

## 1. BridgeWeb

(revisione del 11/12/2023)

### 1.1. API di BridgeWeb

L'API di BridgeWeb è un'applicazione RESTful che prevede le possibilità di *creare risorse* e di *ottenere informazioni su risorse già create*.

Base URL: <https://services.databridge.it/api/v1/public>

### 1.2. Effettuare una chiamata

L'identificazione dell'utente avviene tramite il valore dell'header "*client-code*" (valore comunicato). Una richiesta HTTP POST permette di creare un nuovo record mentre una richiesta GET permette di ottenere informazioni su un record precedentemente creato.

### 1.3. Sicurezza e Chiavi API

L'applicazione è sotto sicurezza di tipologia "*Bearer token*". Il valore del token e la sua validità temporale sono decisi e comunicati direttamente dal cliente utilizzatore di OneWay.

### 1.4. Formato dati trasmesso verso BridgeWeb

Il formato accettato dall'applicativo è "*json*" e deve essere comunicato tramite l'header "*content-type*". I valori validi sono:

- application/json

### 1.5. Formato dati restituiti da BridgeWeb

I formati restituiti dall'applicativo sono multipli e sono:

- application/json
- application/octet-stream

### 1.6. Limitazione API

Tutte le API pubbliche sono limitate a 1000 chiamate ogni 5 minuti per singolo IP. In caso di superamento di tale soglia, ritornerà un errore con http status 429 (Too Many Requests).

Tra gli headers di questa risposta troverete "*X-Rate-Limit-Retry-After-Seconds*" che indicherà i secondi mancanti prima di poter effettuare nuove chiamate API.

## 2. Spedizione GET

Restituisce informazioni riguardo a una spedizione già inserita a sistema.

### 2.1. Struttura

Esempio di chiamata: <https://services.databridge.it/api/v1/public/spedizione/{id}>

### 2.2. Parametri GET

Campo	Tipo	Obbligatorio	Descrizione
id	integer	Si	ID della spedizione restituita dalla chiamata di spedizione POST

### 2.3. Risposta

Campo	Tipo	Descrizione
id	integer	Identificativo dell'applicativo BridgeWeb
includiRitiro	integer	Valori: 1 se è incluso il servizio di ritiro, 0 altrimenti
includiConsegna	integer	Valori: 1 se è incluso il servizio di consegna, 0 altrimenti
importoCod	double	Valore del vincolo di consegna. Es: 43.05
codValutaCod	string	Codice della valuta del vincolo di consegna. Es: "EUR"
dataRitiro	string	Data di ritiro previsto (formato yyyy-MM-dd)
notaCliente	string	Nota del cliente
nomeOperatore	string	Nome dell'operatore che ha inserito la spedizione
telefonoOperatore	string	Telefono dell'operatore che ha inserito la spedizione
numeroRitiroOW	string	Numero della pratica nel sistema OneWay
riferimentoEsterno	string	Riferimento del cliente
annullato	integer	Identifica una spedizione annullata
oraDa	string	Orario di ritiro previsto iniziale. Formato (H:mm)
oraA	string	Orario di ritiro previsto finale. Formato (H:mm)
codTipoServizio	string	Codice tipo servizio richiesto

codIncotermCli	string	Codice incoterm richiesto
codTipoCod	string	Codice del tipo del vincolo di consegna
trackingStatus	integer	Valore dello stato della spedizione. Vedi ValueList
trackingLink	string	URL del tracking del corriere
spedRifCorrispondente	string	Riferimento di consegna
codFornTassativo	string	Codice del fornitore tassativo di consegna
elementiAlfanumerici	List<Servizio>	Servizi supplementari richiesti
elementiNumerici	List<Servizio>	Servizi supplementari richiesti
singoli	List<Singolo>	Singoli merce e relative misure
luogoRitiro	Indirizzo	Informazione del luogo di ritiro
luogoConsegna	Indirizzo	Informazioni del luogo di consegna
riferimenti	List<Riferimento>	Riferimenti aggiuntivi

#### 2.4. Strutture dati compelsse

**Struttura “Singolo”:** {colli:integer, pezzi:integer, kgNetti: double, kgLordi: double, mCubiReali: double, mLineari: double, codTipoCollo: string, codTipoPezzo: string, valoreMerce: double, valutaMerce: string, valoreDogonale: double, valutaDogonale: string, misure: List<Misura>, descMerce: string, nonSovrapponibile:Boolean}  
 “codTipoCollo”: ValueList comunicata dal cliente.

**Struttura “Misura”:** {colli: integer, altezza: integer, larghezza: integer, profondita: integer}

**Struttura “Indirizzo”:** {ragioneSociale: string, indirizzo: string, cap: string, nazione: string, provincia: string, citta: string, telefono: string, email: string, contatto: string}  
 “nazione”: formato ISO 3166-1 alpha-2

“provincia”: per destinazioni estere può essere usata per specificare lo stato (es: nazione: “US”, provincia “CA”)

**Struttura “Servizio”:** { cod\_variabile:string, valore:string}

**Struttura “Riferimento”:** {codTipoRiferimento: string, riferimento: string}

## 2.5. HTTP status

200: ok  
400: Richiesta mal formulata  
404: Risorsa non trovata  
500: Errore generico

## 2.6. Tracking

È possibile ottenere le informazioni sullo status della spedizione consultando il campo "trackingStatus"

I valori possono essere:

- 0: In attesa di convalida da parte di un operatore
- 1: Confermata
- 2: Arrivata al terminal
- 3: In viaggio
- 4: In consegna
- 5: Consegnata

Per alcune spedizioni troverete compilato anche il campo "trackingLink" con un link al tracking del Corriere.

Attenzione questo link sarà disponibile dallo status 3 in poi

### 3. Spedizione POST

Chiamata usata per la creazione di una spedizione (e di un ritiro) nel sistema OneWay

#### 3.1. Struttura

Esempio di chiamata: <https://services.databridge.it/api/v1/public/spedizione> (HttpMethod.POST)

#### 3.2. Parametri Spedizione POST

Il body della request dovrà avere la seguente struttura

Campo	Tipo	Obbligatorio	Descrizione
includiRitiro	integer	Si	Valore: 1 per includere il servizio di ritiro, 0 se la merce viene consegnata dal cliente
includiConsegna	integer	Si	Valore: 1 per includere il servizio di consegna, 0 se la merce viene ritirata dal cliente
importoCod	double	No*	Valore del vincolo di consegna. Es: 43.05
codValutaCod	string	No*	Codice della valuta del vincolo di consegna. Es: "EUR". ValueList comunicata separatamente.
dataRitiro	string	Si	Data di ritiro previsto (formato yyyy-MM-dd)
notaCliente	string	No	Nota del cliente
nomeOperatore	string	Si	Nome dell'operatore che ha inserito la spedizione
telefonoOperatore	string	Si	Telefono dell'operatore che ha inserito la spedizione
riferimentoEsterno	string	No	Riferimento del cliente
oraDa	string	Si	Orario di ritiro previsto iniziale (H:mm). Minuti disponibili 00/15/30/45
oraA	string	Si	Orario di ritiro previsto finale (H:mm). Minuti disponibili 00/15/30/45

codTipoServizio	string	Si	Codice tipo servizio richiesto. ValueList comunicata separatamente.
codIncotermCli	string	Si	Codice incoterm richiesto. ValueList comunicata separatamente.
codTipoCod	string	No*	Codice del tipo del vincolo di consegna. ValueList comunicata separatamente.
codFornTassativo	string	No	Codice del fornitore tassativo di consegna. ValueList comunicata separatamente.
elementiAlfanumerici	List<Servizio>	No	Servizi supplementari richiesti. ValueList comunicata separatamente.
elementiNumerici	List<Servizio>	No	Servizi supplementari richiesti. ValueList comunicata separatamente.
singoli	List<Singolo>	Si	Singoli merce e relative misure
luogoRitiro	Indirizzo	Si	Informazione del luogo di ritiro
luogoConsegna	Indirizzo	Si	Informazioni del luogo di consegna
Riferimenti	List<Riferimento>	No	Riferimenti aggiuntivi

Obbligatorietà "No\*": Diventa obbligatorio se uno dei campi relativi al valore merce è compilato

### 3.3. Strutture dati complesse

**Struttura "Singolo":** {colli:integer, kgNetti: double, kgLordi: double, mCubiReali: double, mLineari: double, codTipoCollo: string, valoreMerce: double, valutaMerce: string, valoreDogonale: double, valutaDogonale: string, misure: List<Misura>, descMerce: string, nonSovrapponibile:Boolean}  
 "codTipoCollo": ValueList comunicata dal cliente.

**Struttura "Misura":** {colli: integer, altezza: integer, larghezza: integer, profondita: integer}

**Struttura "Servizio":** { cod\_variabibile: string, valore: string}

**Struttura "Riferimento":** { codTipoRiferimento: string, riferimento: string}

**Struttura “Indirizzo”:** {ragioneSociale: string, indirizzo: string, cap: string, nazione: string, provincia: string, citta: string, telefono: string, email: string, contatto: string}  
 “nazione”: formato ISO 3166-1 alpha-2  
 “provincia”: per destinazioni estere può essere usata per specificare lo stato (es: nazione: “US”, provincia “CA”)

### 3.4. *Obbligatorietà strutture complesse*

**Campi obbligatori per “Singolo”:** colli, kgLordi, codTipoCollo, descMerce.  
 Se presente l’elemento alfanumerico “yn\_assicurazione\_s” deve essere specificato valore e valuta della merce.

Se il singolo è marchiato come “non sovrapponibile” le misure del singolo merce sono obbligatorie.

Se la spedizione è estera ed il tipo di colli non è “documento” è necessario specificare valore e valuta doganale.

**Campi obbligatori per “Misura”:** Misura è opzionale, ma se inserito nella chiamata va compilato in tutte le sue parti. Attenzione, la somma di “colli” delle varie misure deve coincidere con il numero di colli presente in “Singolo”.

Attenzione se il singolo è “non sovrapponibile” le misure sono obbligatorie.

**Campi obbligatori in “Indirizzo”:** Tutti i campi sono obbligatori

**Campi obbligatori in “Servizio”:** Tutti i campi sono obbligatori. Si consiglia di specificare solo gli elementi presenti (valore deve essere “Y” se presente, “N” se non presente)

**Campi obbligatori in “Riferimento”:** Tutti i campi sono obbligatori.

### 3.5. *Risposta*

La “spedizione” POST ritorna esattamente il risultato di una chiamata “spedizione” GET se la spedizione è stata creata correttamente nei nostri sistemi (http status 200).

In caso di errore ritorna un oggetto con la seguente **struttura**: { msgErrore: string } dove potrete trovare maggiori dettagli.

### 3.6. *HTTP status*

200: ok

400: Richiesta mal formata (controlla “msgErrore” per eventuali messaggi più specifici)

500: Errore generico (controlla “msgErrore” per eventuali messaggi più specifici)

### 3.7. *Note generali*

La strutta dati “Singolo” rappresenta un singolo merce. È obbligatorio definire almeno un singolo merce per spedizione

La strutta dati “Servizio” rappresenta dei servizi aggiuntivi e disponibili per la spedizione come “consegna al piano”, “consegna al sabato” etc.

## 4. Documentale GET

Questa API vi sarà utile per fare il download di alcuni documenti, quali etichette segnacolli, resume di spedizione e prefattura sostitutiva

### 4.1. Struttura

Esempio di chiamata:

`https://services.databridge.it/api/v1/public/documentale/{tipo_documento}/{id}(HttpMethod.GET)`

### 4.2. Parametri GET

Campo	Tipo	Obbligatorio	Descrizione
tipo_documento	string	Si	Tipologia richiesta. Dettaglio in calce
id	integer	Si	ID della spedizione restituito dalla chiamata di spedizione POST

### 4.3. Tipologie documento

- etichette-ritiro: Per un documento (pdf) contenente etichette segna-colli
- sintetico-spedizione: Per un documento riassuntivo della spedizione commissionata
- fattura-proforma: Per un documento utile in dogana

Di default questi documenti vengono inviati come stream di byte. Opzionalmente è possibile ottenerli in base64, specificando un parametro “encoding” con valore “base64” (query parameter).

### 4.4. Risposta

La risposta sarà sempre un Array<byte> e rappresenterà il documento scelto.

È presente l’header “file-name” che contiene il nome del documento, ma potete nominarlo come preferite.

### 4.5. Disponibilità dei documenti

Non tutti i documenti sono sempre disponibili. Essi dipendono dallo stato in cui si trova la spedizione.

Con valori di “trackingStatus” maggiori o uguali a 1 (da confermata in poi) sono disponibili tutti i documenti.

Con valore di “trackingStatus” uguale a 0 (in attesa di conferma) non è possibile stampare le etichette di ritiro, mentre gli altri documenti, anche se potrebbero essere modificati in fase di convalida, sono validi e disponibili.



## 5. Spedizione DELETE

Chiamata per annullare una spedizione che è stata affidata. È possibile annullare una spedizione che si trova sia in stato di Attesa, sia in stato di Confermata a patto che un operatore non la stia lavorando in quel momento, o che lo stato della spedizione non ne permetta l'annullo.

### 5.1. Struttura

Esempio di chiamata: <https://services.databridge.it/api/v1/public/spedizione/{id}> (HttpMethod.DELETE)

### 5.2. Parametri

Campo	Tipo	Obbligatorio	Descrizione
id	integer	Si	ID delle spedizione restituito dalla chiamata di spedizione POST

### 5.3. Risposta

La chiamata può restituire le seguenti risposte. Se positive il body non è presente, se negative nel body è possibile trovare un oggetto contenente la proprietà "msgErrore" nella quale trovare maggiori dettagli.

Ecco un elenco di http status:

- 200: OK. La spedizione è stata cancellata con successo
- 404: NOT FOUND. La spedizione non è stata trovata
- 409: CONFLICT. Lo stato della spedizione non consente l'annullo
- 423: LOCKED. La spedizione è in gestione da un operatore
- 420: METHOD FAILURE. In questo caso è consigliabile riprovare e/o contattare l'assistenza
- 400: BAD REQUEST. La richiesta non è corretta

## 6. SchedulesList GET

BridgeWeb mette a disposizione questa API pubblica per ottenere la lista delle tratte disponibili. Questa chiamata API non prevede autenticazione tramite “Bearer token” ma necessita solamente dell’header “client-code”.

Se viene richiesta una specifica pagina, il set di dati è di 20 tratte per pagina. In caso contrario viene ritornato l’intero set di dati.

### 6.1. Struttura

End point: <https://services.databridge.it/api/v1/public/schedulesList> (HttpMethod.GET)

### 6.2. Query parameters

Per filtrare l’elenco delle “Tratte”, sono stati messi a disposizione i seguenti parametri (query parameters).

Campo	Tipo	Obbligatorio	Descrizione
portoPartenza	String	No	Codice porto partenza. Es: GOA,SHA,NYC
portoArrivo	String	No	Codice porto arrivo. Es: GOA,SHA,NYC
portoIntermedio	String	No	Codice porto di transito. Vuoto se diretto. Es: GOA,SHA,NYC
dataCutOffDa	String	No	Formattazione: dd/MM/yyyy
dataCutOffA	String	No	Formattazione: dd/MM/yyyy
dataPartenzaDa	String	No	Formattazione: dd/MM/yyyy
dataPartenzaA	String	No	Formattazione: dd/MM/yyyy
dataArrivoDa	String	No	Formattazione: dd/MM/yyyy
dataArrivoA	String	No	Formattazione: dd/MM/yyyy
includiDirettiEProselezioni	String	No	“Y” se la ricerca deve includere sia tratte dirette che tratte con proselezioni
pageNumber	Integer	No	

### 6.3. Risposta

Ecco un elenco di http status:

- 200: OK. La ricerca è stata eseguita con successo
- 400: BAD REQUEST. La richiesta non è corretta

### 6.4. Struttura risposta

La risposta è un oggetto json contenente le seguenti proprietà:

- schedulerNavi: Array<Tratte>
- requestPage: Integer (Se è stata richiesta una pagina specifica, ritorna il numero di pagina)
- totalPages: Integer (Il numero totale di pagine per la ricerca effettuata)

Per entità "Tratta" si intende un oggetto con le seguenti proprietà:

```
{
  "codPortoPartenza": String,
  "nazPortoPartenza": String,
  "nomePortoPartenza": String,
  "codPortoArrivo": String,
  "nazPortoArrivo": String,
  "nomePortoArrivo": String,
  "codPortoIntermedio": String,
  "nazPortoIntermedio": String,
  "nomePortoIntermedio": String,
  "nave": String,
  "numeroViaggio": String,
  "dataCutOff": String (dd/MM/yyyy),
  "dataPartenzaPrevista": String (dd/MM/yyyy),
  "dataArrivoPrevista": String (dd/MM/yyyy),
  "travelTime": Integer,
  "stato": String
}
```

## 7. PortsList GET

BridgeWeb mette a disposizione questa API pubblica per ottenere la lista dei possibili porti di partenza/arrivo/transito disponibili a sistema.

Questa chiamata API non prevede autenticazione tramite “Bearer token” ma necessita solamente dell’header “client-code”.

È possibile filtrare i porti che vengono utilizzati per i traffici import o export semplicemente aggiungendo un path parameter a fine chiamata (“/export”, “/import”).

### 7.1. Struttura

End point: <https://services.databridge.it/api/v1/public/portsList> (HttpMethod.GET)

End point porti export: <https://services.databridge.it/api/v1/public/portsList/export> (HttpMethod.GET)

End point porti import: <https://services.databridge.it/api/v1/public/portsList/import> (HttpMethod.GET)

### 7.2. Risposta

Ecco un elenco di http status:

- 200: OK. La ricerca è stata eseguita con successo
- 400: BAD REQUEST. La richiesta non è corretta

### 7.3. Struttura risposta

La risposta è un oggetto json contenente le seguenti proprietà:

- porti: Array<Porto>

L’entità “Porto” ha la seguente struttura:

```
{
  "codice": String,
  "nome": String,
  "nazione": String,
  "codNome": String
}
```