2026

Netpas Enterprise API

Version 7.06

Development Guide

(XML/JSON)

Netpas Development Team / Seafuture Inc.

support@netpas.net

1. Abstract

- Netpas Enterprise API (NEA) provides two (2) types of APIs – JSON and XML, by providing distance and route information between port and port.

NEA provides the following information:

- Optimal path and distance between ports and coordinates (Including waypoint coordinates and routing points)
- Route and distance through ECA
- Optimum path and distance according to Piracy Option
- Bypass route if passing ECA
- License Information

2. Definition

A. NEA user: Licensee's system or computer trying to access Netpas Enterprise API server

B. NEA server: Netpas Enterprise API server providing route and distance

3. How to Work

You will get the result data back with JSON or XML once you send parameters by "GET" method to the URL specified in the development document. Use the parser of your development language to extract the value.

A. You can either use Netpas Distance Map or eMap on your own as a viewer.

- If you are to use Netpas Distance Map as a viewer, please refer to the following document provided at the link below to interface:
  https://www.netpas.net/assets/files/download/RDML_For_Netpas_Enterprise_API.pdf

  Netpas Distance is to be installed on each end user's PC in order to view the map. In such case, the user can use Netpas Distance as a viewer only and cannot use other Netpas Distance functions.

- If you are to use eMap such as Google or BING map, you need to interface it with Netpas NEA data through parsing by your side and with your charge.

B. You can input maximum twenty (20) ports or coordinates for one (1) API call. The queries are to be counted as below:

- For "Port ↔ Port", maximum five (5) ports are counted as one (1) query.
- For "Coordinate ↔ Coordinate" or "Coordinate ↔ Port," two (2) coordinates or ports are

counted as one (1) query.

- For "Bypass ECA function"(API 20), twice as many queries as API 10. are counted.

C. You will get an error message 403 if you exceed your daily limit. (Forbidden: Your calculation number is over the day limit and blocked.)

4. APIs Provided

- Distance and route calculation between port and port **(API 10)**
- Distance calculation bypassing ECA **(API 20)**
- License Information (**API 30**)

5. License

Netpas Enterprise API is protected by copyright law and international treaties. Unauthorized reproduction, copying, storing, recording, transmitting or distribution of this program and database, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under the law.

6. Apply for Demo Trial

In order to try out Netpas API service, you need to apply for "Demo Request" at the link below: http://www.netpas.net/products/detail/nea

Once reviewing and confirming your request, the Access and Pin codes will be sent to you to log in. You can then try Netpas API demo for 30 days from the approval date with limited port list.

7. APIs Specification

API 10 : GET_DISTANCE

The API returns optimal route and distance after receiving Point (Port or Coordinates).

The request format is as follows.

```
https://api.netpas.net/nea/v7/{data_format}/get_distance/?pincode={pin_code}&access_cod
e={access_code}&piracy_code={piracy_code}&ports={ports1}&ports={ports2}&canal_pass_code
={canal_pass_code}&use_local_eca={true or false}&use_china_local_eca={true or false}
```

ex)https://api.netpas.net/nea/v7/**json**/get_distance/?pincode=**netpas_pincode**&access_code=**a523gfc7fH2r**&piracy_code=**000**&ports=**Corinth**&ports=**Busan**&canal_pass_code=**110**&use_local_eca=**false**

Parameters	Description
data_format	Select the data type to return XML or JSON
pin_code	"pincode" you receive when you purchase NEA license.
access_code	"access_code" you receive when you purchase NEA license.
piracy_option	You can calculate the route that bypasses the pirate area announced by JWLA. *** Ref. *Appendix B***
ports	Port Name, Routing Point Name, Coordinates(Lat Lon) ex) Pusan, SUEZ CANAL (RP),37.38W 46.40N
canal_pass_code	You can calculate the routes that exclude specific canal. *** Ref. *Appendix B***
use_local_eca	Select either enable or disable local ECA
use_china_local_eca	Select either enable or disable China Territorial Waters. If use_local_eca is not present or set to false, it will not be applied.

<<tableAPI_10-1: request parameters >>

Description of returned Elements

Element Name	Root		
Sub Elements			
	Name	**Type**	**Description**
	today_used	int	The number of "*getDistance*" used today
	day_limit	int	The number of "*getDistance*" available per day
	totalDistance	double	Total distance
	totalDistanceSECA	double	The distance passing ECA of whole Distance.
	cross_seca	boolean	If "true", cross the ECA.
	token	String	Token for ECA bypass
	Section	Section	Section Element Array In case N port Names, (N-1) sections returned
	code	Int	*** Ref. *Appendix B***
	message	String	Return message
	route_warning	boolean	If ”true”, the system detects abnormal route or distance values in P-to-P calls (point-to-point, port-to-point, or point-to-port).

<< tableAPI_10-2: return Elements >>

Element Name	Section		
Sub Elements			
	Name	**Type**	**Description**
	from_port	port	Departure port
	to_port	port	Arrival port
	distance	double	distance of the Section
	distance_seca	double	ECA distance of the Section
	waypoint	waypoint	Section Element Array In case N ports, (N-1) sections returned

<< tableAPI_10-3: return Elements >>

Element Name	from_port, to_port		
Attribute			
	Name	**Type**	**Description**
	name	String	Port name
	seca_port	boolean	Check out whether it is ECA port. Even the ports are out of ECA, if the ports belong to specific country (e.g. France, Germany and etc.) or if the ports are designaged as ECA port (e.g. Hong Kong, Macao and etc.)， they are considered as ECA Port and ”seca_port” option returns ”true” value.

<< tableAPI_10-4: return Elements >>

Element Name	Waypoint		
Sub Elements			
	Name	**Type**	**Description**
	lat	double	Latitude of waypoint
	lon	double	Longitude of waypoint
	name	double	Total distance
	seca	boolean	Check out whether it is ECA from waypoint to the next waypoint or not. Ref. appendix A

<< tableAPI_10-5: return Elements >>

Returned data example(JSON) :

```
{
    "code":200,
    "message":"OKs",
    "route_warning":false,
    "day_limit":100,
    "today_used":4,
    "cross_seca":true,
    "total_distance":1876.9234,
    "total_seca_distance":778.7666,
    "section":[
        {"from_port":
            {"name":" CAPE FAREWELL (DK)","seca_port":false},
        "to_port":
            {"name":" HAMILTON (BM)","seca_port":false},
        "distance":1876.9234,
        "distance_seca":778.7666,
        "waypoint":[
            {"lat":59.7832984924316, "lon":-43.9000015258789, "name":"CAPE FAREWELL (DK)", "seca":false},
            {"lat":59.7794908471406, "lon":-43.893168135546, "seca":false},
            {"lat":59.6920166015625, "lon":-43.912815093994, "seca":false},
            {"lat":52.2562594558948, "lon":-49.841701515539, "seca":true},
            {"lat":47.5703620910645, "lon":-52.572559356689, "seca":true},
            {"lat":47.2828788757324, "lon":-52.656101226806, "seca":true},
            {"lat":46.9967498779297, "lon":-52.794723510742, "seca":true},
            {"lat":46.5792999267578, "lon":-53.058979034423, "name":"CAPE RACE (RP)","seca":true},
            {"lat":40.8438512122042, "lon":-58.416906721012, "seca":false},
            {"lat":32.3454437255859, "lon":-64.607330322265, "seca":false},
            {"lat":32.341381072998, "lon":-64.665893554687, "seca":false},
            {"lat":32.2503547668457, "lon":-64.784645080566, "seca":false},
            {"lat":32.25, "lon":-64.833297729492,"name":"HAMILTON (BM)", "seca":false}
        ]
    }]
}
```

Return data example(XML) :

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<route code="200" message="OKs" route_warning="false">
    <day_limit>100</day_limit>
    <today_used>3</today_used>
    <cross_seca>true</cross_seca>
    <total_distance>1876.9234</total_distance>
    <total_seca_distance>778.7666</total_seca_distance>
    <section>
        <from_port seca_port="false" >CAPE FAREWELL (DK)</from_port>
        <to_port seca_port="false" > HAMILTON (BM)</to_port>
        <distance>1876.9234</distance>
        <distance_seca>778.7666</distance_seca>
        <waypoint lat="59.7832984924316" lon="-43.900001525878" name="CAPE FAREWELL (DK)" seca="false" />
        <waypoint lat="59.7794908471406" lon="-43.893168135546" seca="false" />
        <waypoint lat="59.6920166015625" lon="-43.912815093994" seca="false" />
        <waypoint lat="52.2562594558948" lon="-49.841701515539" seca="true" />
        <waypoint lat="47.5703620910645" lon="-52.572559356689" seca="true" />
        <waypoint lat="47.2828788757324" lon="-52.656101226806" seca="true" />
        <waypoint lat="46.9967498779297" lon="-52.794723510742" seca="true" />
        <waypoint lat="46.5792999267578" lon="-53.058979034423" name="CAPE RACE (RP)" seca="true" />
        <waypoint lat="40.8438512122042" lon="-58.416906721012" seca="false" />
        <waypoint lat="32.3454437255859" lon="-64.607330322265" seca="false" />
        <waypoint lat="32.341381072998" lon="-64.665893554687" seca="false" />
        <waypoint lat="32.2503547668457" lon="-64.784645080566" seca="false" />
        <waypoint lat="32.25" lon="-64.833297729492 "name="HAMILTON (BM)" seca="false" />
    </section>
</route>
```

API 20 : GET_DISTANCE_BYPASS_SECA

- Same parameters as API 10. are requested for Bypass ECA option. (*Ref. Appendix D). Twice as many queries as API 10. are to be counted even if your route is not crossing the ECA.

If Bypass ECA option is set as "OPTIMIZED", the seca_cost_ratio by following calculation must to be returned together

[Daily Hire + ECA consumption value] ÷ [Daily Hire + Normal consumption value]

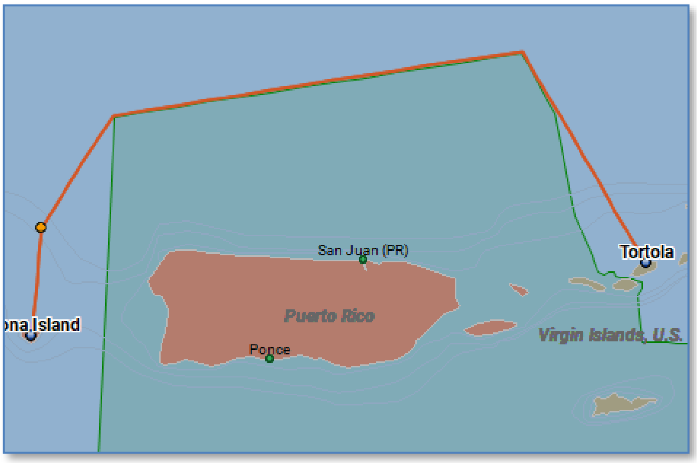
The 'consumption value' is calculated as the consumed bunker quantity multiplied by the unit price. (consumed bunker quantity x unit price)

<< equation 1: ECA cost ratio (All Consumption is daily sea consumption)>>

ECA cost ratio example

```
Daily Hire      : 5000

Normal
  VLSFO Price : 346
  VLSFO Cons. : 29

  MDO Price   : 509
  MDO Cons.   : 0.5

ECA
  ULSFO Price : 509
  ULSFO Cons. : 30

  MGO Price   : 518
  MGO Cons.   : 0.6

ECA cost ratio : 1.077
  [5000 + (30 X 372 + 0.6 X 518)] / [5000 + (29 X 346 + 0.5 X 509)] = 1.077
```

The route bypassed to **API 20** is as follows.



<< pictureAPI_20-3: the route bypassing the ECA >>

The request format is as follows.

(seca_cost_ratio is needed only when the seca bypass option is "OPTIMIZED")

```
https://api.netpas.net/nea/v7/{data_format}/get_distance_bypass_seca/?pincode={pin_code}&access_code={access_code}&piracy_code={piracy_code}&ports={ports1}&ports={ports2}&canal_pass_code={canal_pass_code}&seca_bypass_option={seca_bypass_option}&seca_cost_ratio={seca_cost_ratio}&use_local_eca={true or false}&use_china_local_eca={true or false}
```

ex)https://api.netpas.net/nea/v7/**json**/get_distance_bypass_seca/?pincode=**netpas_pincode**&access_code=**a523gfc7fH2r**&piracy_code=**000**&ports=**Corinth**&ports=**Busan**&canal_pass_code=**110**&use_local_eca=**false**&use_local_eca=**false**&seca_bypass_option=**SHORTEST**

The returned route format is the same as **API 10**.

API 30 : LICENSE_INFO

The API returns your license information.

The request format is as follows.

```
https://api.netpas.net/nea/v7/{data_format}/license_info/?pincode={pin_code}&access_code={access_code}
```

ex)https://api.netpas.net/nea/v7/**json**/license_info/?pincode=**netpas_pincode**&access_code=**a523gfc7fH2r**

Description of returned Elements

Element Name	license		
Sub Elements			
	Name	**Type**	**Description**
	today_used	int	The number of "*getDistance*" used today
	day_limit	int	The number of "*getDistance*" available per day
	end_date	String	License expiring date
	recently_used	recently_used	Recently Calc Log Array (365 days)
	code	Int	*** Ref. *Appendix B***
	message	String	Return message

<< tableAPI_30-1: return Elements >>

Element Name	calc_log		
Sub Elements			
	Name	**Type**	**Description**
	count	int	The number of "*getDistance*" used the day
	date	String	Logging date

<< tableAPI_30-2: return Elements >>

Returned data example(JSON) :

```
{
        "code":200,
        "message":"OKs",
        "day_limit":100,
        "today_used":4,
        "end_date":"Aug/18/2018 12:18",
        "recently_used":
                {"calc_log":[
                        {"date":" Jun/01/2018","count":2},
                        {"date":" May/31/2018","count":6},
                        {"date":" Nov/07/2017","count":1}
                }]

}
```

Return data example(XML) :

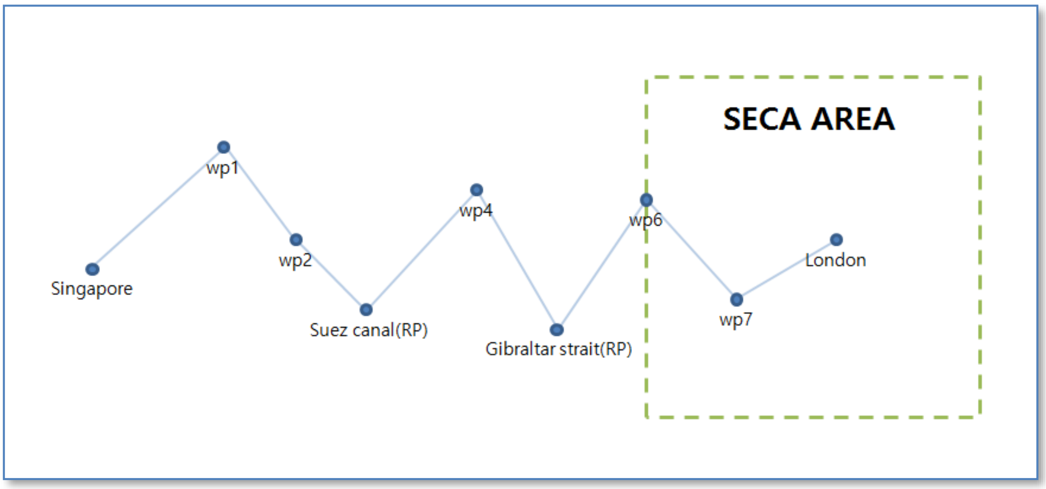
```
<license code="200" message="OKs">
        <end_date> Aug/18/2018 12:18 </ end_date >
        <day_limit>100</day_limit>
        <today_used>3</today_used>
        <recently_used>
                <calc_log count="2" date="Jun/01/2018" />
                <calc_log count="6" date="May/31/2018" />
                <calc_log count="1" date="Nov/07/2017" />
        </recently_used>
</license>
```

Appendix A – ECA

Here are waypoints between Singapore – London. (These numbers are not correct since used for expample purpose only)

Waypoints	**lat**	**lon**	**name**	**seca**
Singarpore	1.2	103.4	Singarpore	False
Wp1	5.3	80.2		False
Wp2	11.2	45.2		False
SUEZ CANAL (RP)	29.9	32.5	SUEZ CANAL (RP)	False
Wp4	30.2	15.1		False
GIBRALTAR STRAIT (RP)	35.9	-5	GIBRALTAR STRAIT (RP)	False
Wp5	40	-3		**True**
Wp6	45	-2		**True**
London	51.5	-0.1	London	N/A

<<tableA-1>>



<<Picture-1>>

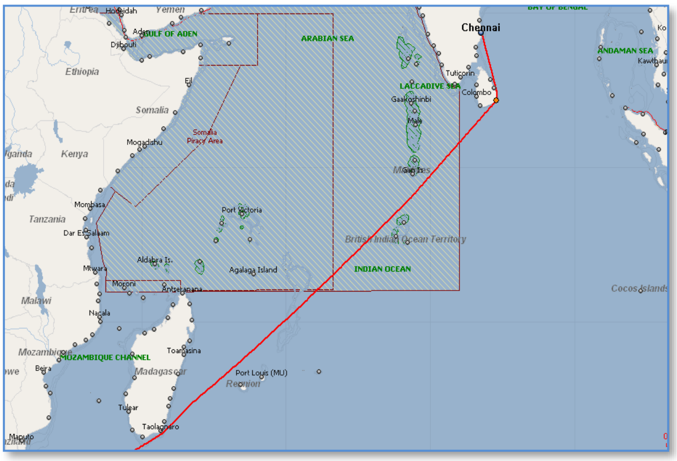
You will know the route wp5-wp6-London crosses ECA because "*seca*" values of wp6 and wp7 are true. The false in London does mean nothing because it is for a route between London – next way point. There is no next waypoint any more.

Appendix B – CODE TABLE

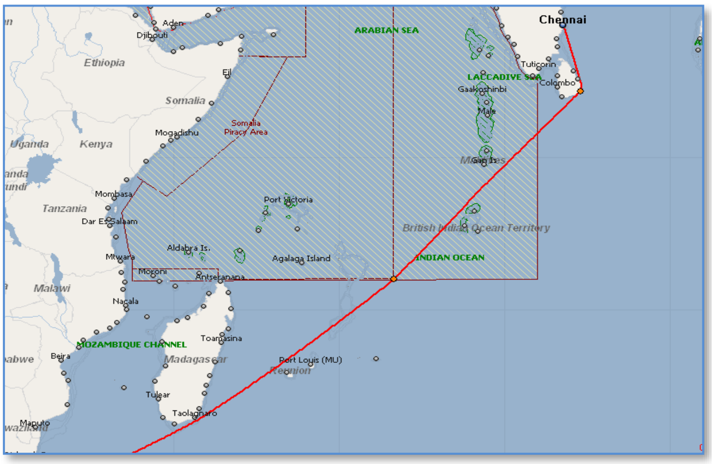
Piracy Code Table

Piracy Code	Description
000	The latest update of piracy area.
001	The shortest path
002	250nm outer route from Somalia east coast
003	600NM outer route from Somalia east coast
004	JWLA015(2nd Aug 2010)
005	JWLA016(16th Dec 2010)
006	JWLA016 Up to Mumbai
007	JWLA016 Max Avoid Route
008	JWLA025(21st Sep 2020)
009	JWLA033(3rd Mar 2026)

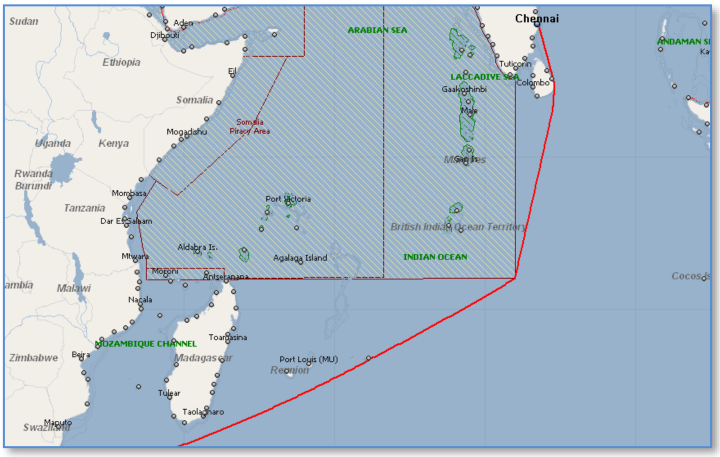
<<tableB-1: parameter code of piracy path >>



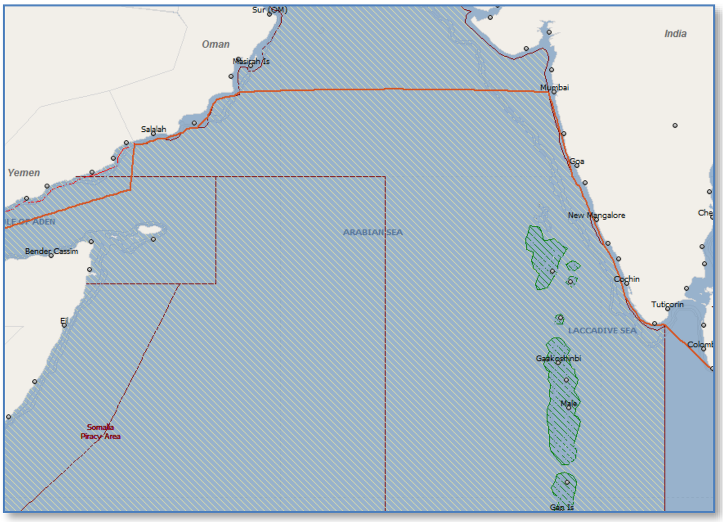
<<pictureB-1: example - the shortest path >>



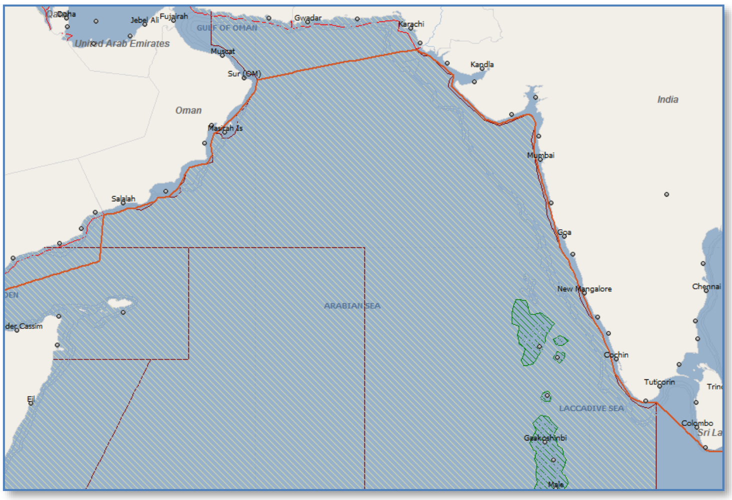
<<pictureB-2: example - JWLA015(2nd Aug 2010) >>


Sudan
Aden
Djibouti
Ethiopia
ARABIAN SEA
Chennai
ANDAMAN SE
Tuticorin
LACCADIVE SEA
Gaakoshinbi
Colombo
Male
Somalia
Eil
Mogadishu
Somalia
Piracy Area
Uganda
Kenya
Rwanda
Burundi
Mombasa
Tanzania
Dar Es Salaam
Port Victoria
British Indian Ocean Territory
Aldabra Is.
Agalega Island
INDIAN OCEAN
Mtwara
Moroni
Antsiranana
Cocos
ambia
Malawi
Nacala
MOZAMBIQUE CHANNEL
Toamasina
Zimbabwe
Beira
Madagascar
Port Louis (MU)
Reunion
Tulear
Taolagnaro
Swaziland
Maputo

<<pictureB-3: example - JWLA016(16th Dec 2010)>>


Sur (OM)
Oman
India
Masirah Is
Mumbai
Salalah
Goa
Yemen
ARABIAN SEA
New Mangalore
GULF OF ADEN
Bender Cassim
Cochin
Eil
Tuticorin
LACCADIVE SEA
Colombo
Gaakoshinbi
Male
Somalia
Piracy Area
Gan Is

<<pictureB-4: example - JWLA016 Up to Mumbai>>

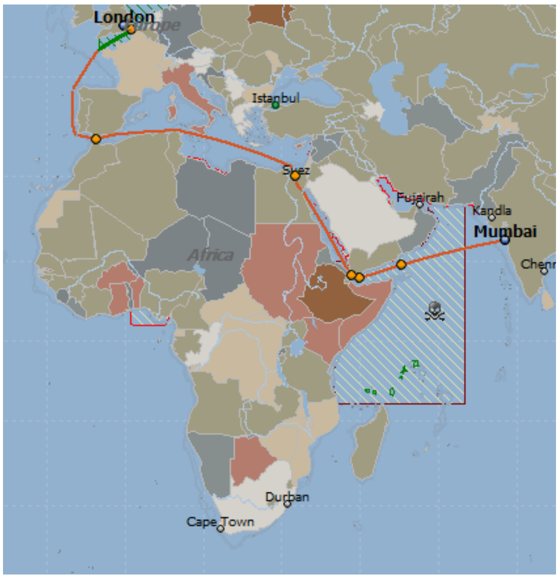

Doha
Jebel Ali
Fujairah
Gwadar
Karachi
GULF OF OMAN
United Arab Emirates
Muscat
Kandla
Sur (OM)
India
Oman
Masirah Is
Mumbai
Salalah
Goa
New Mangalore
Chennai
ARABIAN SEA
der Cassim
Cochin
Tuticorin
Eil
LACCADIVE SEA
Colombo
Sri La
Gaakoshinbi
Male

<<pictureB-5: example - JWLA016 Max Avoid Route>>

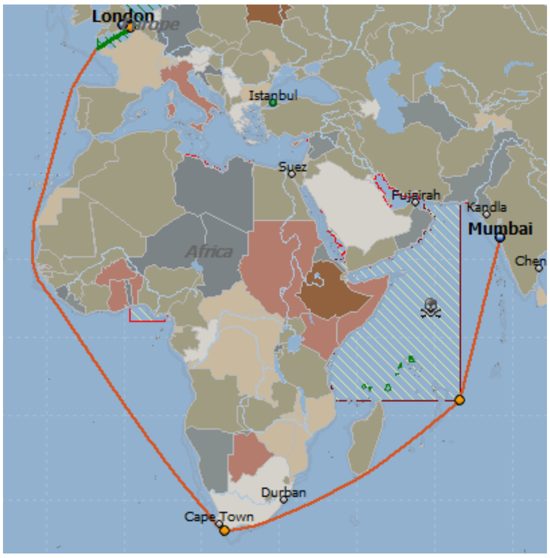
Canal Pass Code Table

Canal pass code	Description
000	Exclude all canals
001	Exclude ‘Suez’ and ‘Panama’
010	Exclude ‘Suez’ and ‘Kiel’
011	Exclude ‘Suez’
100	Exclude ‘Panama’ and ‘Kiel’
101	Exclude ‘Panama’
110	Exclude ‘Kiel’
111	Not exclude canals

<<tableB-2: parameter code of canal pass>>



<<pictureB-6: example - Not exclude "Suez" route (Code 100, 101, 110, 111)>>



<<pictureB-7: example - Exclude "Suez" route (Code 000, 001, 010, 011)>>

Return Code Table

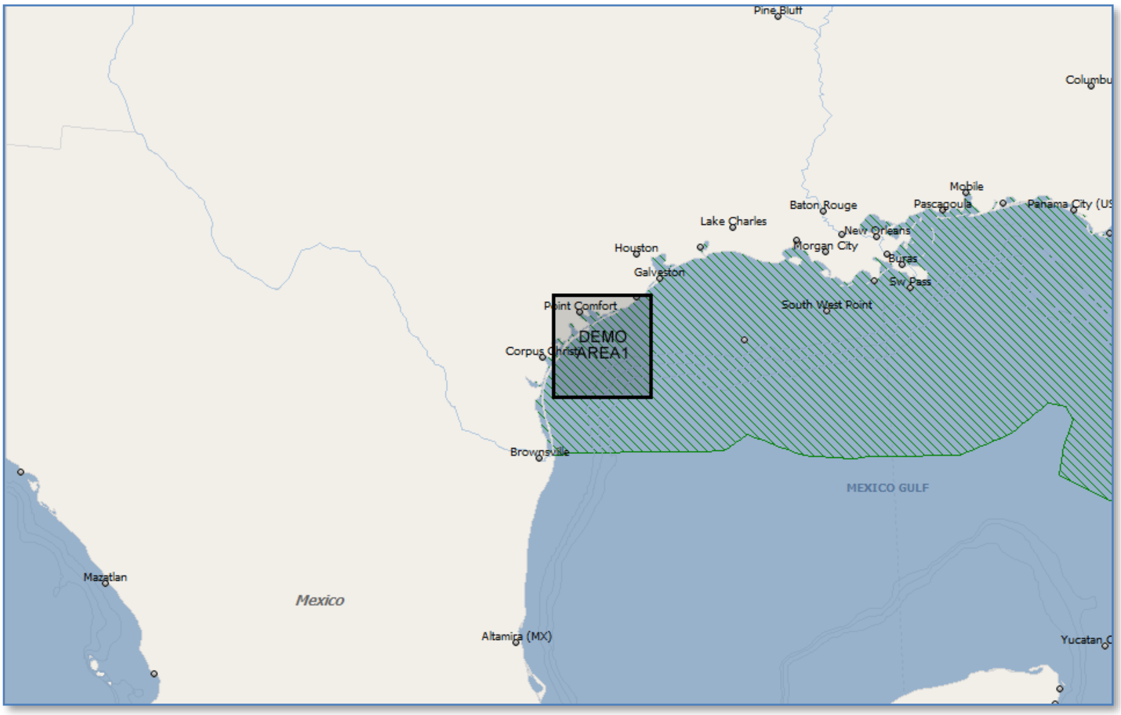
CODE	DESCRIPTION
200	Success
204	No Contents
210	No Port Name
221	Invalid Access Code
231	Day Limit Over
250	Could not find the route
400	Bad Request
401	Unauthorized
403	Forbidden
405	Method Not Allowed";
500	Internal Server Error

<<tableB-3: return code>>

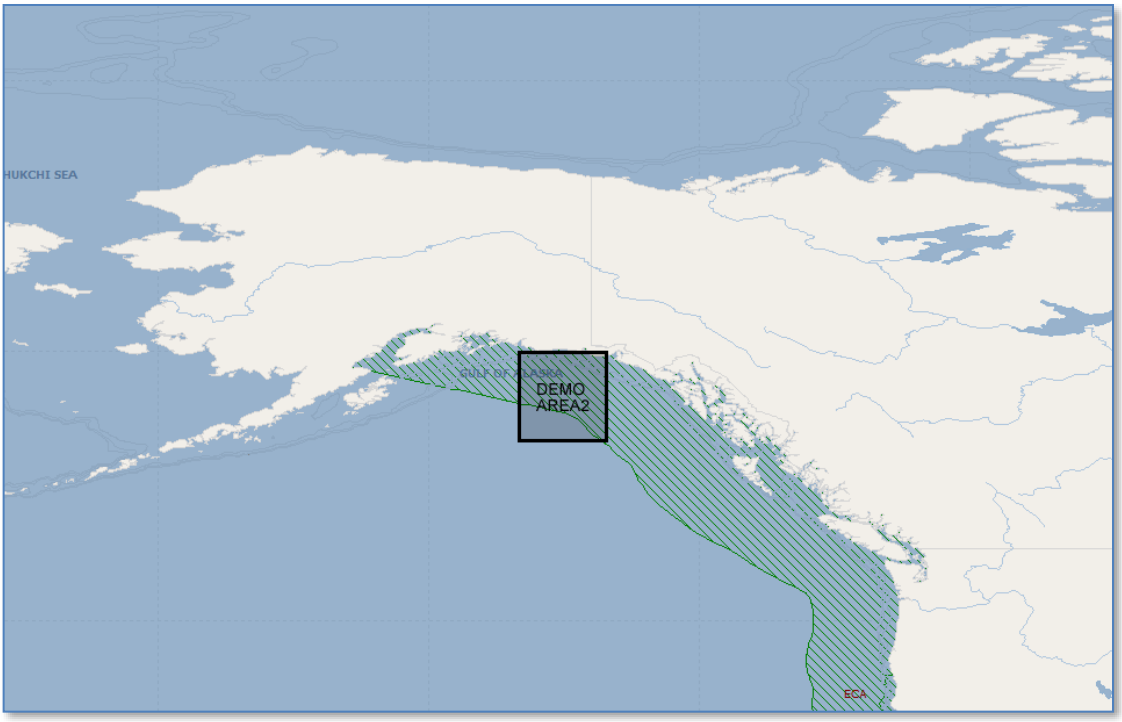
Appendix C – DEMO COORDINATES

Demo users can use "Coordinates to Port"and "Coordinates to Coordinates"function within limited area as below pictures.

Area1 : 27N to 29N , 95W to 97W



Area2 : 55N to 60N , 140W to 145W

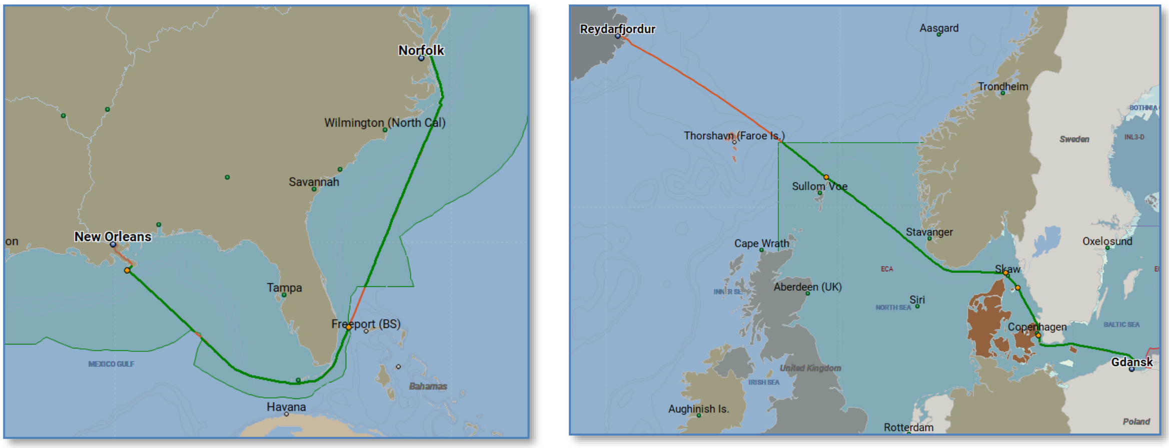


Appendix D – Bypass ECA Option

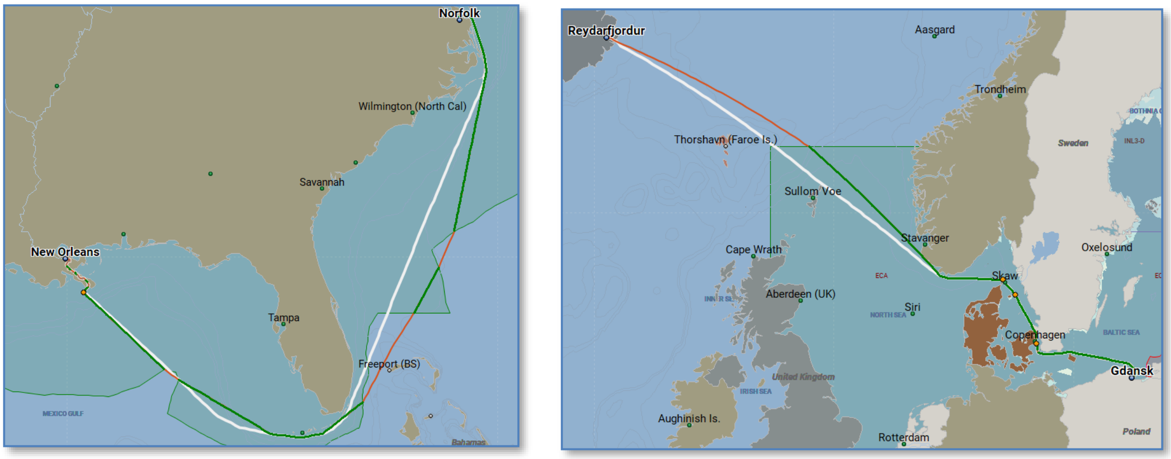
Bypass ECA Option Table

Option value	Description
NONE	Non-Bypass ECA Route
NORMAL	Economical route roughly calculated by average hire and bunker price
SHORTEST	Minimizing ECA distance on the route
OPTIMIZED	The most economical route with calculation total bunker expense and hire.

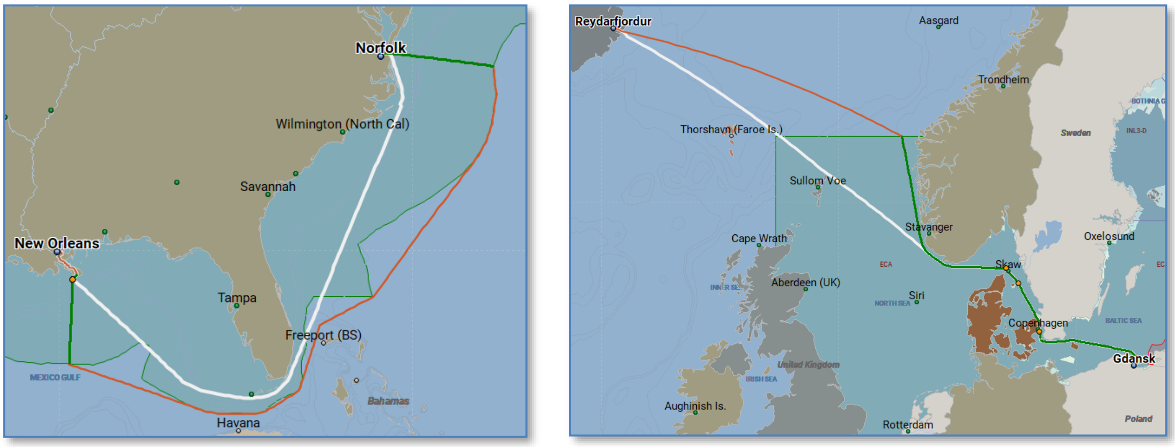
<<tableE-1: parameter option of bypass ECA >>



<<pictureE-1: Non-Bypass ECA Route >>



<<pictureE-2: Normal Bypass ECA Route>>


Norfolk
Wilmington (North Cal)
Savannah
New Orleans
Tampa
Freeport (BS)
MEXICO GULF
Bahamas
Havana
Reydarfjordur
Aasgard
Trondheim
Sweden
Thorshavn (Faroe Is.)
Sullom Voe
Stavanger
Oxelosund
Cape Wrath
ECA
Skaw
Aberdeen (UK)
NORTH SEA
Siri
Copenhagen
BALTIC SEA
Gdansk
United Kingdom
IRISH SEA
Aughinish Is.
Poland
Rotterdam

<<pictureE-3: Shortest bypass ECA Route>>

Appendix E – DEMO PORT

You can find our demo ports list table here below.

EVALUATION PORTS

PTYPE : P - PORT , A - ALIAS, R - ROUTING POINT

PORT NAME	UN-LOCODE	NP-POCODE	COUNTRY	PTYPE
Aabenraa	DKAAB	DKAABA	Denmark	A
Aahus	SEAHU	SEAHUS	Sweden	A
Aasgard		NOAASD	Norway	P
Aggersund	DKASH	DKASHD	Denmark	P
Agnefest	NOAFT	NOAFTT	Norway	P
Aioi	JPAIO	JPAIOI	Japan	P
Ajos	FIAJO	FIAJOS	Finland	P
Akinum	PGAKI	PGAKIM	Papua New Guinea	P
Akitsu	JPAKT	JPAKTU	Japan	P
Al Hoceima	MAAHU	MAAHUA	Morocco	P
Alborg	DKAAL	DKAALG	Denmark	P
Alemoa Terminal		BRALTL	Brazil	P
Algiers	DZALG	DZALGS	Algeria	P
Amio	PGAMI	PGAMIO	Papua New Guinea	P
Ancona	ITAOI	ITAOIA	Italy	P
Antalaha	MGANM	MGANMA	Madagascar	P
Antofagasta	CLANF	CLANFA	Chile	P
Arco Ardjuna		IDARAA	Indonesia	P
Arica	CLARI	CLARIA	Chile	P
Arun Terminal	IDAUN	IDAUNL	Indonesia	P
Ashdod	ILASH	ILASHD	Israel	P
Atyrau		KZATRU	Kazakhstan	P
Babouillat		NCBABT	New Caledonia	P
Bade	IDBXD	IDBADE	Indonesia	P
Bahrain	BHBAH	BHBAHN	Bahrain	P
Baku	AZBAK	AZBAKU	Azerbaijan	P
Bandar Khomeini	IRBKM	IRBKMI	Iran	P
Barahona	DOBRX	DOBRXA	Dominican Republic	P
Barranquilla	COBAQ	COBAQA	Colombia	P
Big Creek	BZBGK	BZBGKK	Belize	P
Bizerte	TNBIZ	TNBIZE	Tunisia	P

Bleckede	DEBKD	DEBLEE	Germany	P
Boizenburg	DEBOI	DEBOIG	Germany	P
Buenaventura	COBUN	COBUNA	Colombia	P
Busan	KRPUS	KRPUSN	South Korea	P
Butinge Terminal	LTBOT	LTBOTL	Lithuania	P
Cape Good Hope (RP)				R
Cape Horn (RP)				R
Cape Londonderry		AUCALY	Australia - W.A	P
Casablanca	MACAS	MACASA	Morocco	P
Ceiba Terminal		GQCETL	Equatorial Guinea	P
Chalna	BDCHL	BDCHLA	Bangladesh	P
Chilung	TWKEL	TWKELG	Taiwan	P
Conakry	GNCKY	GNCKYY	Guinea	P
Corfu Is.		GRCOIS	Greece	P
Corinth	GRCRG	GRCRGH	Greece	P
Cotonou	BJCOO	BJCOOU	Benin	P
Curacao	ANCUR	ANCURO	Netherlands Antil	P
Dampier	AUDAM	AUDAMR	Australia - W.A	P
Den Helder	NLDHR	NLDHRR	Netherlands	P
Dhekelia	CYDHK	CYDHKA	Cyprus	P
Dholera	INDHR	INDHRA	India	P
Diamond Harbour		INDIHR	India	P
Dighi	INDIG	INDIGI	India	P
Djibouti	DJJIB	DJJIBI	Djibouti	P
Dominica Pass		DMDOPS	Dominican Republic	P
Drapetzona	GRDPA	GRDPAA	Greece	P
Drummond		USDRUD	U.S.A - Michigan	P
Elang Terminal		AUELTL	Australia - W.A	P
Elat	ILETH	ILETHT	Israel	P
Elephant Point (BU)		MMELPT	Myanmar	P
Ennore	INENR	INENRE	India	P
Esperance	AUEPR	AUEPRE	Australia - W.A	P
Etame Terminal	GAETA	GAETTL	Gabon	P
Everton		GYEVEN	Guyana	P
Fauji Oil Terminal		PKFAOT	Pakistan	P
Fond Mombin	HTFOM	HTFOMN	Haiti	P
Fort De France	MQFDF	MQFDFE	France - Martinique	P
Freeport (BM)	BMFPT	BMFPTT	Bermuda	P
Fremantle	AUFRE	AUFREE	Australia - W.A	P

Gamba Terminal		GAGATL	Gabon	P
Gandhar	INGAN	INGANR	India	P
Geelong	AUGEX	AUGEXG	Australia - Victo	P
Georgetown (GY)	GYGEO	GYGEON	Guyana	P
Gijon	ESGIJ	ESGIJN	Spain	P
Goa	INGOI	INGOIA	India	P
Gresik	IDGRE	IDGREK	Indonesia	P
Guayaquil	ECGYE	ECGYEL	Ecuador	P
Haifa	ILHFA	ILHFAA	Israel	P
Haiphong	VNHPH	VNHPHG	Vietnam	P
Halong		VNHALG	Vietnam	P
Hastings (AU)	AUHAS	AUHASS	Australia - Victo	P
Hoquiam	USHQM	USHQMM	U.S.A - Washington	P
Hunafloi		ISHUNI	Iceland	P
Hvammstangi	ISHVM	ISHVMI	Iceland	P
Jakarta	IDJKT	IDJKTA	Indonesia	P
Kalmar	SEKLR	SEKLRR	Sweden	P
Kilindini	KEKIL	KEKILN	Kenya	P
Kingston (JM)	JMKIN	JMKINN	Jamaica	P
Kingstown	VCKTN	VCKTNN	Saint Vincent And Grenadines	P
Klaipeda	LTKLJ	LTKLJA	Lithuania	P
Koge	DKKOG	DKKOGE	Denmark	P
Kotor	CSKOT	CSKOTR	SERBIA AND MONTENEGRO	P
Kowloon	HKKWN	HKKWNN	China Hongkong	P
Kuwait	KWKWI	KWKWIT	Kuwait	P
Kwinana	AUKWI	AUKWIA	Australia - W.A	P
Ky Ha	VNKYH	VNKYHA	Vietnam	P
La Baule Escoublac	FROBC	FRLABE	France	P
La Mailleraye		FRLAME	France	P
La Union	SVLUN	SVLUNN	El Salvador	P
Lamma Is	HKLAM	HKLAIS	China Hongkong	P
Lankaran	AZLAN	AZLANN	Azerbaijan	P
Libreville	GALBV	GALBVE	Gabon	P
Los Angeles	USLAX	USLAXS	U.S.A - California	P
Magellan Strait (RP)				R
Malau	FJMAL	FJMALU	Fiji	P
Mandalay		MMMANY	Myanmar	P

Mariager Fjord		DKMAFD	Denmark	P
Marsa Bashayer	SDMBH	SDMBHR	Sudan	P
Matane	CAMNE	CAMNEE	Canada - Quebec	P
Matarani	PEMRI	PEMRII	Peru	P
Mentone	AUMTN	AUMENE	Australia - Victo	P
Mina Abdulla	KWMIB	KWMIBA	Kuwait	P
Mobile	USMOB	USMOBE	U.S.A - Alabama	P
Mocambique Channel		MZMOCL	Mozambique	P
Mombasa	KEMBA	KEMBAA	Kenya	P
Mont Louis	CAMOL	CAMOLS	Canada - Quebec	P
Moorea		FRMOOA	French Polynesia	P
Mori (SB)		SBMORI	Solomon Island	P
Nabq		EGNABQ	Egypt	P
Narsarsuaq	GLUAK	GLUAKQ	Denmark Greenland	P
New Richmond	CANRC	CANRCD	Canada - Quebec	P
Norfolk	USORF	USORFK	U.S.A - Virginia	P
Nouakchott	MRNKC	MRNKCT	Mauritania	P
Nuuk	GLGOH	GLGOHK	Denmark Greenland	P
Nuweiba	EGNUW	EGNUWA	Egypt	P
Ocean Cay	BSOCE	BSOCEY	Bahamas	P
Off Bahrain		BHOFBN	Bahrain	P
Opotiki	NZBPK	NZOPOI	New Zealand	P
Orinoco River		VEORRR	Venezuela	P
Otranto	ITOTO	ITOTOO	Italy	P
Palau	ITPAU	ITPAUU	Italy	P
Pamatacual	VEPAM	VEPAML	Venezuela	P
Panama Canal (RP)				R
Papeete	PFPPT	PFPPTE	French Polynesia	P
Paramaribo	SRPBM	SRPBMO	Suriname	P
Pasajes	ESPAS	ESPASS	Spain	P
Pedra De Lume		CVPEDL	Cape Verde	P
Port Latta	AUPLA	AUPLAA	Australia - Tasma	P
Port Louis (MU)	MUPLU	MUPLUS	Mauritius	P
Port Said	EGPSD	EGPSDD	Egypt	P
Port Said East		EGPOSE	Egypt	P
Port of Skulte	LVSKU	LVSKUE	Latvia	P
Porto Cervo	ITPCE	DMPORH	Italy	P
Portsmouth (DO)	DMPOR	DMPORH	Dominica	P
Portsmouth (Virginia)	USPTM	USPTMH	U.S.A - Virginia	P

Providence Channel		BSPRCL	Bahamas	P
Puerto Barrios	GTPBR	GTPBRS	Guatemala	P
Puerto Patache	CLPAT	CLPATE	Chile	P
Punta Gorda (BE)		BZPUGA	Belize	P
Ras Shukheir	EGRSH	EGRSHR	Egypt	P
Risdon Vale	AURDN	AURDNE	Australia - Tasma	P
Roomassaar	EEROO	EEROOR	Estonia	P
Rota (MP)		MPROTA	Northern Mariana	P
Rybinsk	RURYB	RURYBK	Russia	P
Sainte Croix		CASACX	Canada - Quebec	P
Samara	RUKUF	RUKUFA	Russia	P
San Fernando (TT)	TTSFE	TTSFEO	Trinidad & Tobago	P
San Jose (GT)	GTSNJ	GTSNJE	Guatemala	P
Sande		NOSANN	Norway	P
Sangar		RUSANR	Russia	P
Sawakin	SDSWA	SDSWAN	Sudan	P
Sepangar Bay		MYSEBY	Malaysia	P
Sept Iles	CASEI	CASEIS	Canada - Quebec	P
Serpiente Bay		TTSEBY	Trinidad & Tobago	P
Sevastopol	UASVP	UASVPL	Ukraine	P
Shanghai	CNSHA	CNSHAI	China	P
Siam Seaport	THSBP	THSBPT	Thailand	P
Sirte	LYSRT	LYSRTE	Libya	P
Skopje		MKSKOE	F.Y.R.O.M. (Macedonia)	P
Sokcho	KRSHO	KRSHOO	South Korea	P
St. Lucia Channel		LCSTLC	St. Lucia	P
St. Romuald	CASRO	CASROD	Canada - Quebec	P
Suez Canal (RP)				R
Swansea (AU)		AUSWAA	Australia - Tasma	P
Taboguilla Island	PATBG	PATBGD	Panama	P
Taboguilla Terminal		PATATL	Panama	P
Taean	KRTAN	KRTANN	South Korea	P
Thursday Island	AUTIS	AUTISD	Australia Queensl	P
Tripoli (LB)	LBKYE	LBKYEI	Lebanon	P
Tripoli (LY)	LYTIP	LYTIPI	Libya	P
Uranggan	AUURN	AUURNN	Australia Queensl	P
Ventspils	LVVNT	LVVNTS	Latvia	P
Veracruz	MXVER	MXVERZ	Mexico	P
Vientiane		LAVIEE	Laos	P

Vilvoorde	BEVIL	BEVILE	Belgium	P
Virtsu	EEVIR	EEVIRU	Estonia	P
Vlore	ALVOA	ALVOAE	Albania	P
Vuda Point	FJVUD	FJVUDT	Fiji	P
Wasum		PGWASM	Papua New Guinea	P
Whitegate	IEWHI	IEWHIE	Ireland	P
Yainville	FRYAV	FRYAIE	France	P
Yithion	GRGYT	GRGYTN	Greece	P
Yucatan Channel		MXYUCL	Mexico	P
Zahrani Terminal	LBZHR	LBZHRL	Lebanon	P
Zawia Terminal		LYZATL	Libya	P
Zhenzhou	CNYIZ	CNYIZU	China	P
Zhongshan	CNZSN	CNZSNN	China	P