Copy No. _____

Defence Research and Development Canada
Recherche et développement pour la défense Canada

DEFENCE R&D DÉFENSE

Addition of an Open-Web Data Source Service to the Consistency Application

Marie-Odette St-Hilaire
OODA Technologies Inc.

Prepared by:
OODA Technologies Inc.
4891 Av. Grosvenor
Montreal, QC H3W 2M2

Project Manager: Anthony W. Isenor, 902-426-3100 ext. 106
Contract Number: W7707-115277
Contract Scientific Authority: Anthony W. Isenor, 902-426-3100 ext. 106

Defence R&D Canada – Atlantic

Contract Report
DRDC Atlantic CR 2011-038
April 2011

Canada

This page intentionally left blank.

Addition of an Open-Web Data Source Service to the Consistency Application

Marie-Odette St-Hilaire

Prepared by:

OODA Technologies Inc.
4891 Av. Grosvenor, Montreal Qc, H3W 2M2

Project Manager: Anthony W. Isenor 902-426-3100 Ext. 106
Contract Number: W7707-115277
Contract Scientific Authority: Anthony W. Isenor 902-426-3100 Ext. 106

Defence R&D Canada – Atlantic
Contract Report
DRDC Atlantic CR 2011-038
April 2011

Principal Author

Original signed by Marie-Odette St-Hilaire

Marie-Odette St-Hilaire

Approved by

Original signed by Francine Desharnais

Francine Desharnais
Head/Maritime Information and Combat Systems Section

Approved for release by

Original signed by Ron Kuwahara for Calvin Hyatt

Calvin Hyatt
Head/Document Review Panel

Abstract

The Consistency Application was built upon web services in a Service Oriented Architecture (SOA) paradigm. The Consistency Application (CA) compares the information from diverse data sources, all providing information relevant to Maritime Domain Awareness (MDA). The CA allows an assessment of the information consistency across the data sources (i.e., ASIA database, SeaSpider database, International Telecommunication Union (ITU) database) and visualization of the consistency results within a Google Earth environment.

In this work, an additional data source was added to the CA, namely the ShipSpotting data source available at `www.shipspotting.com`.

Résumé

L'application de vérification de la cohérence a été développée et implantée dans une architecture orientée services. Cette application compare l'information provenant de diverses sources de données, chacune rapportant des renseignements pertinents au domaine de la connaissance de la situation maritime (CSM). L'application permet d'évaluer la cohérence des renseignements disponibles sur les navires provenant de diverses sources de données (i.e. les bases de données ASIA, SeaSpider et ITU) et visualiser les résultats de cohérence dans un environnement Google Earth.

Ce projet consistait à ajouter une source de données additionnelle à l'application de vérification de la cohérence. Cette source, appelée source de données ShipSpotting, est disponible en ligne au `www.shipspotting.com`.

This page intentionally left blank.

Executive summary

Addition of an Open-Web Data Source Service to the Consistency Application

Marie-Odette St-Hilaire; DRDC Atlantic CR 2011-038; Defence R&D Canada – Atlantic; April 2011.

Background: The collection of data and information to support Maritime Domain Awareness (MDA) can take on many forms. In the Maritime Information and Knowledge Management group at DRDC Atlantic, effort is being placed on the utilization of open-source information pertinent to MDA. As an example, the SeaSpider application supports the accumulation of web-based information on ship routes, the Automated Ship Image Acquisition System (ASIA) acquires photographs of ships as they transit the narrows of Halifax Harbour, while the International Telecommunication Union (ITU) provides specific ship attribute information online.

A cross-comparison of the information available from these sources enables an assessment of the information consistency. This contract effort extended existing software that performed such a comparison, to include additional data provided at the ShipSpotting website. This contract effort supported the Applied Research Project 11HL *Technologies for Trusted Maritime Situational Awareness*.

Results: A source providing online information was added to existing software, called the Consistency Application. The source contains the information available at the ShipSpotting website. The ShipSpotting data is then used in comparison with similar information available from ASIA database, SeaSpider database and ITU database. Moreover, this additional data source provides ship photographs that are visualized, along with the statistically based consistency results, within a Google Earth environment.

Significance: The Canadian Forces often utilizes online information sources for augmenting or verifying existing ship-related information. Research into the automation of this task could provide time savings to those involved in the searches. As well, the added benefit of cross-comparing the source information and presenting the results in a stop-light fashion, means users can quickly assess the consistency of information from the various sources.

Future Plans: The intent is to enhance the Consistency Application by including additional data sources and upgrading existing sources. Additional data sources may include access to near-real-time ship geo-location information. Upgraded sources may include alteration of the SeaSpider application to focus on pleasure craft within the Canadian Arctic Archipelago.

Sommaire

Addition of an Open-Web Data Source Service to the Consistency Application

Marie-Odette St-Hilaire ; DRDC Atlantic CR 2011-038 ; R & D pour la défense Canada – Atlantique ; avril 2011.

Contexte : La collecte de données et de renseignements pour appuyer la Connaissance de la Situation Maritime (CSM) peut prendre diverses formes. Le groupe de Gestion de l'Information et du Savoir Maritime (GISM) à DRDC Atlantique dirige des efforts pour utiliser de l'information de sources libres. Par exemple, les trois applications suivantes sont des initiatives de GISM. L'application SeaSpider appuie la CSM en accumulant des renseignements Web sur les itinéraires des navires et en enrichissant une base de données. Le système d'acquisition automatisée d'images de navires (ASIA) prend et sauvegarde des photos de navires prises lors de leur transit dans le port de Halifax. Finalement, l'application en ligne de l'Union Internationale des Télécommunications (ITU) procure des renseignements détaillés sur les navires.

La comparaison des renseignements rendus disponible par ces sources de données permet d'évaluer la cohérence de l'information. L'objectif de ce projet était de créer une extension au logiciel existant responsable de cette comparaison, en incluant les données rapportées par le site Web ShipSpotting. Cette extension a été développée en appui au projet de recherche appliquée 11HL : *Technologies assurant la fiabilité de la connaissance de la situation maritime*.

Résultats : Une source de données a été ajoutée au logiciel appelé application de vérification de la cohérence. La source rapporte des renseignements disponibles sur le site Web de ShipSpotting. Les données de ShipSpotting sont ensuite utilisées en comparaison avec des renseignements similaires disponibles dans les bases de données de ASIA, SeaSpider et ITU. De plus, cette source de données supplémentaire fournit des images de navires qui sont affichées, avec les statistiques de cohérence, dans un environnement Google Earth.

Importance : Les Force Canadiennes utilisent souvent des renseignements disponibles en ligne pour compléter ou vérifier de l'information existante relative aux navires. La recherche dans le domaine de l'automatisation de ces tâches pourrait permettre de sauver du temps à ceux qui cherchent ces renseignements en ligne. De plus, la présentation des résultats de cohérence, à l'aide d'une visualisation de type feux de circulation, permet aux usagers d'évaluer rapidement la capacité d'une source de données à fournir des renseignements pertinents.

Perspectives : Nous prévoyons ajouter d'autres sources de données et améliorer des sources déjà intégrées à l'application de vérification de la cohérence. Il est entre autres envisagé d'ajouter une source de données permettant la géo-localisation de navires en temps quasi réel. Une autre possibilité est d'actualiser l'application SeaSpider pour mettre l'accent sur les bateaux de plaisance dans l'archipel arctique canadien.

This page intentionally left blank.

Table of contents

List of figures

This page intentionally left blank.

1 Scope

In 2009 a Service Oriented Architecture (SOA), termed the Compare-MDA framework, was built by OODA Technologies over a period of seven months, in support of the Applied Research Project 11HL *Technologies for Trusted Maritime Situational Awareness*. This framework was developed to compare diverse data sources all containing similar MDA relevant information.

The Compare-MDA framework is based on an application and supporting data sources. The application, known as the Consistency Application (CA), links to the many supporting data sources. These data sources contain similar attributes of ship-related information. For example, the sources may all contain the call signs for the ships being described by the data source. The CA provides a means for the cross comparison of similar attribute data from the multiple data sources. After comparing across the multiple data sources, the CA displays a simple statical score of consistency for the specific ship. The CA also provides the ability to geo-locate the ship in a Google Earth display using a common web browser, by utilizing geospatial data from one of the designated data sources.

Thus, the objective of the CA is the utilization of multiple data sources, for the cross comparison of data from these sources. In the case of the work described here, a new data source is added to the framework, thus increasing the data available for cross comparison. The added source is from the open-web, thus diversifying the data provenance and hopefully leading to a better understanding of the information consistency. The ShipSpotting website is the new data source.

The ShipSpotting website provides enough ship-related information to be used as a data source for the consistency application and its *Terms of Use* allow its usage in this context. In addition, the ShipSpotting data source provides ship photographs. These photographs can be used for a visual perspective in the framework. Figure 1 illustrates the Compare-MDA framework augmented with the ShipSpotting data source.

This document summarizes all the technical activities and achievements of this project. It includes:

- A description of the ShipSpotting data source (Section 2).
- A description of the Consistency Application addition/modification required by the addition of the source (Section 3).
- A description of the modifications performed on the Web Client (used to visualize consistency results and ship information) to enable selection between the ship photo providers: ASIA and ShipSpotting (Section 4).
- A description of the impacts on the installation procedure (Section 5).

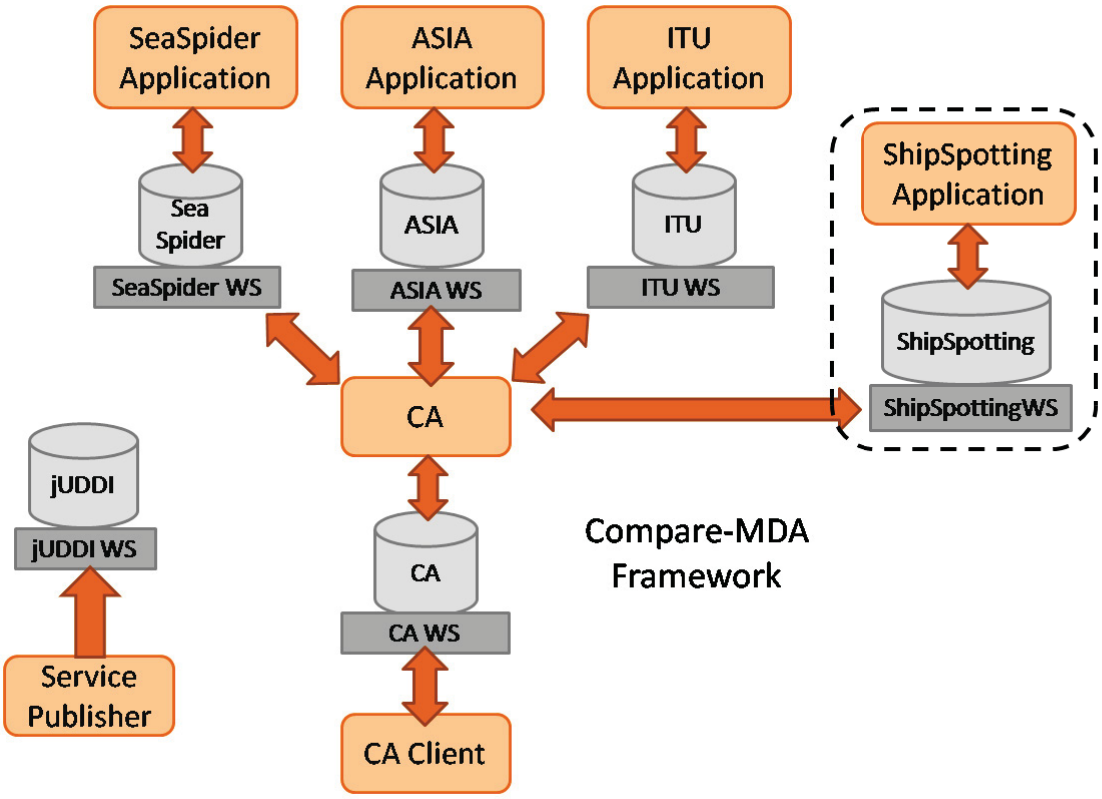

Figure 1: *The Compare-MDA framework initially included all components excluding the ones encompassed by the dashed line. The objective of this project was to augment the Compare-MDA framework with an open-web data source for cross comparison. This additional component is illustrated inside the dashed line area.*

2 ShipSpotting Data Source

The ShipSpotting data source is based on the ShipSpotting website search capability[1]. An application was developed to fill a local database with ship information from the ShipSpotting website. This application and the local database are wrapped as a web service and exposed as a data source to the CA. The architectural decision to use a local database containing the ShipSpotting data source information originates from design decision made for the initial Compare-MDA framework ([1]).

The ShipSpotting data source provides information about ships upon request. The diagram in Figure 2 illustrates its mode of operation. Upon request, the application looks first in the local Database (DB) to obtain the information on the specific ship being queried. If the ship corresponding to the query criteria is absent from the local DB, the application submits a query to the ShipSpotting website in an attempt to retrieve the ship information. This information is then used to update the ShipSpotting local DB. If for some reason, connection to the ShipSpotting website is no longer possible, the ShipSpotting data source can still operate. The impact is that the local database will not be updated.

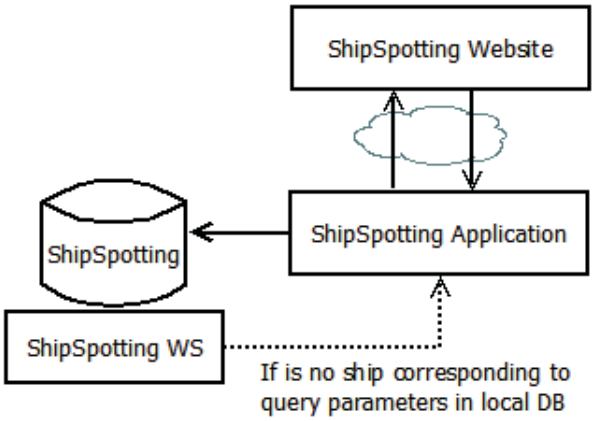

Figure 2: *ShipSpotting data source*

2.1 ShipSpotting Website

ShipSpotting.com is an on line ship photo repository, allowing members to upload ship photos from around the world. At the moment, there are 1,019,815 photos on line.

When a member uploads a photograph, they are required to provide information about the ship that is present in the photo. This information includes: ship name, International Maritime Organization (IMO) number, call sign, Maritime Mobile Service Identity (MMSI) number, ship's category (corresponding roughly to Automatic Identification System (AIS)

[1] `http://www.shipspotting.com/gallery/photo_search.php`

types), beam, draught, length, photo in full, medium or thumbnail version, photographer's name and profile, photo category, and the time and location the photo was taken. Only the ship's name, photo, photographer and photo category are mandatory items. Sometimes the remaining information is missing or incomplete. Each photo is categorized according to ShipSpotting category standards[2]. These categories sometimes differ from the ship's category, but are usually similar.

The website has a search facility that allows the querying of the photo repository. It is recommended to query on: ship's name, IMO and photographer name. It is possible to query on MMSI and call sign, but this has not been found to be reliable. The query result is a list of ship photos that contain photo and ship descriptions that match the query parameters. There are usually more than one photo representing a single ship. Note that the first photo in the result list is the most recent photo.

2.2 ShipSpotting Application

The ShipSpotting application is an interface to the ShipSpotting website's search page. It performs the following actions:

1. Build the Uniform Resource Locator (URL) from the query parameters.
2. Create a connection to the ShipSpotting search web page.
3. Process the response.
4. Return the output.
5. Update the local database with ship descriptions.

2.2.1 Interaction with ShipSpotting Website

The ShipSpotting search web page sends queries to its server using the HTTP GET method. Server-side PHP code dynamically produces the query result pages.

It is possible to remotely access the query result pages using Java NET libraries and classes URL, URLEncoder and URLConnection.

With those classes, actions 1 and 2 above are covered by the following steps:

Step 1 Create the URL object needed to create the connection. The simplest constructor would be that of a String representing that URL.

[2]Description available at `http://www.shipspotting.com/support/`

Step 2 Use the URL object to get a URLConnection to the remote resource and to set the content type.

Step 3 Get a DataOutputStream and start sending to that resource.

Step 4 Get a BufferedReader and start getting a response.

The first step consists of building the URL from the query parameters, which are the ship's name and/or the IMO number. This URL has the following structure:

`http://www.shipspotting.com/gallery/photo_search.php?query=shipName&category=-99&page=1` for a query where the name is `shipName`

and

`http://www.shipspotting.com/gallery/photo_search.php?imo=shipImo&page=1` for a query where the IMO is `shipImo`.

For some queries, the results are displayed on more than one page. In these cases, pages are reached by incrementing the `page` attribute value in the URL.

The resource response is a HyperText Markup Language (HTML) string containing the data needed to produce the output (e.g., flag, ship name, ...). In processing the response (Step 3), the HTML string has to be parsed, i.e. the HTML tag hierarchy embedded in the document is used to extract target information in the document.

In the case of ShipSpotting, the structure of the HTML containing the information was difficult to parse. The relevant information is displayed on several pages and its structure varies upon the information availability. The extraction process was therefore slower than for ITU (see [1] Section 2.3.1 for details about ITU application).

Note that this approach is very sensitive to the response structure. A change in the results layout may jeopardize the information extraction procedure. It was proposed in [1] to use the content analysis tool Alchemy API[3]to fix the sensitive parsing issue. It was suggested to use their Text Extraction / Web Page Cleaning feature to automatically "clean" web pages (removing navigation links, advertisements, and other undesirable content) and thus remove the dependency of the parsing process on the HTML tag structure. Unfortunately, the Alchemy API free version is no longer available, so that option was rejected.

However, the impact of a change in the application design on the ShipSpotting Web Service (WS) is limited. If a query is sent to the ShipSpotting WS and the ship description satisfying this query is not in the local database, the query is sent to the ShipSpotting web page (see Figure 2). If the response can't be parsed correctly, the local database won't be

[3] `http://www.alchemyapi.com/`

updated with new ship descriptions and no result satisfying the query will be sent back by the ShipSpotting WS. Essentially, changes to the website could result in no response information being supplied to the CA, but will not also not result in termination of the CA.

2.2.2 Database

The local ShipSpotting database runs on a MySQL server and has a single table: `shipspottingtable`. This table contains all the information made available by the ShipSpotting website:

imo IMO number (numeric code).

name Ship name.

flag Administration and/or Geographical Area, indicated by the full name.

callsign Call sign (alpha-numeric code).

mmsi MMSI number (numeric code).

shipcategory Ship's category, indicated by the full name.

beam Beam of the ship.

draught Draft (or draught) of the ship.

length Length of the ship.

photo URL location of the thumbnail version of the most recent photo of the ship.

photographer Name of the photographer (for copyright).

photocategory The photo's category, indicated by the full name.

time Date when the photo was added to the site (MM DD, YYYY).

location City and country where the photo was taken.

The photos are not downloaded and stored in a local repository. Only the URL location of the thumbnail version is stored. Since there is more than a million photos in the ShipSpotting repository, keeping and maintaining a local copy of them would require great efforts. With the thumbnail's URL, it is easy to access the photo and it avoids the burden of maintaining a local copy. The downside of that approach is the sensibility to the ShipSpotting photo repository structure. If photos are re-localized, the URL may not point to the photo anymore.

The ShipSpotting application (described in 2.2) was used to fill the table. The ship names stored in the ITU database were used as query for the ShipSpotting website (127,000 distinct ship names in the ITU DB). As mentioned in section 2.1, ShipSpotting.com is a photo repository, meaning that there may be many photos representing the same ship. When there are more than one photo for a given ship name, only the description of the most recent one was stored.

There are currently 119,460 entries in `shipspottingtable`. More than one entry may correspond to a single ship. One of the reasons for multiple entries is that some contributors to the ShipSpotting website make typographical errors in the ship name or misidentify the ship in some way. In other cases, the name of the photo does not represent the ship's name (as supposed): some photos capture more than one ship while some others represent harbors, on-sea museums, submarines, flight decks, ship crests, armaments or harbours. The local database reflects these variations. It is estimated that about 2.8% of the total entries correspond to photos that do not represent a ship.

2.3 ShipSpotting Web Service

The ShipSpotting Web Service, called `ShipSpottingDataSourceService`, is a web service interface to the ShipSpotting DB and the ShipSpotting website. Unlike ASIA and SeaSpider web services, the ShipSpotting web service may update its DB. It was developed following the bottom-up methodology described in section 8.4 of [2] (the same as for the other 3 data source web services).

The ShipSpotting WS offers an operation that returns data pertaining to specific ships depending on the input query parameters. In the ShipSpotting DB, a ship is uniquely defined by the combination of its IMO, MMSI and name. In this case, the returned data from the service contains at least one of these items plus other complex data types containing only static information specific to the ship. No dynamic information is available.

The ShipSpotting WS also offers an operation to acquire the thumbnail versions of the ship images. The service transfers each image into the returning service message using the Message Transmission Optimization Mechanism (MTOM) protocol, a W3C Recommendation designed for optimizing the electronic transmission of attachments (see [3] for details about MTOM and [4] for its use with Axis2).

2.3.1 Operations

The operations offered by the ShipSpotting web service are described in Table 1. The operation `getInformation` gets all ships from the ShipSpotting DB that matches the query parameters. As mentioned in 2.2.2, the database has entries that do not correspond to ships (for instance submarines, flight decks or harbours). The operation `getInformation` filters the ShipSpotting DB entries so that only ships are returned.

Name	Input	Output
`getInformation`	QueryParameter	Array of Ship
`getPhoto`	QueryParameter	Array of byte

Table 1: *ShipSpotting Web Service Operations*

`getPhoto` gets the thumbnail image of the ship matching the query parameters. The word *Copyright* with the name of the photographer are superimposed on the bottom right part of the thumbnail. That way, the photo returned by `getPhoto` conforms to the ShipSpotting copyright policy[4]. Displaying the photographers name and a link to the ShipSpotting website are requirements of the ShipSpotting policy.

2.3.2 Data Model

The structure of QueryParameter and Ship objects are illustrated in Figure 3 and 4 respectively.

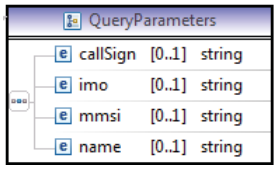

Figure 3: *QueryParameter data structure for ShipSpotting web service. The QueryParameter object is the input of operations* `getInformation` *and* `getPhoto`*.*

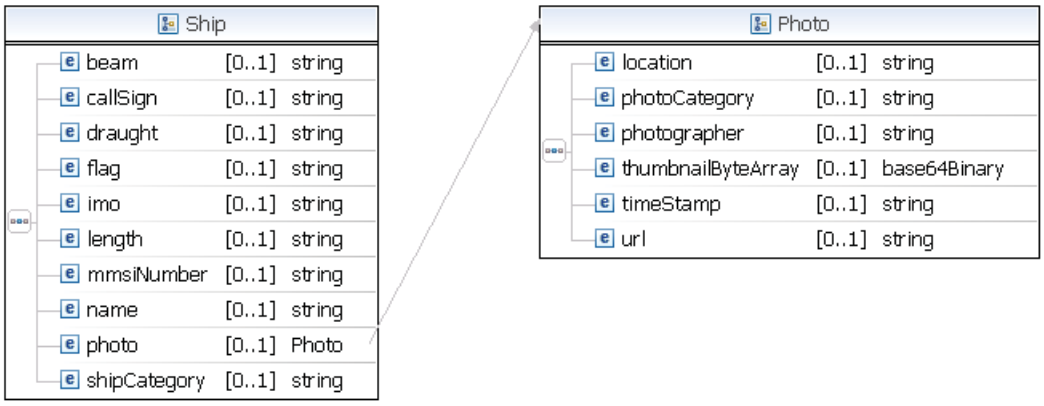

Figure 4: *Ship data structure for ShipSpotting web service. The output of operation* `getInformation` *is a list of Ship objects.*

[4] `http://www.shipspotting.com/about/terms.php`

3 Addition to Consistency Application

The CA was designed to easily add new data sources. The addition of the ShipSpotting data source represents the first addition of a data source since the initial development of the CA and therefore the first test of the procedure.

A few software components were impacted by the addition of the ShipSpotting data source. In particular, those components dealing with the cross comparison score computation performed by the CA. The augmented Consistency Application and its context are illustrated in Figure 5.

The addition of a new data source impacts only the Communication and Control layers. More precisely, it implies the addition of a client for ShipSpotting data source, addition of alignment methods for the mapping between ShipSpotting and CA vocabularies, and modification in the CA Parameters component. No modification to the comparison algorithms (Logic layer), the CA database nor the CA client are required.

Technical details on the addition of a data source and its client are presented in section 8.6 of [2]. This methodology was closely followed for the addition of ShipSpotting.

The ShipSpotting client and vocabulary alignment are described below. For details about the simple procedure for parameters modification, see section 8.6.3 in [2].

3.1 ShipSpotting Client

The ShipSpotting client is a subcomponent in the communication layer. This client provides the interface between the ShipSpotting data source and the CA. Specifically, the ShipSpotting client is a Java class named `ShipspottingClient`. See Section 3.1.0.1 of [1] for details about data source clients.

In order to interface with the CA, each data source client must implement the `DataSourceClient` interface. This interface contains three methods: `invoke`, `getNewShips` and `getLastEntry`. The method `invoke` is used to get ship descriptions from a data source corresponding to the criteria of the input query. The method `getNewShips` retrieves the list of the ship names having activities/reports reported by the source since the input date. The `getLastEntry` method gets the date when the last activity/report was recorded by the source. In the case of ShipSpotting, since the service only provides static information (contrary to ASIA and SeaSpider WS), `getNewShips` returns an empty list and `getLastEntry` always returns 0.

Details about the role of each of these methods in the CA data flow can be found in section 7.4 of [1].

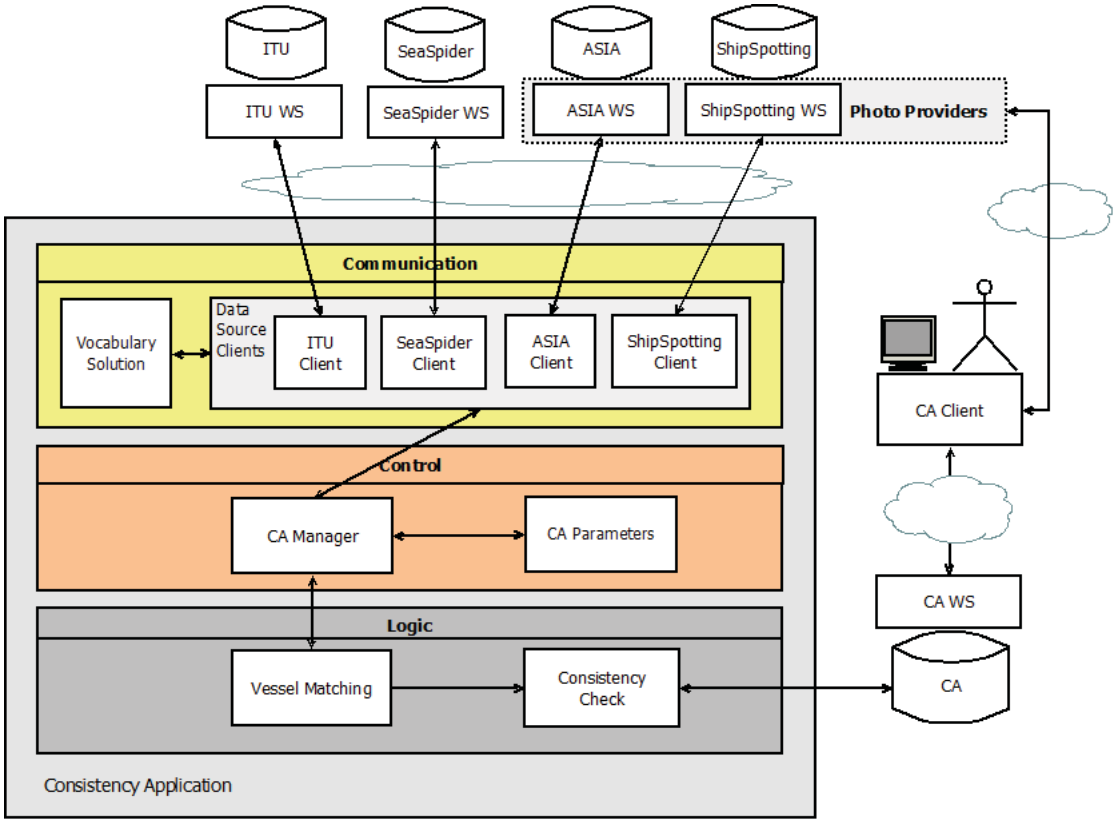

Figure 5: *Consistency Application functional diagram and context, including the ShipSpotting data source and its client.*

3.2 Vocabulary Solution

The data originating from the diverse data sources need to be aligned with the CA vocabulary. Specifically, this means a mapping is required to relate ship properties from the different data source vocabularies to the CA ship properties.

To fulfill this goal, the vocabulary solution offers methods to translate each data source response (expressed as a list of ship objects) into a list of ship objects as expressed in the CA vocabulary; i.e. CAShips. It also offers methods to translate a CA query object into a particular data source query object. In other words it creates a query that is understandable for each data source, but based on a single CA query. This query is then directly sent to the data source by Simple Object Access Protocol (SOAP) messaging.

For the current case of the ShipSpotting data source, two methods were developed to fulfill these goals:

- `shipspottingToCAShip`: transforming shipspottingShip into caShip
- `caToShipspottingQuery`: transforming caQuery into shipSpottingQuery

The following is a pseudo code example to illustrate how `shipspottingToCAShips` aligns data coming in the CA.

```
CAShip shipspottingToCAShips( shipspottingShip )
{
CAShip.setName( shipspottingShip.getName() );
CAShip.setImoNumber( shipspottingShip.getImo( ));
...
return CAShip;
}
```

In order to compare ship attributes, extra information manipulation was required. The following methods were implemented to modify the data:

1. `shipspottingToCAType`,
2. `shipspottingToCAFlag`.

The following sections describe the logic behind these two methods.

3.2.1 Type Alignment

As mentioned in section 2.2.2, there are ship and photo categories in ShipSpotting. For about 73% of ships in the ShipSpotting DB, no ship category is provided, but in all cases, the photo's category is available. Since the photo's category is most of the time similar to the ship's category (when provided), it is used to define the CA type. The CA Type is the photo's category added to the ship's category (when available and different from the photo's category).

Moreover, the photo's category is always provided in plural form. Among all data sources, ShipSpotting is the only one to provide the type (category in ShipSpotting's vocabulary) in plural. This particularity causes problems at the string comparison level, especially when dealing with short words. For instance, the Levenshtein score between *tug* and *tugs* is 0.75, which is under the minimal similarity threshold (0.85). To solve that problem, those words used for photo categories were made singular, rather than plural. This method, for which several Java implementations are available in open source, is a classic rule-based algorithm using the basic English language rules of singularization. This singularization transformation is not necessary for ship's category, since it is always provided in its singular form.

The examples in Table 2 illustrate the `shipspottingToCAType` algorithm.

Photo Category	Ship Category	CA Type
Tugs		Tug
Ferries	Cargo	Ferry/Cargo
Tugs	Tug	Tug

Table 2: *Type alignment examples*

CA Flag	Synonyms
Korea (South)	Republic of Korea, Korea (Republic of), South Korea,Korea
Korea (North)	Democratic People's Republic of Korea, North Korea
Russia	Russian Federation
United Kingdom	United Kingdom of Great Britain and Northern Ireland
Ireland	Irish Republic
Syria	Syrian Arab Republic
Vietnam	Viet Nam
United States of America	United States, United States of Ame
Wallis and Futuna Islands	Wallis and Futuna Is
Saint Vincent and the Grenadines	Saint Vincent and The, Saint Vincent & The
Saint Kitts and Nevis	Saint Kitts and Nevi, Saint Kitts-Nevis
French Antarctic Territory	French Antarctic Ter
Argentina	Argentine Republic

Table 3: *Flag Alignment*

3.2.2 Flag Alignment

It was mentioned in [1] section 4.1 that slightly different names can be used to describe the same country. Since the goal of the Consistency Application is to compare similar data items to develop added confidence in these data, it was decided to align, when possible, the country names.

With the addition of ShipSpotting to the comparison process, additional country name synonyms had to be considered for the flag alignment process. In order to encapsulate country name mapping, a new class was added to the vocabulary solution: `FlagAlignment`.

This class contains a method to transform a given flag value based on the mapping between the country name synonyms described in Table 3. This synonyms list covers all data source country names. However, it is still expected that other cases implying different country names for the same country will arise. In cases where a difference is found, it is straightforward to add a synonym to the `FlagAlignment` class mapping.

All data source flag mapping methods (including `shipspottingToCAFlag`) now use the `FlagAlignment` alignment method described above.

4 Consistency Application Client

The CA web client is used to visualize the CA comparison results and to geo-locate the ships information. It communicates with the CA web service to get comparison results for ships corresponding to query parameters.

The ShipSpotting web service was exposed as a picture provider for the CA client. The CA client communicates with both the ASIA and the ShipSpotting web services to get the ship photos, when available (see Figure 6).

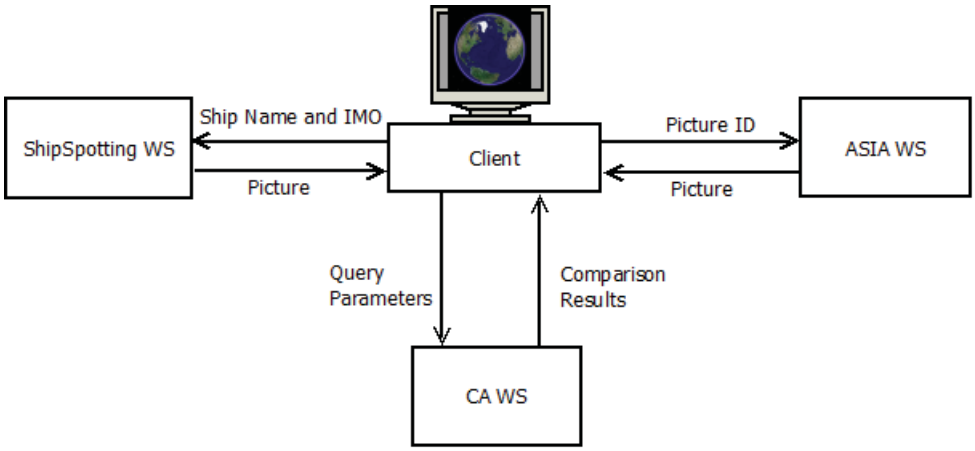

Figure 6: *Consistency Application web client interaction with the Compare-MDA framework. The client queries the CA web service to get the comparison results and the ASIA and ShipSpotting web services for the pictures.*

To get and display photos from the ShipSpotting WS, two modifications were made to the client:

1. Addition of a component to communicate with the ShipSpotting WS (similar to the communication component with ASIA WS). It invokes the `getPhoto` operation with the ship name and IMO number when available (both as provided by ShipSpotting).

2. Addition of a user interface component to select the picture provider.

The ship's photo is displayed at the same place as the previous CA implementation. However, on the right side of the picture there are now radio buttons to select between ASIA and ShipSpotting photos (when available). This allows the switching of photo source. See Figure 7 for a case where both pictures are available.

The photo provider name "ShipSpotting" is a hyper link to `www.shipspotting.com`. Clicking on it opens the ShipSpotting web page in another tab. The ShipSpotting's copyright policy requires that when we display a photo from ShipSpotting, we are displaying

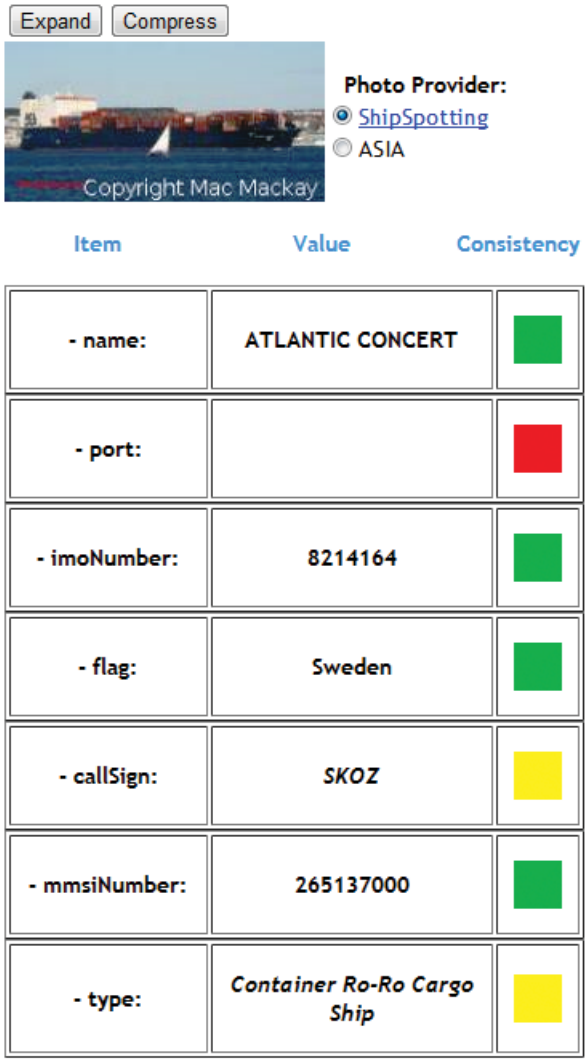

Figure 7: *Comparison Window with picture providers selection rendered by radio buttons.*

- only a thumbnail
- name of the photographer with the word ”Copyright”
- link to the ShipSpotting website

The actual display respects these requirements.

When one or both of the picture providers (ASIA and ShipSpotting) do not provide information for a given ship, the associated radio button is disabled. Figure 8 illustrates some of the display possibilities.

Finally, an additional change was made to the CA client. The ship item *Name* now always

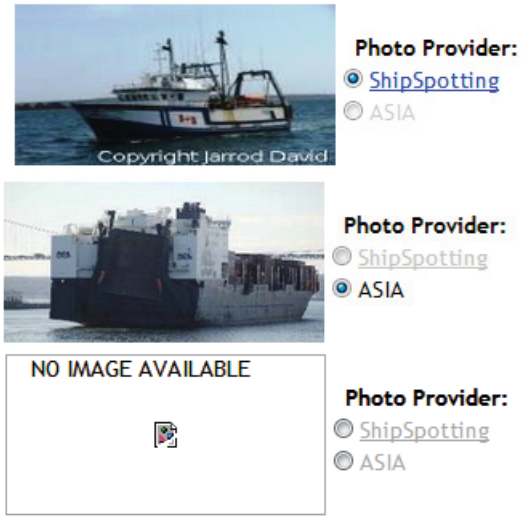

Figure 8: *Ship image when 1) ASIA, 2) ShipSpotting, 3) ASIA and ShipSpotting do not provide information for a ship.*

appears first in the Comparison Window. This change is not related to the ShipSpotting data source addition. It was performed to allow rapid identification of the ship currently being compared by the application.

5 Installation

The Compare-MDA framework now includes seven web services:

- ASIA WS,
- SeaSpider WS,
- ITU WS,
- ShipSpotting WS,
- CA WS,
- CA client WS and
- jUDDI WS;

As well, there is one application: the CA.

Six databases are feeding the web services and the application. Those databases are:

- the `asia` DB,
- the `seaspider` DB,
- the `itu_db` DB,
- the `shipspotting` DB,
- the `consistency` DB and
- the `juddi` DB.

Each of these components are illustrated in Figure 1.

The addition of the ShipSpotting data source to the Compare-MDA framework impacts the installation procedure described in the *Installation and developer guide for the MIKM consistency application* [2]. The impact is limited to the following additions to the procedure:

- Creation and transfer of the shipspotting DB
- Deployment of the shipspotting WS
- Addition of the ShipSpotting project in the development environment.

This section details these modifications and identifies which sections of [2] are impacted.

5.1 ShipSpotting Database

The ShipSpotting WS exposes the information contained in the ShipSpotting database, called `shipspotting` (see section 2.2.2 of this document for details about the database).

The following commands have to be added to the database transfer procedure described in sections 3.1 and 3.2:

```
mysql -u root -p
<enter password>
CREATE DATABASE shipspotting;
```

The following command will load the ShipSpotting database given the content in the `shipspotting.sql` file.

```
mysql -u root -p < shipspotting.sql
```

NOTE: Provide the relative path for `shipspotting.sql`. You can also use MySQL Administrator to transfer the database.

5.2 ShipSpotting Web Service Deployment

The ShipSpotting WS has to be deployed in Tomcat. To install and deploy the web service `ShipSpottingDataSourceService.aar`, follow instructions given in section 5.2 of [2].

5.3 Development Environment

The ShipSpotting web service source code is not required to run the Consistency Application. It is however recommended to create an Eclipse project for ShipSpotting to quickly generate web service archives (which are required if, for instance, the `shipspotting` database is moved) or for further development.

Section 8.3 of [2] describes how to create Eclipse projects from Java code. Follow this methodology to create the `shipspottingWS` project from the code `shipspotting_ws` (on the installation CD). Special care should be taken at the following steps.

Step 4 The project `shipspottingWS` has the following dependencies: Axis2 library and `mysql-connector-java-3.0.17-ga-bin.jar`.

Step 5 Add the `shipspottingWS` to the `consistencyApp` project's build path.

In section 8.3.1, item 3, it is mentioned that if `asiaWS` project has been modified, to export it as `asia.jar` to `caClientWS\public_html\WEB-INF\lib` (there is no need to include WebContent in the archive). The same comment applies if `shipspottingWS` project is modified: export it as `shipspotting.jar` to `caClientWS\public_html\WEB-INF\lib`.

References

[1] St-Hilaire, M.-O. (2010), Determining the consistency of information between multiple systems used in maritime domain awareness, (DRDC Atlantic CR 2010-056) Defence R&D Canada – Atlantic.

[2] M.-O. St-Hilaire, M. Mayrand (2010), Installation and Developer Guide for the MIKM Consistency Application, (DRDC Atlantic CR 2010-057) Defence R&D Canada – Atlantic.

[3] (2005), SOAP Message Transmission Optimization Mechanism, W3C Recommendation (online), W3C, http://www.w3.org/TR/soap12-mtom/.

[4] (2006), MTOM Guide -Sending Binary Data with SOAP (online), Apache Software Foundation - Axis2, http://ws.apache.org/axis2/1_0/mtom-guide.html.

List of symbols/abbreviations/acronyms/initialisms

AIS	Automatic Identification System
ASIA	Automated Ship Image Acquisition
CA	Consistency Application
DB	Database
GE	Google Earth
GUI	Graphical User Interface
HTML	HyperText Markup Language
IMO	International Maritime Organization
ITU	International Telecommunication Union
MDA	Maritime Domain Awareness
MTOM	Message Transmission Optimization Mechanism
PHP	PHP: Hypertext Preprocessor
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
UID	ship unique identifier
URL	Uniform Resource Locator
WS	Web Service
XML	eXtensible Markup Language

This page intentionally left blank.

Distribution list

DRDC Atlantic CR 2011-038

Internal distribution

1 F. Desharnais

1 T. Hammond

2 A. W. Isenor (1 hardcopy; 1 CD)

1 L. Lapinski

1 A. MacInnis

1 M. McIntyre

1 S. Webb

3 Library (1 hardcopy; 2 CDs)

Total internal copies: 11

External distribution

1 Library and Archives Canada,
Atten: Military Archivist Governments Records Branch

1 NDHQ/DRDKIM 2-2-5

2 Marie-Odette St-Hilaire (1 hardcopy; 1 CD)
OODA Technologies Inc.
4891 Av. Grosvenor,
Montreal Qc, H3W 2M2

1 Senior Watch Officer
Joint Task Force (North) Evans Building,
PO Box 6666, Station Main
Yellowknife, NT, X1A 2R3

1 Senior Staff Officer RJOC(A)
PO Box 99000 Stn Forces,
Halifax, NS B3K 5X5

1 Laura Ozimek DSTM 5
DRDC Corporate 305 Rideau Street Ottawa

Total external copies: 7

Total copies: 18

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (The name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.)

OODA Technologies Inc.
4891 Av. Grosvenor
Montreal, QC H3W 2M2

2. SECURITY CLASSIFICATION (Overall security classification of the document including special warning terms if applicable.)

UNCLASSIFIED
(NON-CONTROLLED GOODS)
DMC A
REVIEW: GCEC JUNE 2010

3. TITLE (The complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C, R or U) in parentheses after the title.)

Addition of an Open-Web Data Source Service to the Consistency Application

4. AUTHORS (last name, followed by initials – ranks, titles, etc. not to be used)

St-Hilaire, M.-O.

5. DATE OF PUBLICATION (Month and year of publication of document.)

April 2011

6a. NO. OF PAGES (Total containing information, including Annexes, Appendices, etc.)

38

6b. NO. OF REFS (Total cited in document.)

4

7. DESCRIPTIVE NOTES (The category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Contract Report

8. SPONSORING ACTIVITY (The name of the department project office or laboratory sponsoring the research and development – include address.)

Defence R&D Canada – Atlantic
PO Box 1012, Dartmouth, NS Canada B3A 3Z7

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)

Project 11HL, 11HO

9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)

W7707-115277

10a. ORIGINATOR'S DOCUMENT NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.)

DRDC Atlantic CR 2011-038

10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

11. DOCUMENT AVAILABILITY (Any limitations on further dissemination of the document, other than those imposed by security classification.)

(X) Unlimited distribution
( ) Defence departments and defence contractors; further distribution only as approved
( ) Defence departments and Canadian defence contractors; further distribution only as approved
( ) Government departments and agencies; further distribution only as approved
( ) Defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (Any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in (11) is possible, a wider announcement audience may be selected.))

13. ABSTRACT (A brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

The Consistency Application was built upon web services in a SOA paradigm. The CA compares the information from diverse data sources, all providing information relevant to MDA. The CA allows an assessment of the information consistency across the data sources (i.e., ASIA database, SeaSpider database, ITU database) and visualization of the consistency results within a Google Earth environment.

In this work, an additional data source was added to the CA, namely the ShipSpotting data source available at `www.shipspotting.com`.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

trust
consistent information
multiple information sources
maritime domain awareness
MDA

This page intentionally left blank.

Defence R&D Canada

Canada's leader in defence
and National Security
Science and Technology

R & D pour la défense Canada

Chef de file au Canada en matière
de science et de technologie pour
la défense et la sécurité nationale

DEFENCE **DÉFENSE**

www.drdc-rddc.gc.ca