooksdk.codeplex.com/> sur la forge CodePlex.

# Compilation des éléments du Kit de démarrage ODAF Openturf

Les différentes composantes du Kit de démarrage ODAF Openturf (hormis les applications Android et iPhone) peuvent être compilées en utilisant Visual Studio 2010 ou la commande *msbuild* en ligne de commande. Les deux procédures sont décrites dans cette section.

## Compilation des éléments en ligne de commande

Pour compiler une solution particulière via la ligne de commande, veuillez suivre ces étapes :

1. Ouvrez l’invite de commande du SDK avec des privilèges administratifs et placez-vous dans le sous-répertoire *Source* du répertoire d’installation du Kit de démarrage ODAF Openturf.
2. Saisissez *msbuild ODAF.Website.sln* sur la ligne de commande. Les éléments relatifs à la solution considérée sont compilés dans le répertoire */bin/Debug* relatif au répertoire de chaque projet composant la solution.

## Compilation des éléments en utilisant Visual Studio 2010

Pour compiler une solution particulière via Visual Studio 2010, veuillez suivre ces étapes :

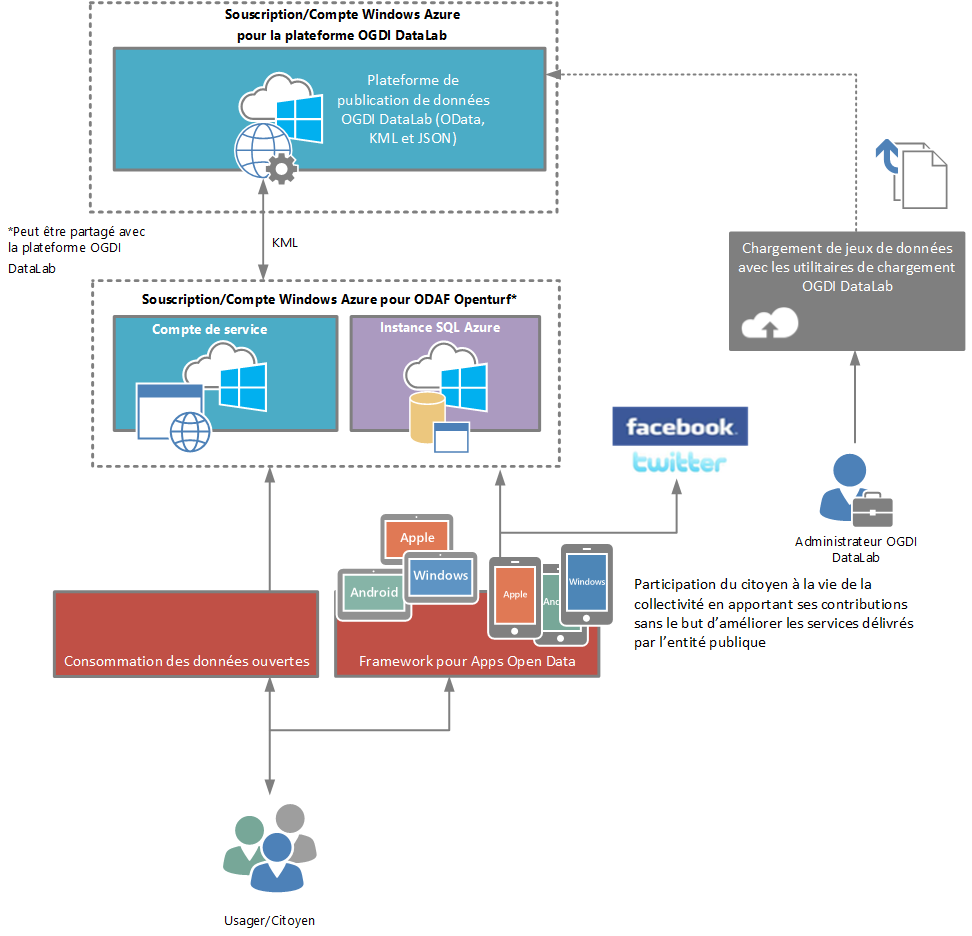
1. Ouvrez Windows Explorer et placez-vous dans le sous-répertoire *Source* du répertoire d’installation du Kit de démarrage ODAF Openturf.
2. Double-cliquez sur l’icône du fichier .*sln* situé dans l’un des dossiers du sous-répertoire *Source* pour ouvrir le fichier dans l’environnement Visual Studio.
3. Dans le menu **Build**, choisissez **Rebuild Solution**. Tous les éléments relatifs à la solution considérée sont compilés dans les répertoires */bin/Debug* relatifs aux répertoires de chaque projet composant la solution.

# Mise en œuvre des éléments du Kit de démarrage ODAF Openturf

Le Kit de démarrage ODAF Openturf se présente sous la forme de composants logiciels qui permettent à toute entité publique :

1. De mettre à disposition de façon attrayante des données publiques sur une carte facilement consultable par les citoyens
2. De permettre au citoyen de jouer pleinement son rôle en apportant sa contribution et ses idées sur les données publiées par la collectivité territoriale dans le but d’améliorer la qualité des services délivrés.

La figure suivante illustre les éléments du kit de démarrage ODAF Openturf tels que décrits précédemment dans un contexte de mise en œuvre complète et opérationnelle dans Windows Azure en consommant des flux de données KML exposés par une plateforme OGDI DataLab.



La plateforme OGDI DataLab est une solution sous licence libre également disponible sur la [forge GitHub](https://github.org/openlab/datalab)[[61]](#footnote-61). Cette plateforme permet via ses composants d’exposer des données publiques sous différents formats standardisés (protocoles OData, KML, AtomPub, JSON et JSONP). Il convient de noter qu’une instance de démonstration est publiée à l’adresse <http://ogdifrance.cloudapp.net>.

Le kit de démarrage ODAF Openturf constitue un client naturel du kit de démarrage OGDI DataLab et s’intègre donc particulièrement bien avec ce dernier. Comme le suggère la figure précédente, votre solution ODAF Openturf peut être déployée dans Windows Azure sur le même compte que votre solution OGDI DataLab et il suffit de faire pointer celle-ci vers les flux de données KML exposés par la solution OGDI DataLab pour disposer rapidement d’une solution complètement fonctionnelle.

Le document Déployer le kit de démarrage ODAF Openturf dans Windows Azure qui accompagne le kit de démarrage ODAF Openturf décrit les modalités de mise en œuvre des composants ODAF Openturf dans Windows Azure.

De façon similaire, les documents Déployer le kit de démarrage ODAF Openturf pour Android, Déployer le kit de démarrage ODAF Openturf pour iPhone et Déployer le kit de démarrage ODAF Openturf pour Windows Phone précisent les prérequis et les modalités de mise en œuvre des applications mobiles à destination des smartphones Android, iPhone et Windows Phone.

# Pour aller plus loin

Pour de plus amples information sur les solutions proposées par Microsoft à destination du Secteur Public, nous vous invitons à consulter le [site portail Web Microsoft Secteur Public](http://www.microsoft.com/france/secteur-public/default.aspx)[[62]](#footnote-62).



Pour de plus amples informations sur l'interopérabilité technique des produits et technologies Microsoft avec des logiciels et matériels d'autres fournisseurs, nous vous invitons à consulter le [site portail Web Microsoft Open](http://www.microsoft.com/france/openness/default.aspx)[[63]](#footnote-63).

Dans le contexte de la révolution numérique en marche, du foisonnement des technologies, des innovations permanentes, etc. l’interopérabilité, c’est-à-dire la capacité des produits, logiciels et matériels à communiquer entre eux, représente un enjeu fondamental aujourd’hui: à terme, tous les produits, logiciels, matériel doivent pouvoir communiquer entre eux.

Derrière la dimension technique de l’interopérabilité, le véritable enjeu est la liberté de chacun : celle de choisir les outils qui correspondent le mieux à ses besoins, celle de pouvoir accéder à ses données au fil du temps, indépendamment de la technologie utilisée.

Avec l’[annonce majeure](http://www.microsoft.com/en-us/news/press/2008/feb08/02-21ExpandInteroperabilityPR.aspx)[[64]](#footnote-64) faite au mois de février 2008, Microsoft s’est engagé, pour garantir la pérennité des données et assurer la conformité aux standards de l’industrie informatique, la prise en charge des principales normes internationales, dans une politique d’ouverture sans précédent dans l’histoire informatique, une démarche prospective qui prouve qu’innovation, propriété intellectuelle et interopérabilité ne s’opposent pas.

Cela se traduit notamment par la mise à la disposition de la communauté des développeurs, de plus de 50 000 pages d’informations techniques relatives aux protocoles de communication et aux interfaces de nos produits-phare, et par le développement des relations collaboratives avec les différents acteurs du secteur, y compris les communautés investies dans les logiciels Open source.

Pour de plus amples informations sur les standards d'interopérabilité et normes internationales que Microsoft soutient ou auxquels Microsoft apporte sa collaboration, nous vous invitons également à consulter le portail Microsoft Open.

Par ailleurs, afin de promouvoir l’industrie du logiciel sans favoriser un modèle de développement plutôt qu’un autre, nous contribuons depuis 2009 à l’évolution du noyau Linux sous licence GPL. Nous finançons également, et mettons gratuitement à disposition, en mode open source, des logiciels permettant d’échanger des documents issus de différentes suites bureautiques (Microsoft Office, Open Office, etc.). Nous investissons également dans le projet Apache Hadoop et contribuons à améliorer dans ce cadre les environnements de développement et d’exécution d’Hadoop sur Windows Azure et Windows Server.

Pour de plus amples informations sur les contributions, le financement et la mise à disposition sous licence libre de logiciels de Microsoft, nous vous invitons également à consulter le portail Microsoft Open.

1. Solution Open Data clé en main: https://mspartner.microsoft.com/fr/fr/Pages/Solutions/Open-Data-Cle-en-main.aspx [↑](#footnote-ref-1)
2. Blog MSDN Open Data France : http://aka.ms/OpenDataFrance/ [↑](#footnote-ref-2)
3. Site portail Web Microsoft Secteur Public : http://www.microsoft.com/france/entreprises/secteur-public/solutions-pour-secteur-public.aspx [↑](#footnote-ref-3)
4. Microsoft Windows Azure : http://www.microsoft.com/france/windows-azure/ [↑](#footnote-ref-4)
5. Technologie Silverlight : http://www.silverlight.net/ [↑](#footnote-ref-5)
6. Technologie ASP.NET MVC : http://www.asp.net/mvc [↑](#footnote-ref-6)
7. Windows Phone Mango : http://www.microsoft.com/windowsphone/en-us/cmpn/mango-overview.aspx [↑](#footnote-ref-7)
8. Microsoft Windows Azure Training Kit : http://www.microsoft.com/downloads/en/details.aspx?FamilyID=413E88F8-5966-4A83-B309-53B7B77EDF78&displaylang=en [↑](#footnote-ref-8)
9. Microsoft Silverlight 4 Training Course : http://msdn.microsoft.com/en-us/silverlight4trainingcourse.aspx [↑](#footnote-ref-9)
10. Windows Phone Mango Training Kit : http://msdn.microsoft.com/en-us/wp7trainingcourse.aspx [↑](#footnote-ref-10)
11. Windows Azure Team Blog : http://blogs.msdn.com/b/windowsazure/ [↑](#footnote-ref-11)
12. Developer's Guide to Social Programming: Building Social Context Using Facebook, Google Friend Connect, and the Twitter API : http://www.amazon.fr/Developers-Guide-Social-Programming-Building/dp/0321680774/ref=sr\_1\_1?ie=UTF8&qid=1313055760&sr=8-1 [↑](#footnote-ref-12)
13. Professional Twitter Development: With Examples in .NET 3.5 : http://www.wrox.com/WileyCDA/WroxTitle/Professional-Twitter-Development-With-Examples-in-NET-3-5.productCd-0470531320.html [↑](#footnote-ref-13)
14. Edition RSLN du premier trimestre 2011 Open Data et nous, et nous et nous ? : http://www.rslnmag.fr/Content/Files/ff/UERGL1YyUkN6ZHdFaTZNSFJOQS5wZGY= [↑](#footnote-ref-14)
15. Dossier Quel sens donner au monde des données ouvertes ? : http://www.rslnmag.fr/blog/dossier/l-open-data\_et-nous\_et-nous\_et-nous\_/2011/4/19/quel-sens-donner-au-monde-des-donnees-ouvertes\_/ [↑](#footnote-ref-15)
16. Conférence [L’Open Data, et nous, et nous et nous ?](http://www.rslnmag.fr/blog/theme/les-rencontres-rsln/2011/2/10/l-open-data_et-nous_et-nous_et-nous_venez-imaginer-le-monde-de-l-abondance-des-donnees_/) : http://www.rslnmag.fr/blog/theme/les-rencontres-rsln/2011/2/10/l-open-data\_et-nous\_et-nous\_et-nous\_venez-imaginer-le-monde-de-l-abondance-des-donnees\_/ [↑](#footnote-ref-16)
17. World e.gov Forum : http://wegf.org/fr/ [↑](#footnote-ref-17)
18. ODAF : http://odaf.codeplex.com [↑](#footnote-ref-18)
19. VanGuide for Windows Phone 7 : http://vanguidewp7.codeplex.com/ [↑](#footnote-ref-19)
20. RFC 4627 The application/json Media Type for JavaScript Object Notation (JSON) : http://tools.ietf.org/html/rfc4627 [↑](#footnote-ref-20)
21. Définition de REST sur Wikipedia : http://en.wikipedia.org/wiki/Representational\_State\_Transfer [↑](#footnote-ref-21)
22. Billet Interoperability Elements of a Cloud Platform Outlined at OSCON : http://blogs.msdn.com/b/interoperability/archive/2010/07/22/interoperability-elements-of-a-cloud-platform-outlined-at-oscon.aspx [↑](#footnote-ref-22)
23. Microsoft Global Foundation Services (GFS) : http://www.globalfoundationservices.com/ [↑](#footnote-ref-23)
24. Certifications ISO 27001 Microsoft, Global Foundation Services Division : http://www.bsiamerica.com/en-us/Assessment-and-Certification-services/Management-systems/Certificate-and-Client-Directory-search/Search/Search-Results/?pg=1&licencenumber=IS+533913&searchkey=companyXeqXmicrosoft [↑](#footnote-ref-24)
25. Certification ISO 27001 Windows Azure : http://www.bsiamerica.com/en-us/Assessment-and-Certification-services/Management-systems/Certificate-and-Client-Directory-search/Search/Search-Results/?pg=1&licencenumber=IS+577753&searchkey=licenceXeqX577753 [↑](#footnote-ref-25)
26. Billet Response to Question about SAS 70 Objectives : http://blogs.technet.com/b/gfs/archive/2009/06/16/response-to-question-about-sas-70-objectives.aspx [↑](#footnote-ref-26)
27. U.S.-EU Safe Harbor : http://www.export.gov/safeharbor/eu/eg\_main\_018365.asp [↑](#footnote-ref-27)
28. Centre de gestion de la confidentialité Windows Azure : http://www.windowsazure.com/fr-fr/support/trust-center/ [↑](#footnote-ref-28)
29. Réponse standard pour les demandes d'information, Sécurité et Respect de la vie privée : http://www.microsoft.com/en-us/download/details.aspx?id=26647 [↑](#footnote-ref-29)
30. Windows Azure Marketplace : https://datamarket.azure.com/ [↑](#footnote-ref-30)
31. Correctif logiciel KB971842 Correction des métadonnées WCF lors de l'hébergement derrière un programme d'équilibrage de charge : http://support.microsoft.com/kb/971842 [↑](#footnote-ref-31)
32. Correctif logiciel cumulatif KB977420 pour Windows Communication Foundation : http://support.microsoft.com/kb/977420 [↑](#footnote-ref-32)
33. Correctif logiciel KB983301 Activation de Windows Azure IntelliTrace sur les systèmes d'exploitation 32 bits : http://archive.msdn.microsoft.com/KB983301 [↑](#footnote-ref-33)
34. Module de réécriture d'URL d'IIS 7.0 : http://www.iis.net/download/URLRewrite [↑](#footnote-ref-34)
35. Windows PowerShell 2.0 : http://go.microsoft.com/fwlink/?LinkId=192992 [↑](#footnote-ref-35)
36. Kit de développement Windows Azure 1.6 : http://www.microsoft.com/windowsazure/sdk/ [↑](#footnote-ref-36)
37. Kit de développement Silverlight 4 : http://www.silverlight.net/getting-started [↑](#footnote-ref-37)
38. OGDI DataLab sur GitHub ! Impliquez-vous dans la communauté : http://blogs.msdn.com/b/ogdifrance/archive/2012/05/14/ogdi-datalab-sur-github-impliquez-vous-dans-la-communaut-233.aspx [↑](#footnote-ref-38)
39. GitHub for Windows: http://blogs.msdn.com/b/ogdifrance/archive/2012/06/14/github-for-windows.aspx [↑](#footnote-ref-39)
40. Blog MSDN Open Data France: http://aka.ms/OpenDataFrance/ [↑](#footnote-ref-40)
41. Visual Studio 2012 Update 2 : http://www.microsoft.com/en-us/download/details.aspx?id=38188 [↑](#footnote-ref-41)
42. Visual Studio Tools for Git : http://visualstudiogallery.msdn.microsoft.com/abafc7d6-dcaa-40f4-8a5e-d6724bdb980c [↑](#footnote-ref-42)
43. Integrating and Using Github in Visual Studio 2012 : http://www.codeproject.com/Articles/581907/IntegratingplusandplusUsingplusGithubplusinplusVis [↑](#footnote-ref-43)
44. Compte Twitter : https://twitter.com/signup [↑](#footnote-ref-44)
45. Projet SubSonic : http://subsonicproject.com/ [↑](#footnote-ref-45)
46. T4 Text Template Transformation Toolkit : http://www.olegsych.com/2007/12/text-template-transformation-toolkit/ [↑](#footnote-ref-46)
47. LINQ : http://msdn.microsoft.com/en-us/netframework/aa904594 [↑](#footnote-ref-47)
48. Patron de conception « Active Record » : http://en.wikipedia.org/wiki/Active\_record\_pattern [↑](#footnote-ref-48)
49. Projet bitly-dot-net : http://code.google.com/p/bitly-dot-net/ [↑](#footnote-ref-49)
50. Licence libre MIT : http://www.opensource.org/licenses/mit-license.php [↑](#footnote-ref-50)
51. Blog Mike Gleason JR : http://blog.mikecouturier.com/search/label/bitly-dot-net [↑](#footnote-ref-51)
52. Clé d’API bit.ly: http://bit.ly/account/ [↑](#footnote-ref-52)
53. Professional Twitter Development: With Examples in .NET 3.5 : http://www.wrox.com/WileyCDA/WroxTitle/Professional-Twitter-Development-With-Examples-in-NET-3-5.productCd-0470531320.html [↑](#footnote-ref-53)
54. Spécification OAuth : http://oauth.net [↑](#footnote-ref-54)
55. Article Twitter Development: The OAuth Specification : http://p2p.wrox.com/content/articles/twitter-development-oauth-specification [↑](#footnote-ref-55)
56. Article Twitter Development: The OAuth Workflow : http://p2p.wrox.com/content/articles/twitter-development-oauth-workflow [↑](#footnote-ref-56)
57. Article Twitter Development: Using OAuth to Authenticate Against the Twitter API: Walkthroughs : http://p2p.wrox.com/content/articles/twitter-development-using-oauth-authenticate-against-twitter-api-walkthroughs [↑](#footnote-ref-57)
58. Kit de développement Bing Cartes pour Silverlight : http://www.microsoft.com/downloads/en/details.aspx?displaylang=en&FamilyID=beb29d27-6f0c-494f-b028-1e0e3187e830 [↑](#footnote-ref-58)
59. Mode étendu du contrôle Bing Cartes pour Silverlight : http://connect.microsoft.com/SilverlightMapModesBeta [↑](#footnote-ref-59)
60. Boite à outils de Silverlight 4 : http://silverlight.codeplex.com/ [↑](#footnote-ref-60)
61. Kit de démarrage OGDI DataLab : https://github.org/openlab/datalab [↑](#footnote-ref-61)
62. Site portail Web Microsoft Secteur Public : http://www.microsoft.com/france/secteur-public/default.aspx [↑](#footnote-ref-62)
63. Site portail Web Microsoft Interopérabilité : http://www.microsoft.com/france/openness/default.aspx [↑](#footnote-ref-63)
64. Communiqué de presse Microsoft Makes Strategic Changes in Technology and Business Practices to Expand Interoperability : http://www.microsoft.com/en-us/news/press/2008/feb08/02-21ExpandInteroperabilityPR.aspx [↑](#footnote-ref-64)