

関係者外秘

B2クラウド APIデータ交換規約 (法人企業版)

B2クラウド API

版数: 0.4.0

2024/5/9

ヤマト運輸株式会社

変更履歴

	版数	年月日	変更内容
1	0.1.0	2016/4/18	1 新規作成 API連携利用概要図(画面なし)シート作成 API連携利用概要図(画面あり)シート作成 API連携利用概要図(画面なし)シート修正
2	0.1.1	2016/8/19	2 印刷データ受け取りからPDFダウンロードに修正
3	0.1.2	2016/9/15	3 API連携利用概要図(画面API利用なし)シート修正 API連携利用概要図(画面API利用)シート修正 API認証トークン取得方法シート修正 APIインターフェースシート修正 印刷状態確認処理サンプルシート修正 印刷ダイアログ表示方法サンプルシート修正 出荷データ仕様シート修正 出荷データサンプル(JSON)シート修正 制限事項シート修正
4	0.1.3	2016/9/21	1 「はじめに」シート追加 「1. インターフェイス概要」シート修正 「2. ユーザ操作フロー」シート追加 「3. API連携利用概要図(画面API利用)」修正 「4. API連携利用概要図(画面API利用なし)」修正
5	0.1.4	2016/9/29	1 「10. 制限事項」修正
6	0.1.5	2016/10/6	1 「3. API連携利用概要図(画面API利用)」修正 2 「2. ユーザ操作フロー」API認証トークン⇒APIアクセス認証キー 3 「5. 認証について」API認証トークン⇒APIアクセス認証キー 4 「5-1 APIアクセス認証キー取得方法」 API認証トークン⇒APIアクセス認証キー 5 「3. API連携利用概要図(画面API利用)」修正 図を詳細に修正
			6 「4. API連携利用概要図(画面API利用なし)」修正 PUT /b2/p/new?issue_edit&print_type=m&api_user_id=XXX ⇒POST /b2/p/new?issue_editA&print_type=m&api_user_id=XXX
			7 「7-1. APIインターフェース」 ■送り状発行 を下記のように修正 PUT⇒POST /b2/p/new?issue_edit&api_user_id=xxx ⇒ /b2/p/new?issue_editA&api_user_id=xxx ■仮データ登録・データチェックにAPIアクセス認証キー項目追加 全体 rxid ⇒ RXID に修正
7	0.1.6	2016/11/10	1 「7-1. APIインターフェース」
		2016/11/25	1 「7-1. APIインターフェース」 ・仮データ登録・データチェックのリクエストURLを修正 ・取込み結果表示画面のリクエストURLを修正 ・出荷データ取得のリクエストURLを修正 ・CookieをリクエストヘッダからURLパラメータへ移動 「3. API連携利用概要図(画面API利用)」 ・APIインターフェースに合わせて各リクエストURLを変更 「4. API連携利用概要図(画面API利用なし)」 ・APIインターフェースに合わせて各リクエストURLを変更 「2. ユーザ操作フロー」 ・前提条件の内容を変更
9	0.1.8	2016/12/27	1 「4. API連携 利用概要図(画面API利用なし)」 「7-1. APIインターフェース」 の一部変更 ■出荷データ取得
10	0.1.9	2017/1/27	1 「7-1. APIインターフェース」 ■出荷データ取得 リクエストボディ ・送り状発行リクエストのリクエストデータ例を修正
10	0.2.0	2017/1/27	1 3. API連携利用概要図(画面API利用)、4. API連携利用概要図(画面API利用なし) ■POST /b2/p/editA?api_user_id=XXXCDの戻りの修正
		2017/1/30	1 「7-1. APIインターフェース」 ■仮データ取得(GET) ■出荷データ取得 リクエストヘッダ ・「X-Requested-With」の説明を削除
12	0.2.2	2017/2/3	1 「3.3 API連携利用概要図(画面API利用)」 HTTPヘッダのRXID設定箇所の修正 7-1. APIインターフェース ■仮データ登録・データチェックのレスポンスボディにRXIDを設定する旨を追記。(もともとリクエストボディに記載があった為)
13	0.2.3	2017/3/6	1 3. API連携利用概要図(画面API利用) P10 ・取込み結果画面のURL修正 (誤) /p/b2/ /html/importapi.html (正) /b2/p/ /html/mult_import_api.html ・印刷状態確認時のレスポンス結果を追加 Http Status: 200(OK) Http Body: RXID=xxx2 ・印刷データ取得レスポンスにRXIDをCookieに設定しているイメージの削除 ・伝票番号取得時のレスポンス結果にRXIDを追加 Http Body: (1)RXID=xxx3 P11 ・画面有り接続URL一覧の文言追加 ・2 URL修正 (誤) /p/b2/ /html/importapi.html (正) /b2/p/ /html/mult_import_api.html ・3 説明追加 伝票番号取得用お客様システムの通知URL
			2 4. API連携利用概要図(画面API利用なし) P13 ・印刷状態確認「202 作成中」ステータス追加 説明修正 (誤) PDFダウンロード処理時にRXIDを設定する (正) 伝票番号取得処理のRXIDのパラメータ値として利用する。 ・PDFダウンロード URLからRXIDを削除 (誤) &RXID=xxxxx ・伝票番号取得 レスポンス結果CookieにRXID設定 ・緊急時出荷登録APIを追加
			3 6. 通信プロトコル JSON形式の例を修正
			4 7. 利用可能API一覧 ■画面あり ・API名を修正 (誤) 印刷データ取得 (正) 伝票番号取得 ・APIの説明を修正 (正) 出荷データの印刷を実行する (正) 出荷データを取得する。 ■画面なし ・下記のAPIを追加 - 伝票番号取得 - PDFダウンロード - 緊急時出荷登録

変更履歴

	版数	年月日	変更内容
			4 7-1 APIインターフェース 全体 リクエストヘッダに必須項目追加 ■仮データ登録・データチェック ・レスポンスボディの項目形式を追加 ■仮データ更新・データチェック ・パラメータ名修正 (誤) multi (正) multi ■仮データ削除 説明の訂正 (誤)「仮データ削除」APIの使用後に、データを更新する場合に利用する。 (正) 仮データ登録・更新したデータを一部削除する場合に用いる。 ・メソッドの訂正 (誤) DELETE (正) POST ・リクエストヘッダ追加 - X-HTTP-Method-Override: DELETE /b2/p/editA ・リクエストボディの説明追加 (誤) 対象の出荷データを設定 (正) 削除対象の出荷データを設定 ・レスポンスボディ説明追加・訂正 (誤) Feed.titleにSuccess (正) Feed.titleにDeleted
			5 ■送り状発行 ・概要の説明を補足 ・パラメータ追加 display リクエストボディ 下記文言は削除 ・shipment_flgを「1」に設定してください。 ・error_flgに空文字を設定してください。 下記文言追加 ・任意項目 プリンタ種別の指定が可能です。"printer_type" ・リクエストデータの例を修正 レスポンス ・指定のプリンタで印刷できない送り状が有るの説明修正 (誤) Printer type is invalid. (正) printer type or print_type is invalid ・文言修正 (誤) ビジー (正) ビジー状態 ・文言訂正 (誤) キュー上限 (正) 出荷データ上限エラー ・レスポンスボディの説明追加
			6 ■印刷状態の確認 ・概要説明の補足 ・レスポンスステータス追加 作成中 ・レスポンスボディの項目説明追加 ■PDFダウンロード レスポンスの成功時の以下の説明削除 ※帳票サーバからinputstreamした受領したデータを画面側にレスポンスとしてoutputstream 印刷ダイアログを表示して印刷を行う場合は、「7-3.印刷ダイアログ表示方法サンプル」をご参照ください。 ■取込み結果表示画面 ・リクエストURL修正 (誤) /p/b2/html/import_api.html (正) /b2/p/html/mult_import_api.html ・パラメータ修正 (誤) RXID (正) RXID ■伝票番号取得 ・パラメータ修正 (誤) RXID (正) RXID ・Cookieの設定追加 ・レスポンスボディの設定追加
			7 ■緊急時出荷登録 新規作成
			8 9.出荷データ仕様 エラーフラグを追加 説明・桁数変更 ・ご依頼主・入力住所 (誤) 半角200/全角100 (正) 半角96/全角48 ・お届け先・入力住所 (誤) 半角200/全角100 (正) 半角96/全角48
			9 9-1.出荷データサンプル(JSON) ■仮データ登録・データチェックのレスポンスデータに修正
14	0.2.4	2017/3/16	1 10.7.推奨環境 最新の推奨OS、ブラウザ情報に更新 7-1 APIインターフェース ■仮データ登録・データチェック リクエストヘッダから下記の項目を削除
15	0.25	2017/5/15	1 排他制御について記載追加 シート「はじめに」 7-3.排他制御 追加 シート「7-1 APIインターフェース」 リクエストボディ、レスポンスボディにFeed.updatedの項目 追加 シート「7-3.排他制御」追加

変更履歴

	版数	年月日	変更内容
16	0.2.6	2017/5/18	<p>1 出荷データ仕様の必須見直し</p> <p>No.1 お客様管理番号 必須 ⇒ 任意</p> <p>No.3 クール区分 追記:未入力は「0:通常」固定</p> <p>No.5 お届け予定日 追記:最短期日を指定する場合、「最短期日」を設定。 「最短期日」が指定された場合、No.75最短期日フラグが「1:最短期日指定あり」になる</p> <p>No.10 請求先・分類コード 追記:それ以外の送り状種類では必須 分類コードがない場合、「1」を指定。 任意⇒必須</p> <p>No.12 請求先・ニックネーム 追記:No.9「12」に紐づく顧客情報の請求先・ニックネーム</p> <p>No.14 クロネコwebコレクト加盟店番号 追記:クロネコwebコレクトデータ登録が「1:クロネコwebコレクト」の場合、必須 ※請求先に紐づく加盟店番号</p> <p>No.15 クロネコwebコレクト申込受付番号1 改訂:クロネコwebコレクトデータ登録が「1:クロネコwebコレクト」の場合、必須</p> <p>No.27 日付印字フラグ 追記:DM便、ネコポスは0固定 ※その他の送り状種類はお届け予定日の入力により、デフォルト設定を行う お届け予定日の入力あり:「1:伝票に日付を印字する(出荷予定日、お届け予定(指定)日)」 お届け予定日の入力なし:「0:伝票に日付を印字しない」</p>
			<p>2 No.29 出荷予定個数口 追記:未入力時は「1」固定</p> <p>No.30 配達時間帯区分 追記:未入力時は「0000:指定なし」固定 ※送り状種類がタイムの場合は「0010:午前10時まで」固定</p> <p>No.31 お届け予定メール利用フラグ 追記:送り状種類:DM便、タイム、ネコポスは「0:利用しない」固定 未入力時は「0:利用しない」固定</p> <p>No.32 お届け予定メール利用フラグ</p> <p>No.34 お届け予定メッセージ</p> <p>No.36 お届け完了E-mailアドレス</p> <p>No.37 お届け完了メッセージ 追記:お届け予定メール利用フラグが「1:利用する」の場合、必須</p> <p>No.38 メール使用電話番号印字フラグ⇒DM使用電話番号印字フラグ</p> <p>No.40 ご依頼主・コード 改訂:全角桁数削除</p> <p>No.44 ご依頼主・敬称 改訂: 設定不要 DM便以外は「様」固定</p>
			<p>3 No.48 ご依頼主・市区郡町村名 追記:No.46「ご依頼主・入力住所」が未入力の場合は必須</p> <p>No.49 ご依頼主・番地 追記:No.46「ご依頼主・入力住所」が未入力の場合は必須</p> <p>No.51 ご依頼主・略称カナ 桁数変更:半角100⇒50 全角50⇒削除</p> <p>No.52 お届け先・コード 全角桁数削除:全角40⇒削除</p> <p>No.59 お届け先・市区郡町村名 追記:No.57「お届け先・入力住所」が未入力の場合は必須</p> <p>No.60 お届け先・番地 追記:No.57「お届け先・入力住所」が未入力の場合は必須</p>
			<p>4 No.64 お届け先・略称カナ 追記:半角100⇒50 全角50削除</p> <p>No.66 止置きサービス利用フラグ 追記:未入力時はデフォルト「0:利用しない」固定</p> <p>No.68 品名コード1</p> <p>No.69 品名コード2 改訂:全角60 削除</p> <p>No.70 品名称1 改訂:DM便以外は必須</p> <p>No.75 最短期日フラグ 改訂: 送り状種類:「発払い、コレクト、タイム、着払い、宅急便コンパクト、コンパクトコレクト」の場合かつ お届け予定日に「最短期日」指定がある場合、「1:最短期日指定あり」固定</p> <p>No.77 投函予定メールアドレス</p> <p>No.79 投函予定メールメッセージ 改訂:投函予定メール利用区分が「1:利用する」の場合、必須</p> <p>No.81 投函完了メール(受人宛て)アドレス</p> <p>No.83 投函完了メール(受人宛て)メッセージ 改訂:投函完了メール(受人宛て)利用区分が「1:利用する」の場合、必須</p> <p>No.85 投函完了メール(出人宛て)アドレス</p> <p>No.87 投函完了メール(出人宛て)メッセージ 改訂:投函完了メール(出人宛て)利用区分が「1:利用する」の場合、必須</p>

変更履歴

	版数	年月日	変更内容
17		2017/5/29	1 ■送り状発行 URLパラメータm&api_user_id=xxxGD MessagePack時 の指定例 以下省略) URLパラメータ m 削除 レスポンスヘッダ追加 返却される形式はJSON形式またはMessagePack形式となります。 ⇒ 返却される形式はJSON形式となります。 ■仮データ取得 URLパラメータ m 削除 レスポンスヘッダ追加 返却される形式はJSON形式またはMessagePack形式となります。 ⇒ 返却される形式はJSON形式となります。 ■仮データ更新・データチェック URLパラメータ m 削除 レスポンスヘッダ追加
			2 ■仮データ削除 URLパラメータ m 削除 レスポンスヘッダ追加 返却される形式はJSON形式またはMessagePack形式となります。 ⇒ 返却される形式はJSON形式となります。 ■緊急時出荷登録 URLパラメータ m 削除 レスポンスヘッダ追加 ■仮データ登録・データチェック URLパラメータ m 削除 レスポンスヘッダ追加
18	0.2.7	2017/6/14	1 ■印刷データチェック 新規作成 ■PDFダウンロード リクエストURL 修正 /b2/p/getfile?display=0&issue_no=xxxxx ⇒ /b2/p/getfile?display=0&issue_no=xxxxx&fileonly=1 URLパラメータ 修正 ⇒ fileonly追加 レスポンス 修正 ⇒ 発行番号誤り / 印刷データなし ⇒ 削除
		2017/6/21	2 9.出荷データ仕様 時間帯変更
		2017/8/8	1 10.制限事項 10.7推奨環境 MacOSバージョン変更
19	0.2.8	2018/1/17	シート「7-1.APIインターフェース」 汎用API(画面なし)A5マルチ印刷時のプリンタタイプを追記 汎用API(画面なし)A5マルチ印刷時のパラメータ使用例を追記
20	0.2.9	2018/11/8	シート「7-1.APIインターフェース」 汎用API(画面なし)送り状発行APIにて、 リクエストボディ内の項目を追加
21	0.3.0	2020/9/2	シート「4.API連携利用概要図(画面API利用なし)」 サーバ間連携の概要図を追加。
22	0.3.1	2021/11/15	シート「7-1.APIインターフェース」 ■仮データ削除APIの X-HTTP-Method-Overrideの設定値を修正 修正内容: 修正前: "X-HTTP-Method-Override":"DELETE /b2/p/editA" 修正後: "X-HTTP-Method-Override":"DELETE /editA" ■仮データ更新・データチェックAPIのリクエストボディに設定必須 項目を指定 修正内容: created_msをshipmentに追加必須。
23	0.3.2	2022/1/27	シート「1.インターフェース概要」 ・B2クラウドで扱える伝票種別に 発払い(複数口) を追加。 シート「9.出荷データ仕様」 ・送り状種類の説明欄に発払い(複数口)を追加 ・クール区分、個数口印字フラグ、 クロネコwebコレクト申込受付番号2、 クロネコwebコレクト申込受付番号3、複数口くりキー、 出荷予定個数口、お届け予定メール利用フラグ、 お届け完了メール利用フラグの説明欄に、 発払い(複数口)の補足説明を追加 シート「9-2.発払い(複数口)の入力チェックについて」 ・シートを追加し、発払い(複数口)の入力チェックを記載
24	0.3.3	2022/3/14	シート「9-2.発払い(複数口)の入力チェックについて」 ・営業所コードのチェックを追加
25	0.3.4	2022/11/17	シート「7-1.APIインターフェース」 ・データ更新APIのリクエストデータを追加
26	0.3.5	2023/11/22	シート「5-APIアクセス認証キー取得方法」 ・汎用API未利用時のキャプチャを追加 シート「10.制限事項」 ・推奨環境情報を最新化
27	0.3.6	2023/12/11	シート「1.インターフェース概要」、「5-APIアクセス認証キー取得方 法」、「7-1.APIインターフェース画像」、「9.出荷データ仕様」 ・クロネコゆうバケット、クロネコゆうメール対応による最新化
28	0.3.7	2023/12/26	シート「7-1.APIインターフェース画像」 ・送り状発行API 送り状発行リクエストのリクエストデータ例を更新
29	0.3.8	2024/4/1	シート「4.API連携利用概要図(画面API利用なし)」 シート「7.利用可能API一覧」 シート「7-1.APIインターフェース」 シート「7-3.辞他制御」 ・緊急時出荷登録API クラウド+提供停止に伴い緊急時出荷登録APIの記載を削除 シート「10.制限事項」 ・推奨環境 Firefoxの文言を削除 印刷ツールの説明の変更(サービス提供終了) Acrobat Readerの説明を削除

変更履歴

	版数	年月日	変更内容
30	0.3.9	2024/4/12	シート「9.出荷データ仕様」 ・お届け先電話番号 クロネコゆうメールの必須チェックなしに伴い必須項目・説明欄 修正
31	0.4.0	2024/5/9	シート「2.ユーザー操作フロー」 ・⑦PDF有効期間の登録(画面API利用なしの場合) API検証環境の場合の記載を追加 シート「3.API連携利用概要図(画面API利用)」 シート「4.API連携利用概要図(画面API利用なし)」 シート「5.認証について」 シート「5-1APIアクセス認証キー取得方法」 ・トークン API検証環境の場合の記載を追加 シート「10.制限事項」 ・送り状PDF API検証環境の場合の記載を追加 ・推奨環境 印刷ツールの記載を削除

はじめに

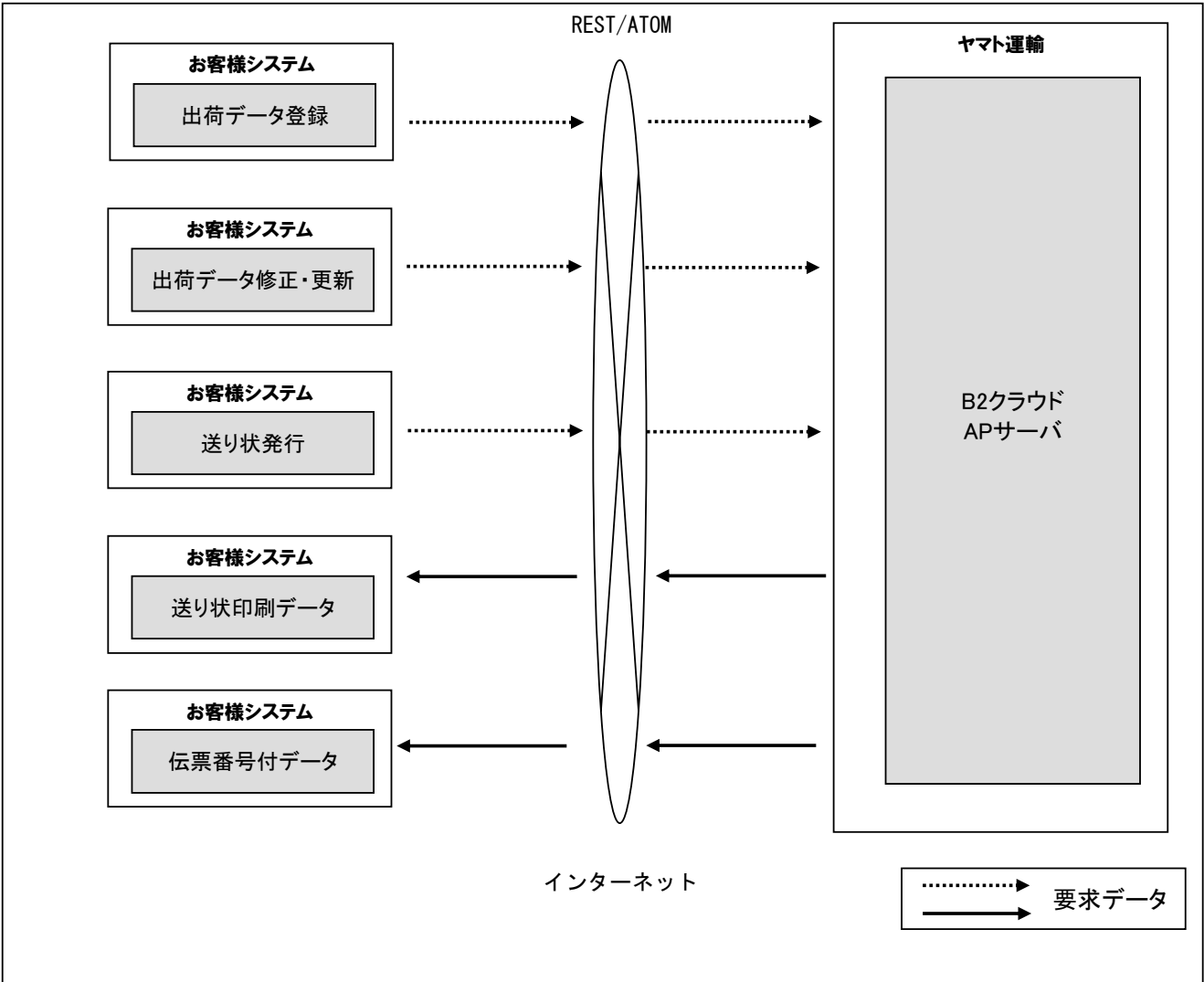
本規約は、ヤマト運輸 B2クラウドシステムと法人企業システムとのAPI連携を行う際の仕様を定義しています。

目次

1. インターフェース概要
2. ユーザ操作フロー
3. API連携利用概要図(画面API利用)
4. API連携利用概要図(画面API利用なし)
5. 認証について
 - 5-1 APIアクセス認証キー取得方法
6. 通信プロトコル
 - 6-1. 通信データ作成ライブラリの利用
 - 6-2. MessagePackの利用
7. 利用可能API一覧
 - 7-1. APIインターフェース
 - 7-2. 印刷状態確認処理サンプル
 - 7-3. 排他制御
8. HTTPステータスについて
9. 出荷データ仕様
 - 9-1. 出荷データサンプル(JSON)
 - 9-2. 発払い(複数口)の入力チェックについて
10. 制限事項
11. 免責事項

1.インターフェイス概要

当インターフェースは以下に示すようなインターネットのAPIインターフェースです。
これを利用することで法人企業のシステムからシームレスに送り状を発行することができます。



インターフェース概要図

B2クラウド システムで扱える伝票種別は以下のとおりです。

- | | | |
|------------|------------------|-----------|
| ①発払い | ②コレクト | ⑨発払い(複数口) |
| ③クロネコゆうメール | ④タイム | |
| ⑤着払い | ⑥ネコポス・クロネコゆうパケット | |
| ⑦宅急便コンパクト | ⑧コンパクトコレクト | |

また、当インターフェースでは、2つのご利用パターンを用意しております。
お客様システムに合わせてご利用ください。

・B2クラウド APIのご利用パターン

- ①B2クラウドの画面APIを利用する
お客様のシステムで入力された情報を、B2クラウドの画面で表示し、送り状の修正～送り状発行が行えます。
- ②B2クラウドの画面APIを利用しない
お客様のシステムとB2クラウド間で出荷データのやり取りを行い、送り状の修正～送り状発行が行えます。
お客様システム画面で編集や発行を行いたい場合にご利用ください。

当ガイドでは、

「①B2クラウドの画面APIを利用する」に関わる説明には、

画面API利用あり

「②B2クラウドの画面APIを利用しない」に関わる説明には、

画面API利用なし

両ご利用パターンに関わる説明には、

画面API利用あり

画面API利用なし

とアイコンを設定しています。

2.ユーザ操作フロー

画面API利用あり

画面API利用なし

前提条件:

- ①ヤマト運輸との未収取引契約を結んでいること
- ②ヤマトビジネスメンバーズの本登録を済ませていること
- ③APIアクセス認証キーは、ヤマトビジネスメンバーズ管理者ユーザー（個人ユーザーIDなし）のみ確認ができる。
- ④API連携会社コード
B2クラウドシステム担当より、API連携会社コードが発行されます。（お客様契約時）
API連携会社コードは、お客様の1つのシステムに対して1つのコードが割り当てられます。
API連携会社コードは、APIのURLパラメータ「api_user_id」に設定する必要があります。
- ⑤許可ドメイン登録
B2クラウドシステムにアクセスするお客様ドメインを登録。（お客様契約時）
クロスサイトリクエストになるため、お客様システムのドメインを登録する必要があります。
- ⑥通知先URL登録（画面API利用ありの場合）
B2クラウドシステムからお客様のシステムへ印刷結果を通知するURLを登録。（お客様契約時）
- ⑦PDF有効期間の登録（画面API利用なしの場合）
ダウンロードしたPDFファイルの有効時間を登録（時間単位）。（お客様契約時）
※API検証環境をご利用の場合、発行される伝票はサンプルPDFのため有効時間はございません。

No	ユーザー操作	メニュー（画面）	システム
事前準備	1 ヤマトビジネスメンバーズの「お客様コード」(*)「パスワード」でログイン ※ログインIDは、空白で設定	ヤマトビジネスメンバーズ	ヤマトビジネスメンバーズの認証
	2 メニュー画面より、「外部システムとの連携」ボタンを押下	B2クラウド (*2)	APIアクセス認証キーを発行 ※APIアクセス認証キーはB2クラウドで管理す
	3 APIアクセス認証キー確認画面にて利用規約への確認	B2クラウド (*2)	—
	4 画面にて表示された「APIアクセス認証キー」をメモ	B2クラウド (*2)	「APIアクセス認証キー」をブラウザへ表示
	5 お客様システムへログイン	お客様システム	—
	6 お客様システム管理メニューへ「お客様コード」(*)「APIアクセス認証キー」を登録	お客様システム	—
利用のシーン（画面API利用する）	7 お客様システムへログイン	お客様システム	—
	8 受注リスト等の表示	お客様システム	—
	9 受注リスト等よりB2連携対象の選択 →B2連携ボタンを押下 ※同時にクール区分などの選択も行って頂く	お客様システム	B2-APIの認証 → 受注データ等の連携 ※B2クラウドで認証の後、ヤマトビジネスメンバーズより顧客情報の最新情報を取得する
	10 B2画面へ遷移 →9.の情報が登録されている	B2クラウド (*2)	B2クラウド内部処理
	11 エラーがある場合はB2クラウドより修正する	B2クラウド (*2)	B2クラウド内部処理
	12 個数追加がある場合は、B2クラウドより追加する（API専用画面ではなくB2クラウドの基本機能を利用する）	B2クラウド (*2)	B2クラウド内部処理
	13 印刷ボタンを押下 →用紙選択	B2クラウド (*2)	B2クラウド内部処理（印刷完了でお客様システムに通知する。 ※9.で通知先URLを受領するイメージ お客様システム→B2へ伝票No紐付け情報をGetする）
	14 お客様システムメニューへ戻る	お客様システム	自動的な遷移はしない
利用のシーン（画面API利用しない）	15 お客様システムへログイン	お客様システム	—
	16 受注リスト等の表示	お客様システム	—
	17 受注リストを登録する。	お客様システム	B2クラウドに受注リストを登録 ※加工データとエラー情報を返却する。
	18 エラーを修正する	お客様システム	B2クラウドで修正したデータの更新を行う。 ※更新データを返却する。（エラー含む場合有）

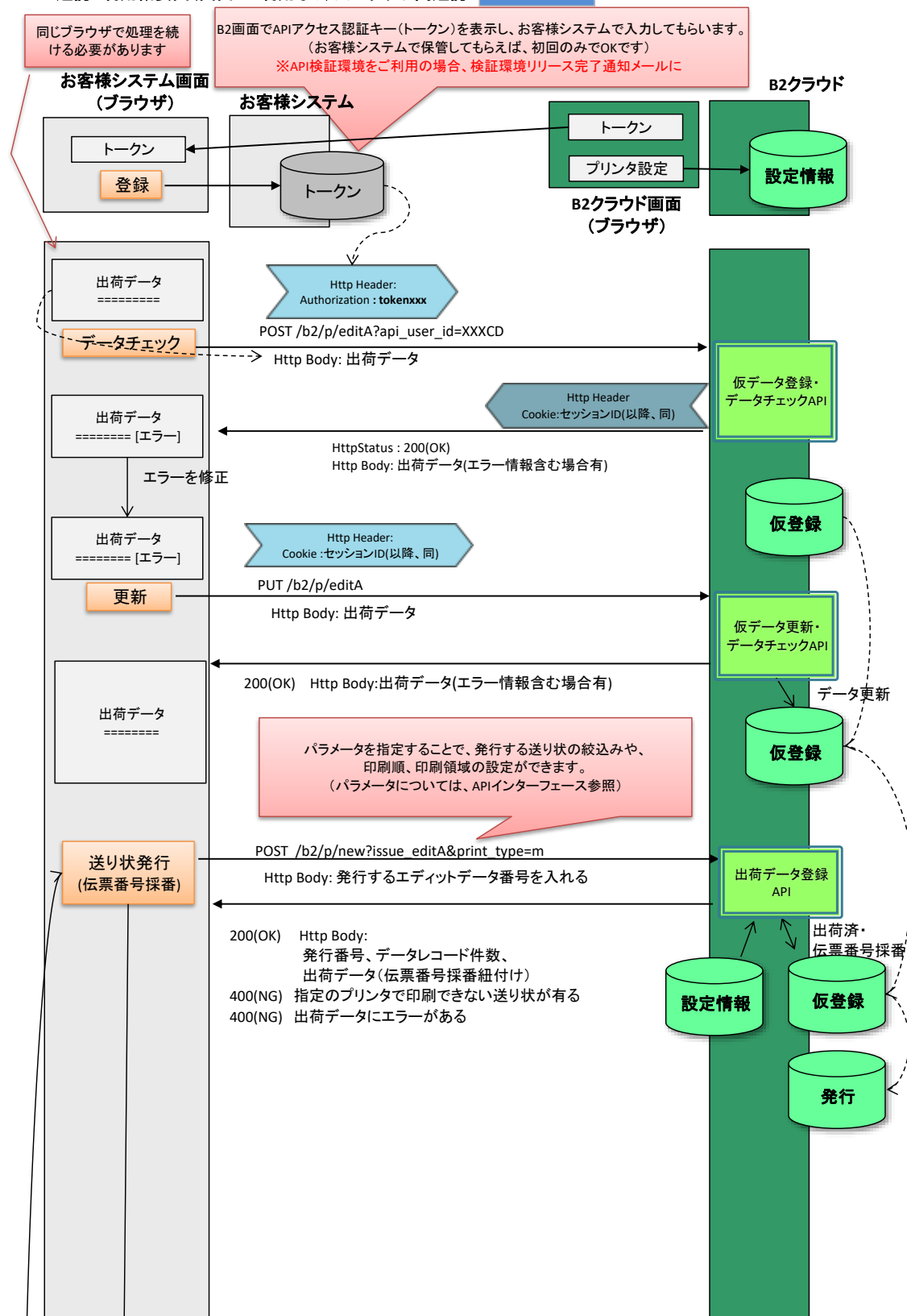
A P I 利 用 な し)	19	送り状を発行する。	お客様システム	B2クラウドに発行データの登録を行う。 ※伝票番号付データを返却する。
	20	印刷状態確認をする。	お客様システム	発行データの印刷状態を確認する。
	21	印刷イメージを取得し、送り状を印刷する。	お客様システム	発行データの印刷イメージ(PDF)を取得する。

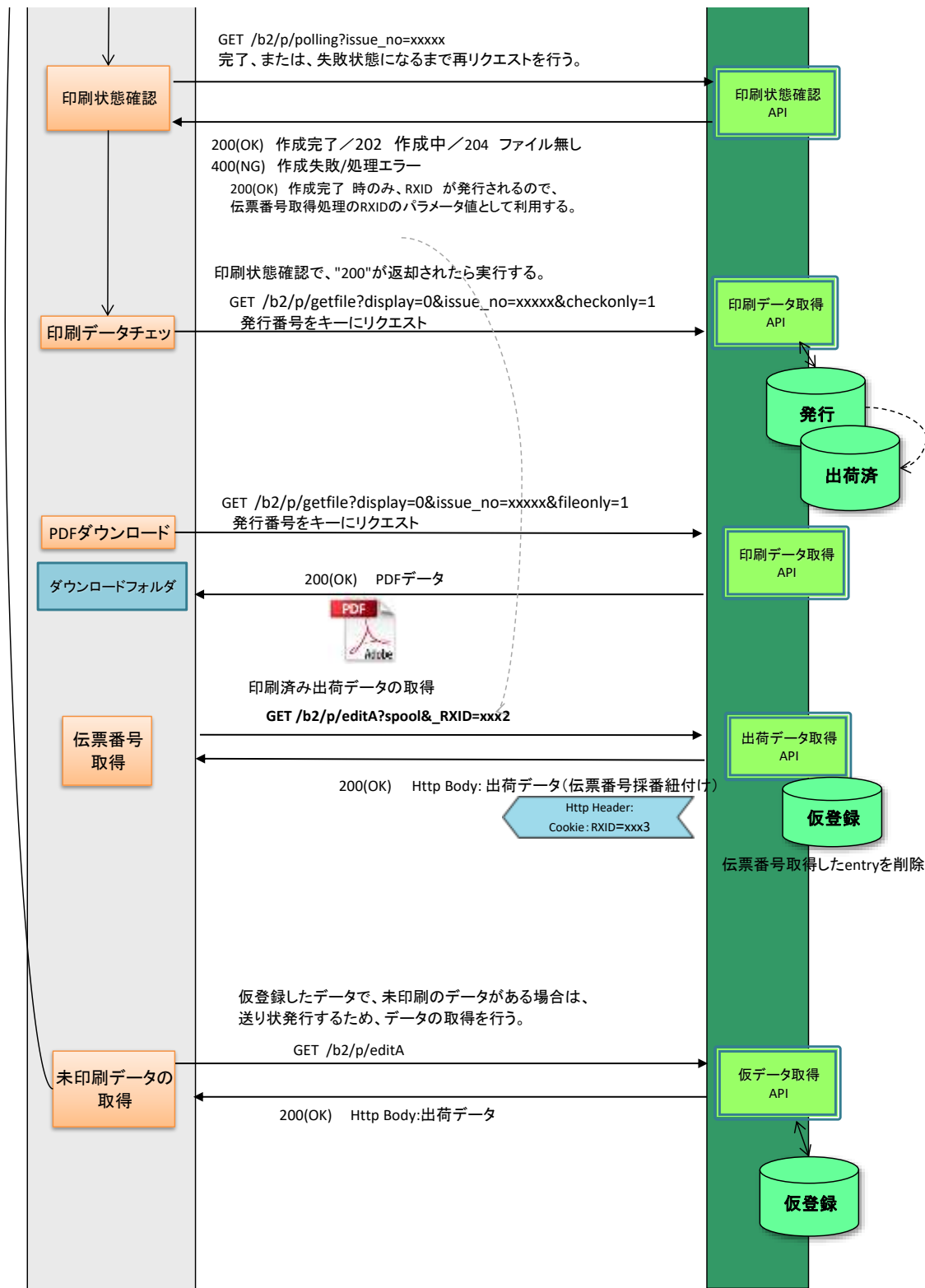
*1 未収契約時に発行するコード「顧客コード＋分類コード(枝番)」

*2 送り状発行に必要な機能に制限。(情報検索、マスタメンテナンス等は利用できない想定です。)

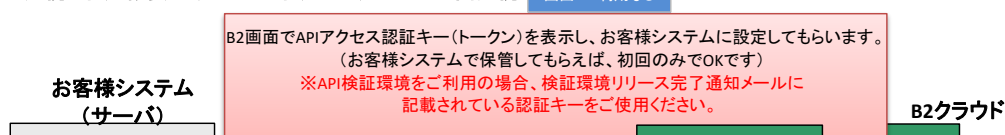
4.1 API連携 利用概要図(画面API利用なし)※ブラウザ間連携

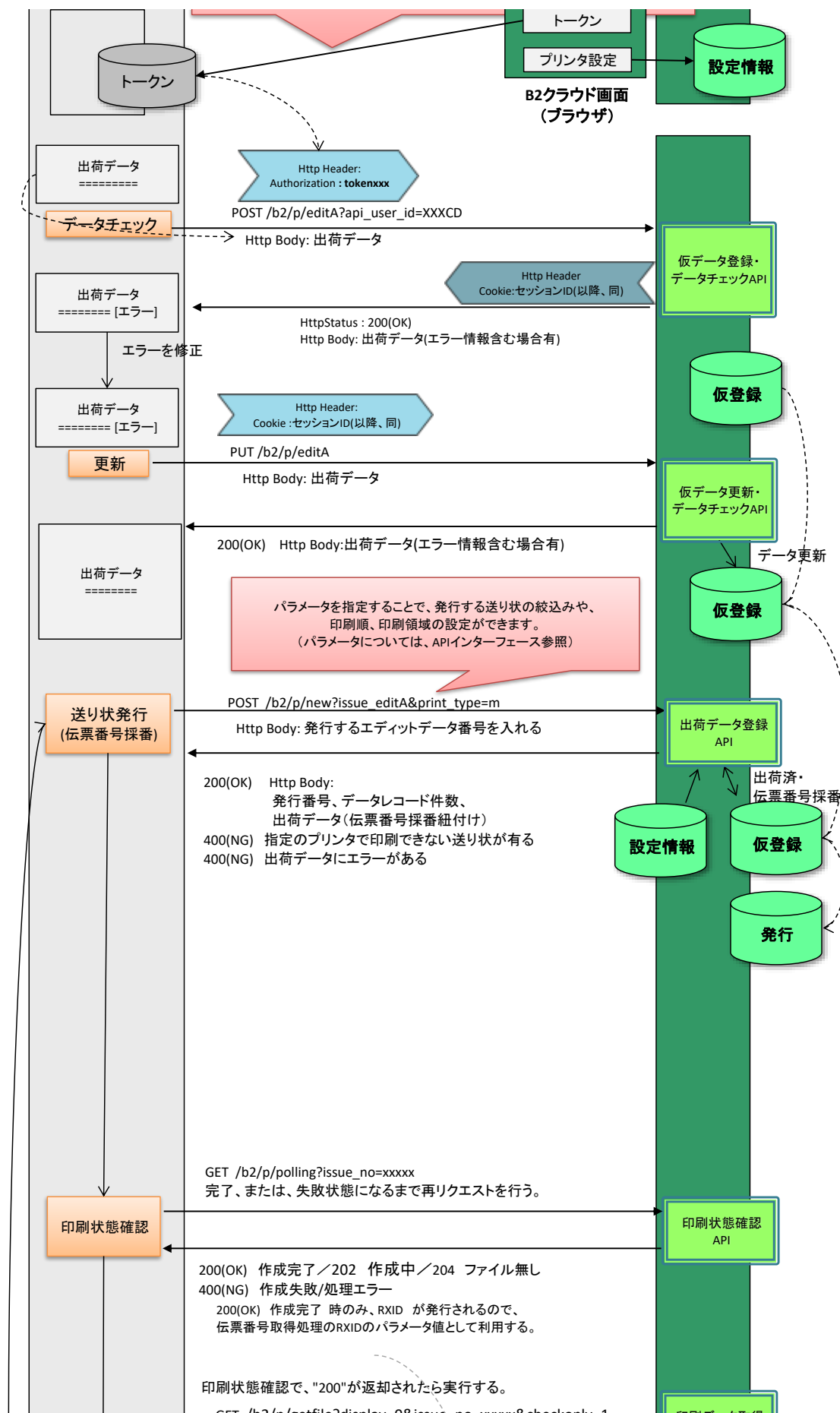
画面API利用なし

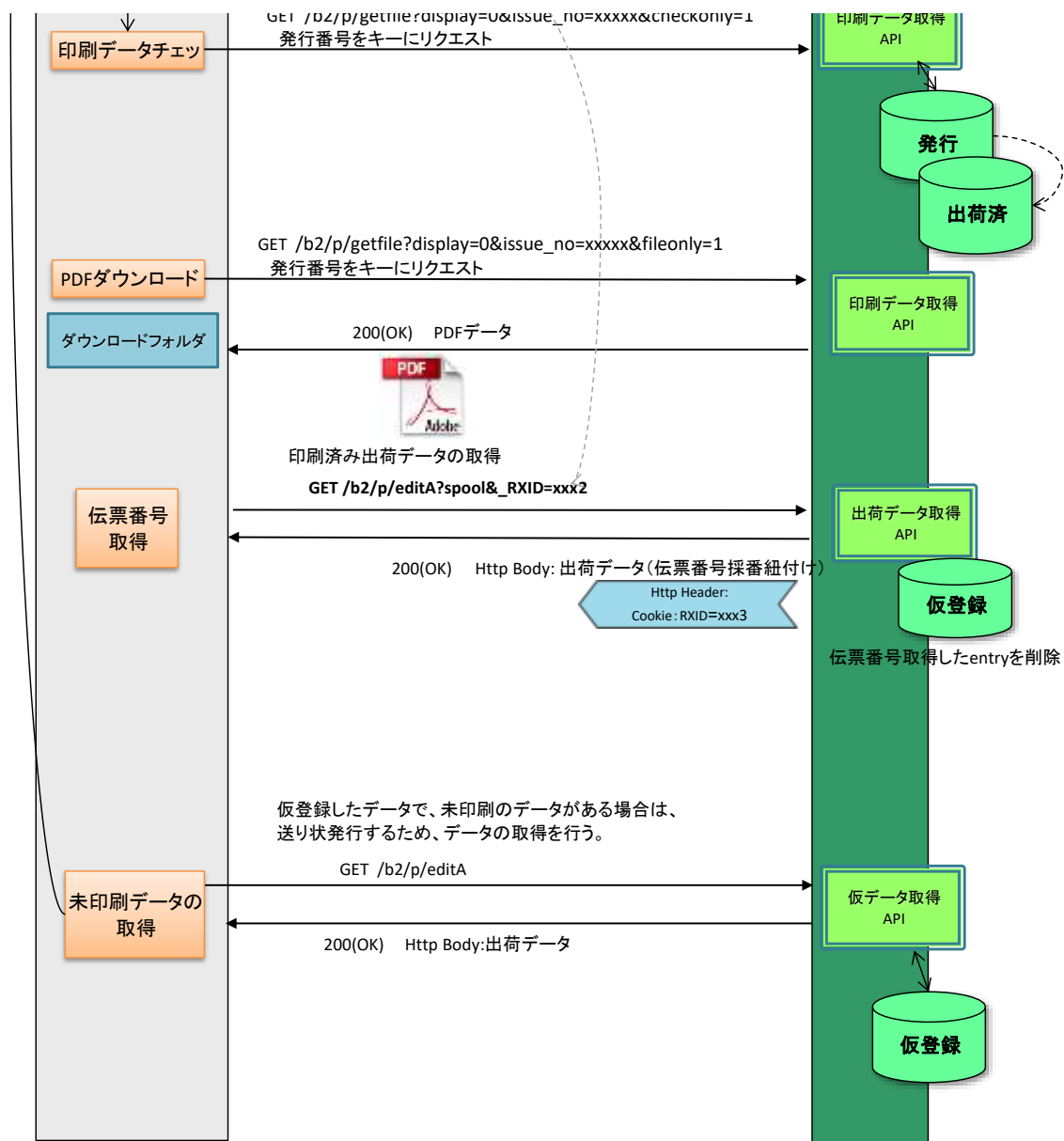




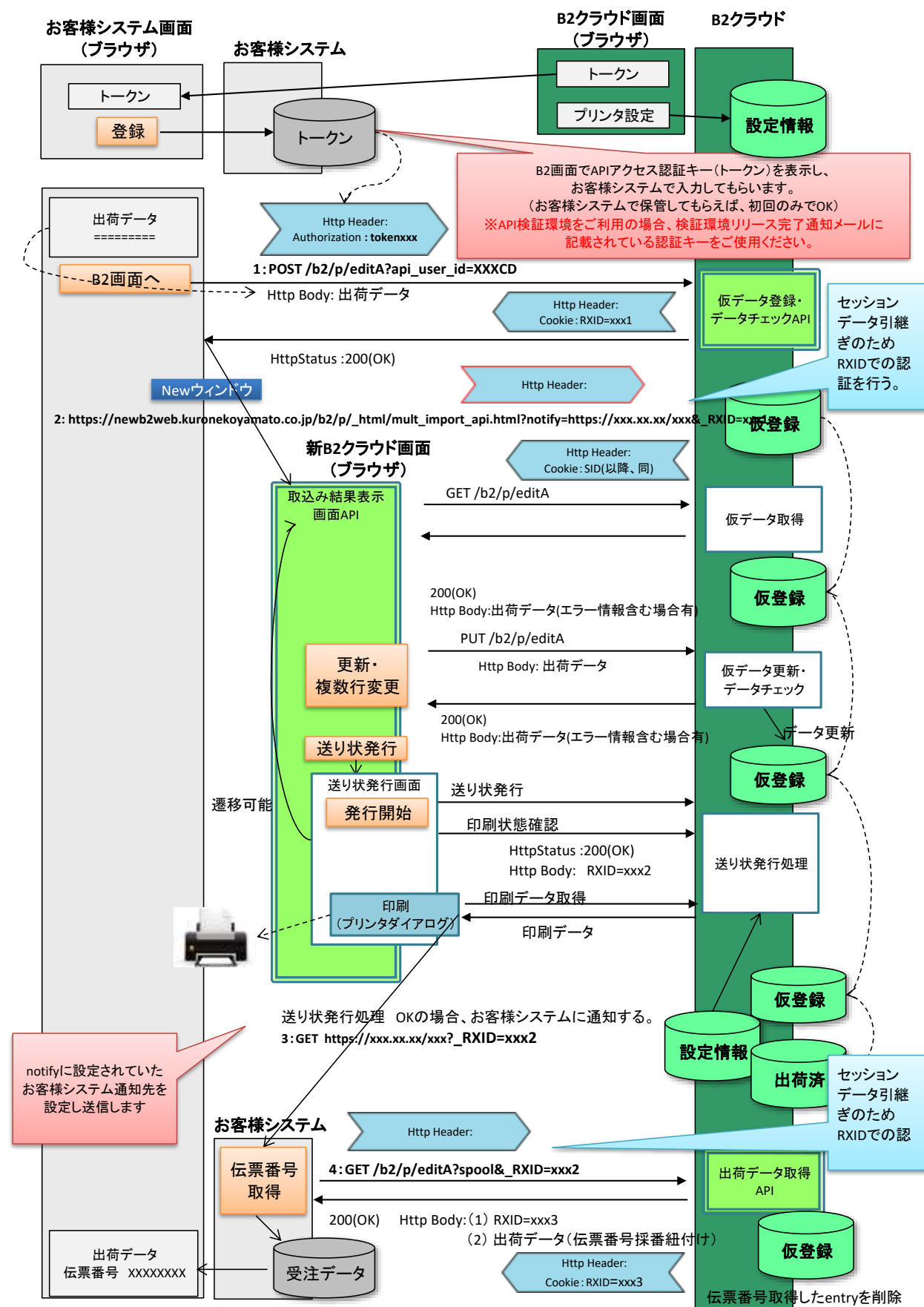
4.2 API連携 利用概要図(画面API利用なし)※サーバ間連携







3.API連携利用概要図(画面API利用) 画面API利用あり



画面有り接続URL一覧

1	POST /b2/p/editA&api_user_id=XXXCD (API連携会社コード)	5.認証について 7-1.APIインターフェース ■仮データ登録・データチェック 参照
2	https://newb2web.kuronekoyamato.co.jp/b2/p/_html/mult_import_api.html?notify=https://RMS.co.jp/xxx&RXID={前処理で返却されたRXID①}	7-1.APIインターフェース ■取込み結果表示画面 参照
3	GET https://xxx.xx.xx/xxx?RXID={RXID②}	伝票番号取得用お客様システムの通知URL
4	GET /b2/p/editA?spool&RXID={伝票番号取得用お客様システムの通知URLで返却されたRXID②}	7-1.APIインターフェース ■出荷データ取得 参照

1. はリクエストの際、Authorizationヘッダに各ユーザーに払い出しされるトークンを設定してください。

※5.認証について 参照

5. 認証について

画面API利用あり

画面API利用なし

B2クラウドでは、Accesstoken(時間制限無しの認証トークン)を使用して、認証を行います。
以下の値をHTTPリクエストヘッダに付与することで、B2クラウドで認証が実行されます。
なお、認証は、一回の業務の流れの中でB2クラウドへ接続する一番最初のリクエストで行ってください。
認証後は、セッションIDがCookieに設定され、セッションIDを用いて認証いたします。

使用方法:

・業務上、一番最初のリクエスト

HTTP リクエストヘッダに以下を指定します。

```
Authorization : Token {Accesstoken}
```

{Accesstoken}には、個々のユーザーのトークンを設定します。
個々のユーザーのトークンは、B2クラウドの画面で「API認証トークン」として表示されます。
ユーザーが「APIアクセス認証キー」を取得する方法につきましては、「5-1 APIアクセス認証キー取得方法」をご参照ください。

※API検証環境をご利用の場合、検証環境リリース完了通知メールに記載されている認証キーをご使用ください。

認証エラーや有効期限切れの場合は、HTTPステータスコードとボディ部の“title”にエラーが返却されます。
HTTPステータスコードとエラー内容については、「8.HTTPステータスについて」をご参照ください。

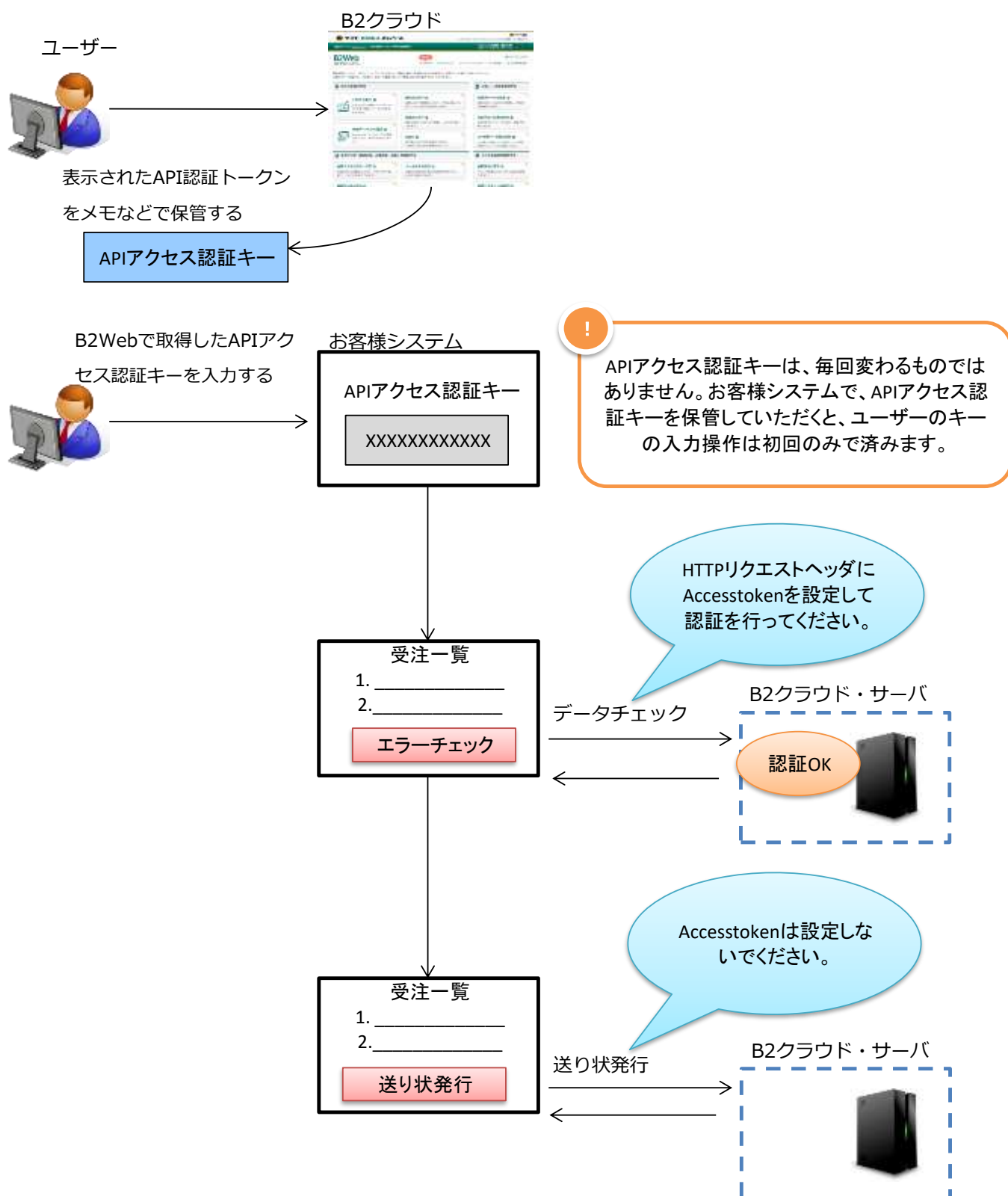
クロスサイトアクセスになるため、JavaScriptを利用したアクセスを行う場合
XMLHttpRequest の **withCredentials** の値を **true** に設定します。

・それ以降のリクエスト

CookieにセッションIDまたはRXIDが設定されるため、Accesstokenの設定はしないでください。

使用例:

一回の業務が、受注一覧のデータのエラーチェック⇒送り状発行 である場合



5-1.APIアクセス認証キー取得方法

画面API利用あり

画面API利用なし

当インターフェースの認証情報を、メインメニュー→「外部システムの連携」画面にて確認します。

※API検証環境をご利用の場合、検証環境リリース完了通知メールに記載されている認証キーをご使用ください。

The screenshot displays the Yamato Business Members B2 Cloud interface. At the top, there's a header with the Yamato logo and navigation links. Below this, a main menu bar includes options like 'Main Menu' and 'B2 Cloud'. The central area is titled '送り状発行システムB2クラウド' (B2 Cloud Shipping Label Issuance System). It features several functional blocks:

- 送り状を発行する (Issue Shipping Labels):** Includes options for 'かんたん発行' (Easy Issue), '1件ずつ発行' (Issue One by One), '外部データから発行' (Issue from External Data), '保存分の発行' (Issue from Saved Data), '履歴から発行' (Issue from History), and '再発行' (Re-issue).
- 状況/一覧表を確認する (Check Status/Summary Table):** Includes '発行済データの検索' (Search for Issued Data), '出荷予定一覧表の印刷' (Print Shipping Schedule Summary Table), and 'クロネコゆうメール番号一覧表の印刷' (Print Kuroneko Yuu Mail Number Summary Table).
- 各種マスタ（お届け先、ご依頼主、品名）を登録する (Register Various Master Data):** Includes '各種マスタの登録・編集' (Register/Edit Various Master Data), 'メール本文の登録' (Register Email Body), and '外部データの登録' (Register External Data).
- システム設定を変更する (Change System Settings):** Includes '各種設定の変更' (Change Various Settings) and '外部システムとの連携' (Linkage with External Systems).

The '外部システムとの連携' (Linkage with External Systems) option is highlighted with a red box, indicating the path to the API authentication key settings.

一度もAPIアクセス認証キーを設定していないお客様は、
「APIアクセス認証キー取得」ボタンを押下し、APIアクセス認証キーを取得してください。
APIアクセス認証キーが表示されます。
表示されたAPIアクセス認証キーをお客様システムに登録していただきます。
※ユーザ登録直後でまだ発行を行ったことがない場合は、B2クラウドにログインした際に初期設定ウィザードにてプリンタの設定を行っていただく必要があります。

 ヤマト ビジネス メンバーズ

ヤマト運輸

[ヤマトビジネスメンバーズについて](#) [よくあるご質問](#) [お問合せ](#)

現在のページ: [メインメニュー](#) > [B2クラウド](#) > [APIアクセス認証キー確認](#)

ヤマトシステム開発株式会社 [ます山 征夫 様](#) [ログアウト](#)
前回ログイン日時: 2011/09/01 14:12:34

外部システムとの連携 [詳細はこちら](#)

[B2クラウド TOPへ](#)

APIアクセス認証キー確認

B2クラウドの利用規約に同意されました。接続元システムにAPIアクセス認証キーを登録してください。

[送り状発行システム「B2クラウド」利用規約](#)

APIアクセス認証キー

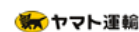
APIアクセス認証キー取得

[運送RMSをご利用の場合はこちら](#)

[利用規約](#) [個人情報の取り扱いについて](#) [各種行政](#) [ヤマト運輸TOP](#)

Copyright © 2016 Yamato Transport co., Ltd. All rights reserved.

表示されたAPIアクセス認証キーをお客様システムに登録していただきます。
「APIアクセス認証キー」ボタンを押下すると新しいものが表示されます。
※ユーザ登録直後でまだ発行を行ったことがない場合は、B2クラウドにログインした際に初期設定ウィザードにてプリンタの設定を行っていただく必要があります。



[ヤマトビジネスメンバーズについて](#) [よくあるご質問](#) [お問い合わせ](#)

現在のページ: [メインメニュー](#) > [B2クラウド](#) > [APIアクセス認証キー確認](#)

B2Webチーム 黒猫 太郎 様 [ログアウト](#)
前回ログイン日時: 2016/10/05 19:12

外部システムとの連携 [詳細はこちら](#)

[← B2クラウド TOPへ](#)

APIアクセス認証キー確認

B2クラウドの利用規約に同意されましたら、接続元システムにAPIアクセス認証キーを登録してください。

[送り状発行システム「B2クラウド」利用規約](#)

APIアクセス認証キー

password

[APIアクセス認証キー変更](#)

[楽天RMSをご利用の場合はこちら](#)

[利用規約](#) [個人情報の取り扱いについて](#) [各種約款](#) [ヤマト運輸TOP](#)

6.通信プロトコル

画面API利用あり

画面API利用なし

1. データの構造

B2クラウドで取り扱うデータはATOM(RFC4287)によって表現されます。

文字コードは、UTF-8です。

JSON、または、MessagePack形式に変換してください。

・feedとentryについて

- ・ データはリスト形式で“feed”の中に複数の“entry”が入る形になります。
- ・ “entry”はリソースの最小単位であり、1件のデータが1entryとなります。
- ・ “entry”は一つの“id”を持ちます。これは、システム全体で一意となるように、KeyとRevisionを組み合わせた形をしています。
- ・ “id”の値には、Entryの実体を示す一意の値(Key,Revision)が入ります。
- ・ Revisionは“entry”の更新回数です。
- ・ データをPOST、PUTする際、B2クラウドサーバで持っているデータのRevisionとリクエストデータのRevisionを比較します。
- ・ リクエストで送付されたRevisionのほうが古い場合、HTTPステータスコード409で排他エラーを返却します。
- ・ 排他エラーが発生した場合、「仮データ取得API」を使用して、最新のRevisionのデータを取得してください。最新のRevisionのデータを使用して、再度リクエストの送信を行ってください。

・Entityについて

- ・ “entry”を構成する要素を“Entity”と呼びます。
- ・ “shipment”は、送り状発行に関する項目を持ったEntityです。
論理名では出荷エンティティと呼ばれます。
出荷エンティティが持つ項目を「9.出荷データ仕様」に記載しております。

上記の内容をJSON形式で表わすと以下ようになります。

```
{
  "feed":{
    "entry":[
      {
        "shipment":{
          "tracking_number":"1",
          "shipment_number":"15",
          "service_type":"7",
          "is_cool":"0",
          "shipment_date":"20160830",
            ~ 省略 ~
        },
        "id":"1",
        "link":[
          {
            "___href":"1",
            "___rel":"self"
          }
        ]
      },
      {
        "shipment":{
          "tracking_number":"2",
          "shipment_number":"16",
          "service_type":"7",
          "is_cool":"0",
          "shipment_date":"20160830",
            ~ 省略 ~
        },
        "id":"2",
        "link":[
          {
            "___href":"2",
            "___rel":"self"
          }
        ]
      }
    ]
  }
}
```


2. RESTについて

B2クラウドは、REST(Representational State Transfer)の概念に基づき設計されており、リソースを一意に特定するURLに対してCRUD操作をします。
当インターフェースの通信プロトコルはHTTPSで、利用できる操作はGET、POST、PUTになります。
HTTPステータスコードとエラー内容については、「8.HTTPステータスについて」をご参照ください。

GETが成功の場合、レスポンスとして以下の内容が返却されます。

HTTP Status	200
ボディ部	応答データ

POSTが成功の場合、レスポンスとして以下の内容が返却されます。

HTTP Status	200
ボディ部	なし、または、応答データ

PUTが成功の場合、レスポンスとして以下の内容が返却されます。

HTTP Status	200
ボディ部	応答データ

リクエストが不成功の場合、レスポンスとして以下の内容が返却されます。

HTTP Status	200以外(HTTPリターンコード)
feed.title	エラーメッセージ
feed.subtitle	エラーコードなど(ない場合も有る)

・リクエストが不成功の場合のボディ部の例

```
{
  "feed" :{
    "title" : "Access denied."
  }
}
```

6-1.通信データ作成ライブラリの利用

画面API利用あり

画面API利用なし

Feedの作成は、ライブラリを使用することで簡単に行うことができます。
必要な場合は提供いたしますので、ご連絡ください。

```
reflexcore-4.0.1.jar  
reflexmapper-4.0.1.jar  
commons-codec-1.9.jar  
javassist-3.18.1-GA.jar  
json-20140107.jar  
json-simple-1.1.1.jar  
msgpack-0.6.12.jar  
xpp3-1.1.3.4-RC8.jar  
xstream-1.1.2.jar  
entity.jar
```

・ライブラリを使用した出荷データの作成例

```
import _b2.Entry;  
import _b2.Feed;  
import _b2.Shipment;  
import jp.reflexworks.atom.entry.FeedBase;  
import jp.reflexworks.atom.mapper.FeedTemplateMapper;  
  
public class EntryTest {  
    /** Mapper */  
    protected FeedTemplateMapper mapper;  
  
    public EntryTest() {  
        createTestData();  
    }  
  
    private void createTestData() {  
        // feedの作成  
        FeedBase feed = new Feed();  
  
        // entryの作成  
        Entry entry = new Entry();  
  
        // shipment Entityの作成  
        Shipment shipment = new Shipment();  
  
        // shipmentの項目に値を設定する。  
        shipment.shipment_number = "12345678901234567890";  
        shipment.service_type = "0";  
        shipment.is_cool = "0";  
  
        // shipmentの内容をentryに設定する。  
        entry.setShipment(shipment);  
  
        // feedにentryを設定する。  
        feed.addEntry(entry);  
  
        // feedをJSONに変換する。  
        String jsonData = mapper.toJSON(feed);
```

6-2.MessagePackの利用

画面API利用あり

画面API利用なし

リクエストデータやレスポンスデータに、MessagePack形式を使用することで、通信量を軽減させることができます。また、MessagePack形式のデータを圧縮することで、更に通信量を減らすことができます。そのため、B2クラウドでは、MessagePack形式で圧縮されたデータを使用することを推奨しております。

使用方法:

FeedからMessagePack形式への変換は、ライブラリ(「6-1.通信データ作成ライブラリの利用」を参照)を使用すると簡単に行うことができます。

・ライブラリを使用したMessagePack形式の利用とMessagePackの圧縮例

```
public void createMessagePack(FeedBase feed) {
    byte[] msgData = null;
    if (feed != null) {
        // MessagePack形式へ変換
        msgData = mapper.toMessagePack(feed);
        if (deflateUtil != null) {
            // MessagePack形式のデータを圧縮
            msgData = deflateUtil.deflate(msgData);
        }
    }
}
```

7.利用可能API一覧

画面API利用あり

画面API利用なし

・“B2クラウド画面を利用する場合”に利用可能なAPI一覧

画面API利用あり

API	説明
仮データ登録・データチェック	今回発行する出荷データを一時的に登録し、出荷データのチェックを行う。
取込み結果表示画面	B2クラウド画面の画面を表示する。取込み結果表示画面から、出荷データの編集や、送り状の発行ができる。
印刷データ登録	印刷実行した出荷データを登録し、送り状PDFデータの存在チェックを行う。
伝票番号取得	印刷発行済みの出荷データを取得する。

・“B2クラウド画面を利用しない場合”に利用可能なAPI一覧

画面API利用なし

API	説明
仮データ登録・データチェック	今回発行する出荷データを一時的に登録し、出荷データのチェックを行う。
仮データ取得	B2に登録した今回発行用の未印刷の出荷データを取得する。
仮データ更新・データチェック	出荷データの編集やエラーの修正後、出荷データのチェックを行う。
送り状発行	伝票番号を採番し、発行済みの出荷データとし、印刷データの作成を開始する。
印刷状態確認	印刷データの作成が完了しているかどうか確認を行う。
印刷データ登録	印刷実行した出荷データを登録し、送り状PDFデータの存在チェックを行う。
PDFダウンロード	印刷実行した送り状をPDFの形式で取得する。
伝票番号取得	印刷発行済みの出荷データを取得する。

7-1.APIインターフェース

画面API利用あり

画面API利用なし



B2クラウドの認証については、「5.認証について」、
MessagePack形式については、「6-2.MessagePackの利用」をご参照ください。

■仮データ登録・データチェック

画面API利用あり

画面API利用なし

概要

HTTPメソッド

リクエストURL

出荷データのチェックを行い、出荷データを登録する。

POST

/b2/p/editA?api_user_id=xxxCD

URLパラメータ

パラメータ	必須	有効な設定値	説明
api_user_id	○	指定されたAPI連携会社コード	「2.ユーザ操作フロー」を参照

リクエストヘッダ

リクエストヘッダ	必須	設定値	説明
Content-Type	○	application/x-msgpack application/json	リクエストデータがMessagePack形式の場合の指定。 リクエストデータがJSON形式の場合の指定。
Content-Encoding		deflate	リクエストデータがMessagePack形式で、かつ、圧縮されている場合の指定。 リクエストデータがJSON形式の場合は、指定不要。
Authorization	○	APIアクセス認証キー	B2画面に表示しているAPIアクセス認証キー(トークン)
Accept-Encoding		deflate	レスポンスデータをMessagePack形式で受け取る時、 圧縮されたデータが戻る。 JSON形式でレスポンスデータを受け取る場合、指定不要。

クロスサイトアクセスになるため、
XMLHttpRequest の withCredentials の値を true に設定してリクエストしてください。

リクエストボディ

対象の出荷データを設定

JSON形式、または、MessagePack形式のデータが指定可能

出荷データは、本稿の「6.通信プロトコル」のデータの構造、「9.出荷データ仕様」に沿って作成してください。

レスポンス

成功	ステータスコードは200で、レスポンスボディにはチェック済みの出荷データが返却されます。
その他	詳細は、「8.HTTPステータスについて」をご参照ください。

レスポンスヘッダ

リクエストヘッダ	設定値	説明
Content-Type	application/json	レスポンスデータはJSON形式。

レスポンスボディ

本稿の「6.通信プロトコル」のデータの構造に沿って返却されます。

返却される形式はJSON形式となります。

feed の title に、RXID が設定されます。

画面API利用あり の 取込み結果表示画面 を行う際に、パラメータのRXIDに設定してください。

項目	説明
feed	
title	RXID(正常の場合)
updated	正常時設定。更新日時(ミリ秒)「例」1494822648136 ※7-3.排他制御参照
entry[]	
shipment	本稿の「9.出荷データ仕様」参照
error	リクエストの出荷データにエラーがある場合、設定される。「8.HTTPステータスについて」参照

■仮データ取得

画面API利用なし

概要
HTTPメソッド GET
リクエストURL /b2/p/editA

未発行の出荷データを取得する。

リクエストヘッダ

リクエストヘッダ	必須	設定値	説明
Accept-Encoding		deflate	レスポンスデータをMessagePack形式で受け取る時、圧縮されたデータが戻る。 JSON形式でレスポンスデータを受け取る場合、指定不要。

クロスサイトアクセスになるため、XMLHttpRequest の withCredentials の値を true に設定してリクエストしてください。

リクエストボディ なし

レスポンス

成功	ステータスコードは200で、レスポンスボディには未発行の出荷データが返却されます。詳細は、「9.出荷データ仕様」をご参照ください。
その他	詳細は、「8.HTTPステータスについて」をご参照ください。

レスポンスヘッダ

リクエストヘッダ	設定値	説明
Content-Type	application/json	レスポンスデータはJSON形式。

レスポンスボディ 本稿の「6.通信プロトコル」のデータの構造に沿って返却されます。返却される形式はJSON形式となります。

項目	説明
feed	
updated	正常時設定。更新日時(ミリ秒)「例」1494822648136 ※7-3.排他制御参照
entry	
shipment	本稿の「9.出荷データ仕様」参照

■ 仮データ更新・データチェック

画面API利用なし

概要 出荷データのチェックを行い、出荷データを更新する。
「仮データ登録・データチェック」APIの使用後に、データを更新する場合に利用する。

HTTPメソッド PUT

リクエストURL /b2/p/editA

URLパラメータ

パラメータ	必須	有効な設定値	説明
multi		-	複数行変更の場合に設定

リクエストヘッダ

リクエストヘッダ	必須	設定値	説明
X-Requested-With	○	XMLHttpRequest	CSRF対策のため、必須で設定する。設定がなければエラーを返却する。(HTTPステータスコード417)
Content-Type	○	application/x-msgpack application/json	リクエストデータがMessagePack形式の場合の指定。 リクエストデータがJSON形式の場合の指定。
Content-Encoding		deflate	リクエストデータがMessagePack形式で、かつ、圧縮されている場合の指定。 リクエストデータがJSON形式の場合は、指定不要。
Accept-Encoding		deflate	レスポンスデータをMessagePack形式で受け取る時、 圧縮されたデータが戻る。 JSON形式でレスポンスデータを受け取る場合、指定不要。

クロスサイトアクセスになるため、
XMLHttpRequest の **withCredentials** の値を **true** に設定してリクエストしてください。

リクエストボディ

対象の出荷データを設定
JSON形式、または、MessagePack形式のデータが指定可能
出荷データは、本稿の「6.通信プロトコル」のデータの構造、「9.出荷データ仕様」に沿って作成してください。

項目	説明
feed	
updated	必須項目。更新日時(ミリ秒)「例」1494822648136 ※7-3.排他制御参照
entry[]	
shipment	
created_ms	「仮データ登録・データチェック」のレスポンス結果で取得した値 (必須)
tracking_number	「仮データ登録・データチェック」のレスポンス結果で取得した値 (必須)
上記以外の項目	本稿の「9.出荷データ仕様」参照

レスポンス

成功(登録/更新)	ステータスコードは200で、レスポンスボディにはチェック済みの出荷データが返却されます。 出荷データでチェックエラーがあった項目についても、レスポンスボディに記載されて返却されます。詳細は、「9.出荷データ仕様」をご参照ください。
その他	詳細は、「8.HTTPステータスについて」をご参照ください。

レスポンスヘッダ

リクエストヘッダ	設定値	説明
Content-Type	application/json	レスポンスデータはJSON形式。

レスポンスボディ

本稿の「6.通信プロトコル」のデータの構造に沿って返却されます。
返却される形式はJSON形式となります。

項目	説明
feed	
updated	正常時設定。更新日時(ミリ秒)「例」1494822648136 ※7-3.排他制御参照
entry[]	
shipment	本稿の「9.出荷データ仕様」参照
error	リクエストの出荷データにエラーがある場合、設定される。「8.HTTPステータスについて」参照

■仮データ削除

画面API利用なし

概要	出荷データの削除をする。 仮データ登録・更新したデータを一部削除する場合に用いる。
HTTPメソッド	POST
リクエストURL	/b2/p/editA

リクエストヘッダ

リクエストヘッダ	必須	設定値	説明
X-Requested-With	○	XMLHttpRequest	CSRF対策のため、必須で設定する。設定がなければエラーを返却する。(HTTPステータスコード417)
Content-Type	○	application/x-msgpack application/json	リクエストデータがMessagePack形式の場合の指定。 リクエストデータがJSON形式の場合の指定。
Content-Encoding		deflate	リクエストデータがMessagePack形式で、かつ、圧縮されている場合の指定。 リクエストデータがJSON形式の場合は、指定不要。
Accept-Encoding		deflate	レスポンスデータをMessagePack形式で受け取る時、 圧縮されたデータが戻る。 JSON形式でレスポンスデータを受け取る場合、指定不要。
X-HTTP-Method-Override	○	DELETE /editA	削除処理で必須となる設定。設定がない場合は通常のPOSTとして取り扱われる。

クロスサイトアクセスになるため、
XMLHttpRequest の **withCredentials** の値を **true** に設定してリクエストしてください。

リクエストボディ	削除対象の出荷データを設定 JSON形式、または、MessagePack形式のデータが指定可能 出荷データは、本稿の「6.通信プロトコル」のデータの構造、「9.出荷データ仕様」に沿って作成してください。
----------	---

項目	説明
feed	
updated	必須項目。更新日時(ミリ秒)「例」1494822648136 ※7-3.排他制御参照
entry[]	削除対象のEntryを設定する
shipment	本稿の「9.出荷データ仕様」参照
tracking_number	「■仮データ登録・データチェック」で取得した際に採番された番号の設定が必須

レスポンス

成功	ステータスコードは200で返却されます。
その他	詳細は、「8.HTTPステータスについて」をご参照ください。

レスポンスヘッダ

リクエストヘッダ	設定値	説明
Content-Type	application/json	レスポンスデータはJSON形式。

レスポンスボディ	本稿の「6.通信プロトコル」のデータの構造に沿って返却されます。 返却される形式はJSON形式となります。 成功時に、feedのtitleに“Deleted.”を設定。
----------	--

項目	説明
feed	
updated	正常時設定。更新日時(ミリ秒)「例」1494822648136 ※7-3.排他制御参照
title	“Deleted.”
entry[]	

■送り状発行 画面API利用なし

概要

伝票番号を採番し、出荷データを発行する。また、印刷データの作成を開始する。
 出荷データにエラーがある場合は、印刷データの作成を行えない。
 事前に仮データ登録、更新処理で出荷データチェックが実施された
 レスポンスデータをリクエストBODYに設定する。

HTTPメソッド

POST

リクエストURL

/b2/p/new?issue_editA

URLパラメータ【送り状発行の指定】

パラメータ	必須	有効な設定値		説明
display	○	0	画面なし	固定値
print_type	○	m	A4マルチ印刷	印刷する送り状種類を指定する。 複数の値を指定不可。 【マルチ印刷について】 ・プリンタが、レーザー、インクジェットの場合 発払い、コレクト、タイム、着払い、コンパクト、コンパクトコレクトの送り状を一括で印刷できる。 ・プリンタがサーマルの場合 発払い、コンパクトの送り状を一括で印刷できる。
		m5	A5マルチ印刷	
		0	発払い	
		2	コレクト	
		4	タイム	
		3	クロネコゆうメール	
		7	ネコボス・クロネコゆうパケット	
		CP	ネコボス・クロネコゆうパケット	
print_position		print_type=3(クロネコゆうメール)の場合 1~8の数字 print_type=7(ネコボス・クロネコゆうパケット)の場合 1~6の数字		クロネコゆうメール、ネコボス・クロネコゆうパケットの印刷位置(※1)を設定する。 複数の値を指定可能。 送り状種類がクロネコゆうメールかネコボス・クロネコゆうパケットの場合に設定する。設定無しの場合は、印刷位置“1”から順番に印刷される。
sort1		service_type	送り状種類	印刷の順序を設定する。 sort1・・・最優先される項目 sort2・・・2番目に優先される項目 sort3・・・3番目に優先される項目 指定無しの場合、B2クラウドの各種設定で指定されている値になる。 指定する場合、sort1、sort2、sort3を必ず全て指定する。(どれかが指定されていない場合、B2クラウドの各種設定で指定されている値になるので、ユーザーの意図通りの印刷順にならない可能性があるため)
sort2		tracking_number	伝票番号	
sort3		is_cool	クール区分	
		shipment_number	お客様管理番号	
		sorting_code	仕分コード	
		shipment_date	出荷予定日	
		delivery_date	お届け予定日	
		consignee_code	お届け先コード	
		consignee_telephone	お届け先電話番号	
		consignee_zip_code	お届け先郵便番号	
		item_code1	品名コード1	
		item_code2	品名コード2	

※1 クロネコゆうメール、ネコボス・クロネコゆうパケットの印刷位置について

使用済みラベルがある場合は、その部分を除いて印刷することができます。
 印刷用紙の一枚目に、この設定が適用されます。

・クロネコゆうメール
印刷位置

1	2
3	4
5	6
7	8

・ネコボス・クロネコゆうパケット印刷位置

1	2
3	4
5	6

パラメータ使用例

- ・A4マルチ印刷で、印刷の優先順序を、送り状種類、クール区分、伝票番号で指定する場合
/b2/p/new?issue_edit&print_type=m&sort1=service_type&sort2=is_cool&sort3=tracking_number
- ・A5マルチ印刷で、印刷の優先順序を、送り状種類、クール区分、伝票番号で指定する場合
/b2/p/new?issue_edit&print_type=m5&sort1=service_type&sort2=is_cool&sort3=tracking_number
- ・クロネコゆうメールの、印刷位置“1”から順番に印刷する場合
/b2/p/new?issue_edit&print_type=3
- ・クロネコゆうメールの、印刷位置“5”、“6”、“7”、“8”に印刷する場合
/b2/p/new?issue_edit&print_type=3&print_position=5&print_position=6&print_position=7&print_position=8
- ・ネコボス・クロネコゆうパケットの、印刷位置“2”、“5”、“6”に印刷する場合
/b2/p/new?issue_edit&print_type=7&print_position=2&print_position=5&print_position=6

リクエストヘッダ

リクエストヘッダ	必須	設定値	説明
X-Requested-With	○	XMLHttpRequest	CSRF対策のため、必須で設定する。設定がなければエラーを返却する。(HTTPステータスコード417)
Content-Type	○	application/x-msgpack application/json	リクエストデータがMessagePack形式の場合の指定。 リクエストデータがJSON形式の場合の指定。
Content-Encoding		deflate	リクエストデータがMessagePack形式で、かつ、圧縮されている場合の指定。 リクエストデータがJSON形式の場合は、指定不要。
Accept-Encoding		deflate	レスポンスデータをMessagePack形式で受け取る時、 圧縮されたデータが戻る。 JSON形式でレスポンスデータを受け取る場合、指定不要。

クロスサイトアクセスになるため、
XMLHttpRequest の **withCredentials** の値を **true** に設定してリクエストしてください。

リクエストボディ

発行するデータの“id”を取得し、リクエストデータを送ってください。
JSON形式、または、MessagePack形式のデータが指定可能
※仮データ登録・更新のレスポンスデータは出荷データチェックが行われた結果です。
出荷データが行われたデータをリクエストに設定してください。

項目	説明
feed	
updated	必須項目。更新日時(ミリ秒)「例」1494822648136」※7-3.排他制御参照
entry	
shipment	
tracking_number	「仮データ登録・データチェック」のレスポンス結果で取得した値 (必須)
created_ms	「仮データ登録・データチェック」のレスポンス結果で取得した値 (必須)
service_type	送り状種類の指定が可能です。(必須) (「9.出荷データ仕様」を参照)
printer_type	プリンタ種別の指定が可能です。(任意) (「9.出荷データ仕様」を参照)
is_agent	ネコボス・クロネコゆうパケット送り状であれば「0」、 ネコボス・クロネコゆうパケット払込票であれば「1」を固定値として入力
shipment_flg	「1」を固定値として入力

・送り状発行リクエストのリクエストデータ例

```
{
  "feed":{
    "entry":[
      {
        "id":"1",
        "shipment":{
          "tracking_number":"1",
          "created_ms":"1489061253169"
          "service_type":"0",
          "printer_type":"1",
          "is_agent":"1",
          "shipment_flg":"1",
        }
      },
      {
        "id":"2",
        "shipment":{
```

```
shipment :{  
  "tracking_number":"2",  
  "created_ms":"1489061253169"  
  "service_type":"7",  
  "printer_type":"0",  
  "is_agent":"1",  
  "shipment_flg":"1",  
}  
},  
"updated":"1494823394373"
```

レスポンス

成功	ステータスコードは200です。 レスポンスボディには、Feedの“title”に「発行番号」、「subtitle”に「データレコード件数」が入ります。 Feedのentryには、伝票番号が付与された出荷データが返却されます。 出荷データの詳細は、「9.出荷データ仕様」をご参照ください。
出荷データにエラーあり	ステータスコードは400です。 レスポンスボディには、Feedの“title”に“Shipment data is invalid.”が入ります。 Feedのentryには、エラー情報が付与された出荷データが返却されます。 詳細は、「8.HTTPステータスについて」、「9.出荷データ仕様」をご参照ください。
指定のプリンタで印刷できない送り状が有る	ステータスコードは400です。 レスポンスボディには、Feedの“title”に“printer type or print_type is invalid.”が入ります。 Feedのentryには、出荷データが返却されます。 詳細は、「8.HTTPステータスについて」、「9.出荷データ仕様」をご参照ください。
出荷データ上限エラー	ステータスコードは416で返却されます。 出荷データの上限件数は1000件。※個数口の件数は1件とカウントされます。
ビジ状態	ステータスコードは419で返却されます。
その他	詳細は、「8.HTTPステータスについて」をご参照ください。

レスポンスヘッダ

リクエストヘッダ	設定値	説明
Content-Type	application/json	レスポンスデータはJSON形式。

レスポンスボディ

正常の場合

項目	定義
feed	
updated	正常時設定。更新日時(ミリ秒)「例」1494822648136」 ※7-3.排他制御参照
subtitle	帳票生成想定時間(ミリ秒) ※印刷状態確認時のリクエスト間隔に利用します。 1秒未満の場合は、1秒以上の間隔をあげてリクエストしてください。
title	発行番号 ※印刷状態確認時のリクエストパラメータ「issue」に設定します。

・成功した時のレスポンスデータ例

```
{
  "feed":{
    "title":"TMIN0000001638",
    "subtitle":"428",
    "updated":"1494823394373"
  }
}
```

■印刷状態確認

画面API利用なし

概要

印刷データの作成状況の確認を行う。
送り状発行のレスポンスで返却された帳票生成想定時間間隔をあけて、印刷状態の確認を行い印刷状態の確認で作成完了を確認後、印刷データの取得を行うことができます。

印刷データの作成が完了するまで、印刷状態確認処理を実行するajaxを利用したプログラムのサンプルをご用意しております。「7-2.印刷状態確認処理サンプル」をご参照ください。

HTTPメソッド リクエストURL

GET
/b2/p/polling?issue_no=xxx&display=0

URLパラメータ

パラメータ	必須	有効な設定値	説明
display	○	「0」 固定	送り状発行のレスポンス結果から取得する発行番号
issue_no	○	発行番号	印刷データを取得するための一意のキー

リクエストヘッダ

リクエストヘッダ	必須	設定値	説明
Content-Type	○	application/x-msgpack application/json	リクエストデータがMessagePack形式の場合の指定。 リクエストデータがJSON形式の場合の指定。
Content-Encoding		deflate	リクエストデータがMessagePack形式で、かつ、圧縮されている場合の指定。 リクエストデータがJSON形式の場合は、指定不要。
Accept-Encoding		deflate	レスポンスデータをMessagePack形式で受け取る時、 圧縮されたデータが戻る。 JSON形式でレスポンスデータを受け取る場合、指定不要。

クロスサイトアクセスになるため、
XMLHttpRequest の withCredentials の値を true に設定してリクエストしてください。

リクエストボディ

なし

レスポンス

作成完了	ステータスコードは200で返却されます。
作成中	ステータスコードは202で返却されます。
ファイルなし	ステータスコードは204で返却されます。
作成失敗 / 処理エラー	ステータスコードは400で返却されます。
その他	詳細は、「8.HTTPステータスについて」をご参照ください。

レスポンスボディ

項目	定義
Feed	
title	RXIDを設定(正常時のみ)
subtitle	ステータスコード
summary	エラーメッセージ
entry[]	
summary	印刷状態: 作成完了 "created" / 作成中 "queued" / 作成失敗 "failed to print".
title	正常(作成完了): "Success", 異常(作成完了以外): "Error"
subtile	ステータスコード

■印刷データチェック

画面API利用なし

概要

発行済みである出荷データを登録し、印刷データの存在チェックを行う。

HTTPメソッド リクエストURL

GET
/b2/p/getfile?display=0&issue_no=xxxxx&checkonly=1

URLパラメータ

パラメータ	必須	有効な設定値	説明
display	○	「0」 固定	
issue_no	○	発行番号	該当の出荷データを登録し、印刷データの存在チェックを行う。
checkonly	○	「1」 固定	

リクエストヘッダ

リクエストヘッダ	設定値	説明
なし		

クロスサイトアクセスになるため、
XMLHttpRequest の **withCredentials** の値を **true** に設定してリクエストしてください。

リクエストボディ

なし

レスポンス

HTTPステータス

成功	ステータスコードは200で返却されます。
印刷データなし / 発行番号誤り	ステータスコードは204で返却されます。
処理失敗	ステータスコードは400で返却されます。
その他	詳細は、「8.HTTPステータスについて」をご参照ください。

■PDFダウンロード

画面API利用なし

概要 発行済みである出荷データのPDFデータを取得する。

HTTPメソッド GET
リクエストURL /b2/p/getfile?display=0&issue_no=xxxxx&fileonly=1

URLパラメータ

パラメータ	必須	有効な設定値	説明
display	○	「0」 固定	
issue_no	○	発行番号	該当の印刷データを取得する
fileonly	○	「1」 固定	

リクエストヘッダ

リクエストヘッダ	設定値	説明
なし		

クロスサイトアクセスになるため、XMLHttpRequest の **withCredentials** の値を **true** に設定してリクエストしてください。

リクエストボディ なし

レスポンス

HTTPステータス

成功	ステータスコードは200で、PDFファイルを返却します。
発行番号誤り / 印刷データなし	ステータスコードは204で返却されます。
作成失敗 / 処理エラー	ステータスコードは400で返却されます。
その他	詳細は、「8.HTTPステータスについて」をご参照ください。

■取込み結果表示画面

画面API利用あり

概要

リクエストURL

出荷データの編集、送り状の発行ができるB2クラウド画面の画面を表示する。

[https://newb2web.kuronekoyamato.co.jp/b2/p/_html/multi_import_api.html?](https://newb2web.kuronekoyamato.co.jp/b2/p/_html/multi_import_api.html?notify=http://xxxx/xxx&RXID=xxx1)

[notify=http://xxxx/xxx&RXID=xxx1](https://newb2web.kuronekoyamato.co.jp/b2/p/_html/multi_import_api.html?notify=http://xxxx/xxx&RXID=xxx1)

URLパラメータ

パラメータ	必須	有効な設定値	説明
notify	○	システム通知先URL	発行完了後、B2クラウドから発行完了を知らせる通知先を設定（あらかじめ登録した先のみ設定可）
_RXID	○	RXID	「仮データ登録・データチェック」の実行結果で返却されたRXIDを設定する。（取得した値をそのまま設定）

レスポンス

成功時 B2クラウド取込み結果表示画面を表示する。



失敗 /
notify設定エラー

ステータスコードは400で返却されます。

notifyに設定したURLが事前にURL許可されているものと違う場合にはエラーとなります。

■伝票番号取得

画面API利用あり

画面API利用なし

概要

B2クラウド画面から送り状発行をした後、伝票番号が付与された出荷データを取得する。

HTTPメソッド リクエストURL

GET
/b2/p/editA?spool&RXID=xxx2

URLパラメータ

パラメータ	必須	有効な設定値	説明
_RXID	○	RXID	画面有りの場合 通知先URLのパラメータで指定されたRXIDを設定する。 画面無しの場合 印刷状態確認のレスポンスで返却されるRXIDを設定する。

リクエストヘッダ

リクエストヘッダ	必須	設定値	説明
Accept-Encoding		deflate	レスポンスデータをMessagePack形式で受け取る時、 圧縮されたデータが戻る。 JSON形式でレスポンスデータを受け取る場合、指定不要。

リクエストボディ

なし

レスポンス

成功	ステータスコードは200で、レスポンスボディには伝票番号が付与された出荷データが返却されます。 詳細は、「9.出荷データ仕様」をご参照ください。
その他	詳細は、「8.HTTPステータスについて」をご参照ください。

レスポンスヘッダ

リクエストヘッダ	設定値	説明
Content-Type	application/json	レスポンスデータはJSON形式。

Cookie

RXID	再度リクエストを設定する場合、Cookie内のRXIDを取得してリクエストパラメータに指定します。
------	---

レスポンスボディ

本稿の「6.通信プロトコル」のデータの構造に沿って返却されます。
返却される形式はJSON形式となります。
※件数が0の場合は、HTTPステータスコード200で以下のようなfeedが返ってきますので
リトライ実行をお願いします。(3秒×3回)

```
{
  "subtitle": "204"
  "summary": "no entry."
  "title": "Error"
}
```

項目	説明
feed	
title	RXID(正常の場合、Cookieと同じものを設定。リトライ時にURLパラメータに指定)
entry[]	
shipment	本稿の「9.出荷データ仕様」参照

7-2.印刷状態確認サンプル

画面API利用なし

```
function issuePrint(issue_no, cnt ,timer){
```

```
    var url = '/p/polling';
```

```
    timer = timer / 2;
```

```
    url += "?issue_no=" + issue_no
```

```
    $.b2fetch.get({  
        'url': url  
    }).then(function(){
```

印刷処理

```
    }, function(jqXHR){ // GET失敗  
        if(cnt > 10){ // 10回まで印刷確認処理を行う。  
            loadingGuard(false);  
            alert('印刷に失敗しました。');  
        }else{  
            cnt=cnt+1;  
            if(timer < 1000){ //一定時間待機する  
                setTimeout(function(){issuePrint(issue_no , cnt, timer);},1000);  
            }else{  
                setTimeout(function(){issuePrint(issue_no , cnt, timer);},timer);  
            }  
        }  
    }  
});
```

```
function (opt){
```

```
    var defer = $.Deferred();  
    _doInit();
```

```
    $.ajax({  
        type: 'get',  
        url: '/b2' + opt.url,  
        beforeSend : function(xhr){  
            xhr.setRequestHeader("If-Modified-Since", "Thu, 01 Jun 1970 00:00:00 GMT");  
        },  
        success: function(data, status, xhr){
```

```
            if (data && data.feed && data.feed.entry && data.feed.entry[0].title && data.feed.entry[0].title === 'Error') {  
                var status = data.feed.entry[0].subtitle;  
                var summary = data.feed.entry[0].summary;  
                defer.reject({'status': status,'responseText':summary});  
            } else {  
                defer.resolve(data, status, xhr);  
            }  
        },
```

```
        error: function(jqXHR, textStatus, errorThrown){  
            defer.reject(jqXHR, textStatus, errorThrown);  
        },  
        complete: function(){  
            _doFinal();  
        }  
    });  
    return defer.promise();  
};
```

7-3.排他制御について

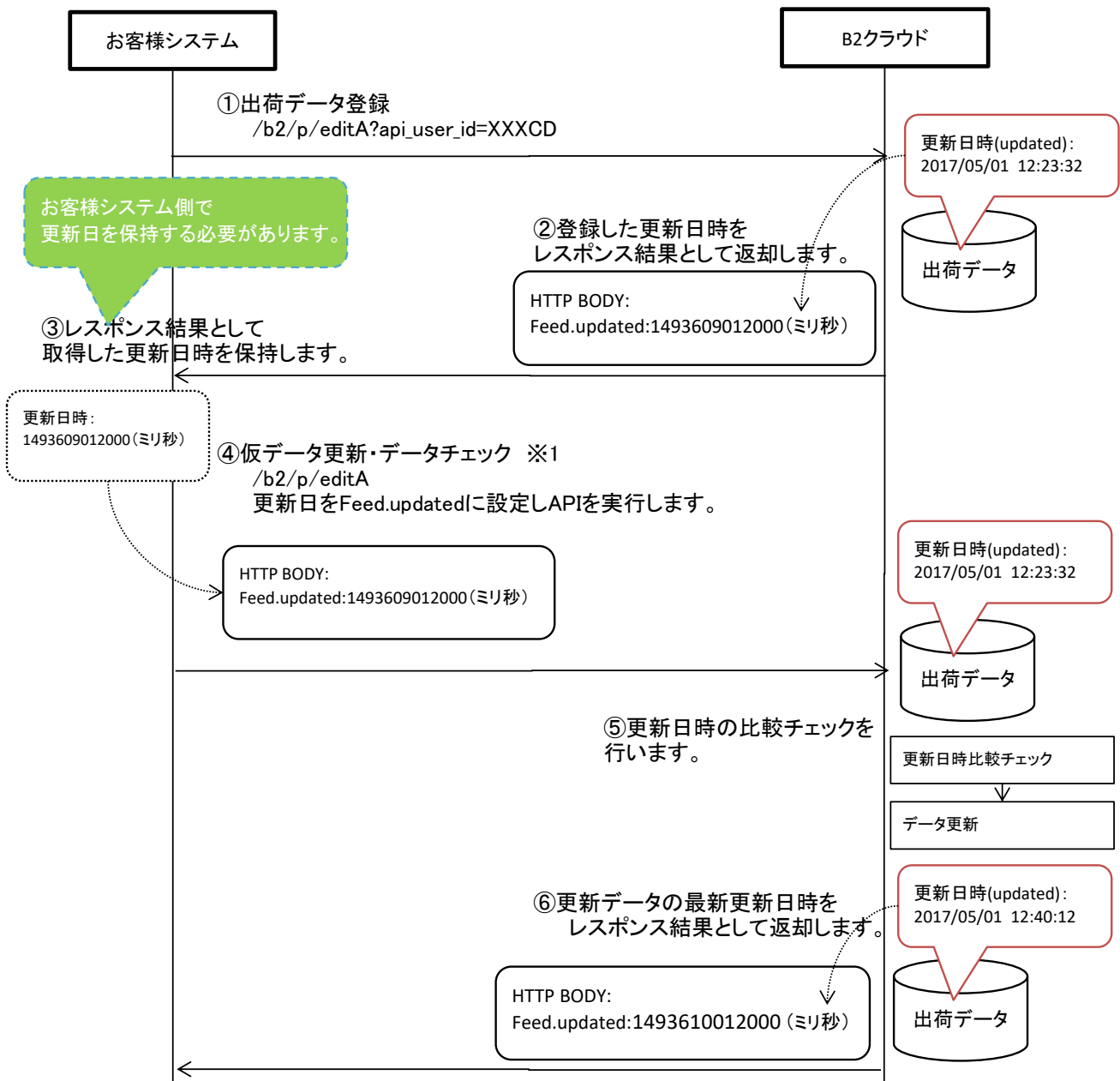
画面API利用なし

B2クラウドAPIではデータ整合性の確保するため、排他制御を行うことを必須としています。
排他制御処理が必須となるAPIは以下となります。
本項では、排他制御の仕組みとインターフェイスについて説明します。

- 仮データ更新・データチェック
- 仮データ削除
- 送り状発行
- 緊急時出荷登録

排他制御の仕組みとフロー

1-1 通常排他制御フロー

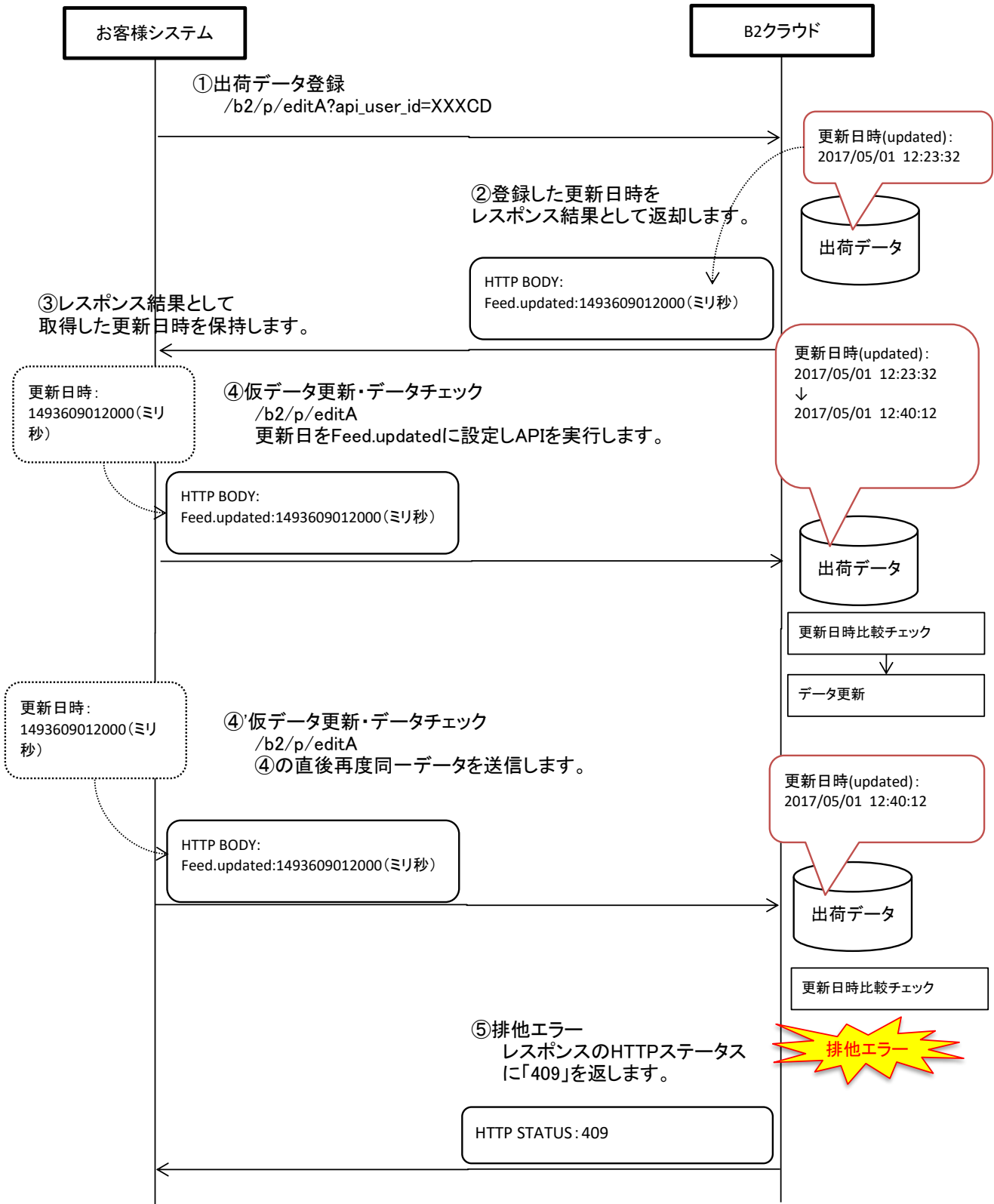


※1

上記の例では仮データ更新処理同様、下記4つのAPIで処理フローは同じになります。

- 仮データ更新・データチェック
- 仮データ削除
- 送り状発行
- 緊急時出荷登録

1-2 排他エラーとなった場合のフロー



○ 排他エラーとなった場合の対処方法について

排他エラーとなった場合は、「仮データ取得」APIを実行し、最新の更新日時を取得してください。
「仮データ取得」の取得の詳細は、「7-1.APIインターフェース」参照。

排他制御の共通インターフェイス

リクエスト
feed内のupdatedフィールドにAPIで取得した更新日を設定します。

項目		説明
feed		
	updated	お客様システムで送信時に、レスポンス結果で取得した最終更新日を設定する。
	entry□	
	shipment	

レスポンス
実行結果が正常(HTTPステータス:200)の場合、feed内のupdatedに最新の更新日が設定されます。

項目		説明
feed		
	updated	B2クラウド側で最新更新日が設定する。(次のリクエストに利用する。)
	entry□	
	shipment	

API一覧と更新日設定の関係

No	API名	更新日設定	
		リクエスト	レスポンス
1	仮データ登録・データチェック		○
2	仮データ取得		○
3	仮データ更新・データチェック	○	○
4	仮データ削除	○	○
5	送り状発行	○	○
6	印刷状態確認		
7	PDFダウンロード		
8	伝票番号取得		

8.HTTPステータスについて

画面API利用あり

画面API利用なし

※RESTについて

B2クラウドは、REST(Representational State Transfer)の概念に基づき設計されており、リソースを一意に特定するURLに対してCRUD操作をします。

当インターフェースの通信プロトコルはHTTPSで、利用できる操作はGET、POST、PUTになります。

GETが成功の場合、レスポンスとして以下の内容が返却されます。

HTTP Status	200
ボディ部	応答データ

POSTが成功の場合、レスポンスとして以下の内容が返却されます。

HTTP Status	200
ボディ部	なし、または、応答データ

PUTが成功の場合、レスポンスとして以下の内容が返却されます。

HTTP Status	200
ボディ部	応答データ

リクエストが不成功の場合、レスポンスとして以下の内容が返却されます。

HTTP Status	200以外(HTTPリターンコード)
feed.title	エラーメッセージ
feed.subtitle	エラーコードなど(ない場合も有る)

※エラー発生時のFeedの戻り方について

通常の処理ではFeedにデータが格納された状態で返却されますが、その他例外が発生した場合などの返却処理は以下の通りとなります。

・出荷データチェックなどの業務警告・エラーの場合

```
feed
└─"entry"
  └─"shipment"
    └─"agent_id"
      └─"1234567890"
    └─"amount"
      └─"0"
    └─"error_flg"
      └─"1"
    └─"..."
  └─"error"
    └─"error_code"
      └─"EF011025"
    └─"error_description"
      └─"収納代行支払期限日が入力されていません"
    └─"error_property_name"
      └─"agent_duedate"
  └─"title"
    └─"RXID (仮データ登録時のみ 設定されます。)"
```

業務警告・エラーなどの場合はshipmentなどのデータも返却されます。

業務警告・エラーなどの場合に設定されます。詳細は、「9.出荷データ仕様」を参照。

E始まりが「エラー」W始まりが「警告」

警告・エラー情報はFeed直下のerror(配列形式)に設定されて返却されます。

HTTP Statusは200 OK で返却されます。

・0件の場合

```
feed
└─"entry"
  └─"subtitle"
    └─"204"
  └─"summary"
    └─"No entry"
  └─"title"
    └─"Error"
```

HTTP Statusは200 OK で返却されます。

・件数制限オーバー

```
feed
└─"entry"
  └─"subtitle"
    └─"416"
  └─"summary"
    └─"Save limit over. limit[300] count:[600] + entrySize:[1]"
  └─"title"
    └─"Error"
```

HTTP Statusは416 Not Acceptable で返却されます。

- ・ 通常のシステム例外

```
feed
└─"entry"
  └─"subtitle"
    └─"500"
  └─"summary"
    └─"Server error."
└─"title"
  └─"Error"
```

9.出荷データ仕様

画面API利用あり 画面API利用なし

○ :必須
(○) :条件付き(説明参照)
空白 :任意

No.	エンティティ		桁数		必須	説明
	エンティティ項目		半角	全角		
-	出荷	shipment				
1	お客様管理番号	shipment_number	50	-		英数字＋記号の一部(-)
2	送り状種類	service_type	1	-	○	0:発払い、2:コレクト、3:クロネコゆうメール、4:タイム、5:着払い、6:発払い(複数口)、7:ネコポス・クロネコゆうパケット、8:宅急便コンパクト、9:コンパクトコレクト
3	クール区分	is_cool	1	-		0:通常、1:クール冷凍、2:クール冷蔵 ※タイムサービス、発払い(複数口)、ネコポス・クロネコゆうパケット、宅急便コンパクト、コンパクトコレクトの場合は"0"を設定してください。 未入力時は"0:通常"固定
4	出荷予定日	shipment_date	8	-	○	YYYYMMDD形式
5	お届け予定日	delivery_date	8	-		YYYYMMDD形式 ※入力なしの場合、印字されません。 最短日を指定する場合、"最短日"を設定。 "最短日"が指定された場合、No75最短日フラグが「1:最短日指定あり」になる
6	コレクト代金引換額	amount	7,0	-	(○)	送り状種類がコレクト、コンパクトコレクトの場合は必須。 下記の通りの金額が設定可能です。 ・コレクトの場合、 300,000円以下 ・コンパクトコレクトの場合、 30,000円以下
7	コレクト内消費税額等	tax_amount	7,0	-	(○)	送り状種類がコレクト、コンパクトコレクトの場合は必須
8	個数口印字フラグ	is_printing_lot	1	-		送り状種類が発払い、タイムの場合は有効。 1:個数口枠のみ印字する 2:個数口枠を印字しない 3:個数口枠と口数を印字する(デフォルト:1設定) ※複数口は指定不可(「3:枠と口数を印字する」固定動作)
9	請求先・顧客コード	invoice_code	12	-	(○)	クロネコゆうメール、着払いの場合、不要 それ以外の送り状種類では必須
10	請求先・分類コード	invoice_code_ext	3	-	(○)	クロネコゆうメール、着払いの場合、不要 それ以外の送り状種類では必須 分類コードがない場合、「'''」を指定。
11	請求先・運賃管理番号	invoice_freight_no	2	-	(○)	クロネコゆうメール、着払いの場合、不要 それ以外の送り状種類では必須
12	請求先・ニックネーム	invoice_name	30	15		No.9~12に紐づく顧客情報の請求先・ニックネーム
13	クロネコwebコレクトデータ登録	payment_flg	1	-		送り状種類が発払いの場合は有効。 0:決済サービスを利用しない、1:クロネコwebコレクト
14	クロネコwebコレクト加盟店番号	payment_number	9	-	(○)	クロネコwebコレクトデータ登録が「1:クロネコwebコレクト」の場合、必須 ※請求先に紐づく加盟店番号
15	クロネコwebコレクト申込受付番号1	payment_receipt_no1	23	-	(○)	クロネコwebコレクトデータ登録が「1:クロネコwebコレクト」の場合、必須
16	クロネコwebコレクト申込受付番号2	payment_receipt_no2	23	-		発払い(複数口)の場合、設定不可
17	クロネコwebコレクト申込受付番号3	payment_receipt_no3	23	-		発払い(複数口)の場合、設定不可
18	複数口くくりキー	closure_key	20	10		送り状種類6:発払い(複数口)を選択時、お届け日時とお届け住所が同じ荷物に対し、同一のくくりキーを設定 伝票番号取得時に親伝票番号(先頭出荷データ)で置き換わる同一くくりキーが設定された出荷データについての入力チェックについては別紙「9-2.発払い(複数口)の入力チェックについて」を参照
19	検索キータイトル1	search_key_title1	20	10		
20	検索キー1	search_key1	20	10		
21	検索キータイトル2	search_key_title2	20	10		
22	検索キー2	search_key2	20	10		
23	検索キータイトル3	search_key_title3	20	10		
24	検索キー3	search_key3	20	10		
25	検索キータイトル4	search_key_title4	20	10		
26	検索キー4	search_key4	20	10		

No.	エンティティ	物理名	桁数		必須	説明
	エンティティ項目		半角	全角		
27	日付印字フラグ	is_printing_date	1	-		0: 伝票に日付を印字しない 1: 伝票に日付を印字する (出荷予定日、お届け予定(指定)日) ・DM便、ネコポスは0固定 ・その他の送り状種類はお届け予定日の入力により、デフォルト設定を行う。 お届け予定日の入力あり:「1: 伝票に日付を印字する (出荷予定日、お届け予定(指定)日)」 お届け予定日の入力なし:「0: 伝票に日付を印字しない」
28	登録元システム種別	input_system_type	3	-		固定値“api”を設定
29	出荷予定個数口	package_qty	2	-		送り状種類が発払い、タイム、着払い、ネコポス・クロネコゆうパケット、宅急便コンパクトの場合、「1」～「99」を設定可能。 複数口は同一くくりキー内で「2」～「99」となるように設定。 それ以外の場合は「1」 未入力時は「1」固定
30	配達時間帯区分	delivery_time_zone	4	-		【発払・コレクト・着払・発払い(複数口)・宅急便コンパクト・コンパクトコレクト】 空白 or 0000: 指定なし 0812: 午前中 1416: 14時～16時 1618: 16時～18時 1820: 18時～20時 1921: 19時～21時 【タイム】 0010: 午前10時まで 0017: 午後5時まで 未入力時は「0000: 指定なし」固定 ※送り状種類がタイムの場合は「0010: 午前10時まで」固定
31	お届け予定メール利用フラグ	is_using_shipment_email	1	-		0: 利用しない、1: 利用する ※送り状種類: クロネコゆうメール、タイム、発払い(複数口)、ネコポス・クロネコゆうパケットは「0: 利用しない」固定 未入力時は「0: 利用しない」固定
32	お届け予定e-mailアドレス	shipment_email_address	60	-	(○)	半角 お届け予定メール利用フラグが「1: 利用する」の場合、必須
33	お届け予定入力機種種別	input_device_type	1	-		1: パソコン、2: 携帯
34	お届け予定メッセージ	shipment_message	148	74	(○)	全角 お届け予定メール利用フラグが「1: 利用する」の場合、必須
35	お届け完了メール利用フラグ	is_using_delivery_email	1	-	○	0: 利用しない、1: 利用する ※発払い(複数口)は「0: 利用しない」固定
36	お届け完了e-mailアドレス	delivery_email_address	60	-	(○)	半角 お届け予定メール利用フラグが「1: 利用する」の場合、必須
37	お届け完了メッセージ	delivery_message	318	159	(○)	全角 お届け予定メール利用フラグが「1: 利用する」の場合、必須
38						入力不要
39	プリンタ種別	printer_type	1	-		印刷するプリンタのタイプを指定する 1: レーザー、2: インクジェット、3: ラベルプリンタ 設定がない場合、B2クラウドの各種設定情報で設定されているプリンタになる。
40	ご依頼主・コード	shipper_code	20	-		
41	ご依頼主・電話番号(表示用)	shipper_telephone_display	15	-	(○)	ハイフンを含む クロネコゆうメール以外は必須
42	ご依頼主・電話番号枝番	shipper_telephone_ext	2	-		半角
43	ご依頼主・名	shipper_name	32	16	(○)	全角 クロネコゆうメール以外は必須
44	ご依頼主・敬称	shipper_title	4	2		全角 設定不要 クロネコゆうメール以外は「様」固定 クロネコゆうメールは空白
45	ご依頼主・郵便番号	shipper_zip_code	7	-	(○)	半角(ハイフンを含む) ハイフンなし半角7文字も可 クロネコゆうメール以外は必須
46	ご依頼主・入力住所	shipper_address	96	48		クロネコゆうメールの場合は設定不要。 ※住所情報をまとめる場合は、32文字/64文字以内で都道府県・市区郡町村・町番地までを【shipper_address】に入力する。 (【shipper_address1～3】を指定しない。 【shipper_address】、【shipper_address1～3】を指定した場合は、【shipper_address】を優先して設定する。) 尚、【shipper_address】に設定する場合もアパートマンション名は、【shipper_address4】に入力する

No.	エンティティ	物理名	桁数		必須	説明
	エンティティ項目		半角	全角		
47	ご依頼主・都道府県名	shipper_address1	8	4		全角/半角 クロネコゆうメールの場合は設定不要。
48	ご依頼主・市区郡町村名	shipper_address2	24	12	(○)	全角/半角 クロネコゆうメールの場合は設定不要。 No.46「ご依頼主・入力住所」が未入力の場合は必須
49	ご依頼主・番地	shipper_address3	32	16	(○)	全角/半角 クロネコゆうメールの場合は設定不要。 No.46「ご依頼主・入力住所」が未入力の場合は必須
50	ご依頼主・建物名	shipper_address4	32	16		全角/半角 クロネコゆうメールの場合は設定不要。
51	ご依頼主・略称カナ	shipper_name_kana	400 50	50		半角カナ
52	お届け先・コード	consignee_code	20	40		半角・半角カナ
53	お届け先・電話番号(表示用)	consignee_telephone_display	15	-	(○)	ハイフンを含む クロネコゆうメール以外は必須
54	お届け先・電話番号枝番	consignee_telephone_ext	2	-		半角
55	お届け先・名	consignee_name	32	16	○	全角
56	お届け先・郵便番号	consignee_zip_code	7	-	○	半角(ハイフンを含む) ハイフンなし半角7文字も可
57	お届け先・入力住所	consignee_address	96	48		※住所情報をまとめる場合は、32文字/64文字以内で都道府県・市区郡町村・町番地までを【consignee_address】に入力する。 (【consignee_address1～3】を指定しない。 【consignee_address】、【consignee_address1～3】を指定した場合は、【consignee_address】を優先して設定する。) 尚、【consignee_address】に設定する場合もアパートマンション名は、【consignee_address4】に入力する
58	お届け先・都道府県名	consignee_address1	8	4		全角/半角
59	お届け先・市区郡町村名	consignee_address2	24	12	(○)	全角/半角 No.57「お届け先・入力住所」が未入力の場合は必須
60	お届け先・番地	consignee_address3	32	16	(○)	全角/半角 No.57「お届け先・入力住所」が未入力の場合は必須
61	お届け先・建物名	consignee_address4	32	16		全角/半角
62	お届け先・部門名1	consignee_department1	50	25		全角
63	お届け先・部門名2	consignee_department2	50	25		全角
64	お届け先・略称カナ	consignee_name_kana	400 50	50		半角カナ
65	お届け先・敬称	consignee_title	4	2		クロネコゆうメールのみ、指定が有効になる。 【入力例】 様・御中・殿・行・係・宛・先生・なし
66	止置きサービス利用フラグ	is_using_center_service	1	-		0: 利用しない、1: 利用する 未入力時はデフォルト「0: 利用しない」固定 クロネコゆうメール、ネコボス・クロネコゆうバケットの場合、設定不可 設定されている場合、「0: 利用しない」に変更。
67	止め置き営業所コード	consignee_center_code	6	-	(○)	止置きサービス利用する場合は必須 クロネコゆうメール、ネコボス・クロネコゆうバケットの場合、設定不可 設定されている場合、「0: 利用しない」に変更。
68	品名コード1	item_code1	30	60		半角・半角カナ
69	品名コード2	item_code2	30	60		半角・半角カナ
70	品名称1	item_name1	50	25	(○)	全角 クロネコゆうメール以外は必須

No.	エンティティ	物理名	桁数		必須	説明
	エンティティ項目		半角	全角		
71	品名称2	item_name2	50	25		全角
72	荷扱い1	handling_information1	20	10		全角
73	荷扱い2	handling_information2	20	10		全角
74	記事	note	44	22		全角
75	最短日フラグ	short_delivery_date_flag	1	-		0:最短日指定なし、1:最短日指定あり 送り状種類:「発払い、コレクト、タイム、着払い、発払い(複数口)、宅急便コンパクト、コンパクトコレクト」の場合かつお届け予定日に「最短日」指定がある場合、「1:最短日指定あり」固定
76	投函予定メール利用区分	is_using_shipment_post_email	1	-		0:利用しない、1:利用する ネコポス・クロネコゆうパケット以外は0固定 クロネコゆうパケットの場合、投函予定メールを指定頂いてもメールは送信されません。
77	投函予定メールアドレス	shipment_post_email_address	60	-	(○)	半角 投函予定メール利用区分が「1:利用する」の場合、必須
78	投函予定メール入力機種種別	shipment_post_input_device_type	1	-		1:パソコン、2:携帯(新規作成時:1設定)
79	投函予定メールメッセージ	shipment_post_message	148	74	(○)	全角 投函予定メール利用区分が「1:利用する」の場合、必須
80	投函完了メール(受人宛て)利用区分	is_using_cons_deli_post_email	1	-		0:利用しない、1:利用する クロネコゆうパケットの場合、投函完了メール(お届け先宛)を指定頂いてもメールは送信されません。
81	投函完了メール(受人宛て)アドレス	cons_deli_post_email_address	60	-	(○)	半角 投函完了メール(受人宛て)利用区分が「1:利用する」の場合、必須
82	投函完了メール(受人宛て)入力機種種別	cons_deli_post_input_device_type	1	-		1:パソコン、2:携帯(新規作成時:1設定)
83	投函完了メール(受人宛て)メッセージ	cons_deli_post_message	318	159	(○)	全角 投函完了メール(受人宛て)利用区分が「1:利用する」の場合、必須
84	投函完了メール(出人宛て)利用区分	is_using_shipper_deli_post_email	1	-		0:利用しない、1:利用する(新規作成時:0設定) クロネコゆうパケットの場合、投函完了メール(ご依頼主宛)を指定頂いてもメールは送信されません。
85	投函完了メール(出人宛て)アドレス	shipper_deli_post_email_address	60	-	(○)	半角 投函完了メール(出人宛て)利用区分が「1:利用する」場合、必須
86	投函完了メール(出人宛て)入力機種種別	shipper_deli_post_input_device_type	1	-		1:パソコン、2:携帯(新規作成時:1設定)
87	投函完了メール(出人宛て)メッセージ	shipper_deli_post_message	318	159	(○)	全角 投函完了メール(出人宛て)利用区分が「1:利用する」場合、必須
88	連携管理番号	cooperation_number	70		(○)	半角英数字記号 直送りを利用する場合は必須
89	通知メールアドレス	notification_email_address	60			半角英数字記号
89	エラーフラグ	error_flg	1			レスポンス時に設定される 出荷データチェックの結果 0: 正常、1:警告、9:エラー

9-1.出荷データサンプル(JSON)

画面API利用あり

画面API利用なし

例) ■仮データ登録・データチェックのレスポンスデータ

```
{
  "feed":{
    "title":"20170309213149P09-dU*tXBTJt*x-ghEg6pAkT4c2am5zBdKTnb1X9nOYt2XvAXUxA0*ovZp-!002233445604-001:o2
    "entry":[
      {
        "shipment":{
          "tracking_number":"1",
          "shipment_number":"15",
          "service_type":"7",
          "is_cool":"0",
          "shipment_date":"20160830",
          "delivery_date":"",
          "amount":"0",
          "tax_amount":"0",
          "is_printing_lot":"",
          "invoice_code":"076522056601",
          "invoice_code_ext":"002",
          "invoice_freight_no":"01",
          "invoice_name":"",
          "payment_flg":"",
          "payment_number":"",
          "payment_receipt_no1":"",
          "payment_receipt_no2":"",
          "payment_receipt_no3":"",
          "closure_key":"",
          "search_key_title1":"",
          "search_key1":"",
          "search_key_title2":"",
          "search_key2":"",
          "search_key_title3":"",
          "search_key3":"",
          "search_key_title4":"",
          "search_key4":"",
          "search_key_title5":"",
          "search_key5":"",
          "sorting_code":"",
          "sorting_ab":"",
          "is_printing_date":"0",
          "input_system_type":"api",
          "package_qty":"3",
          "delivery_time_zone":"0000",
          "is_using_shipment_email":"0",
          "shipment_email_address":"",
          "input_device_type":"",
          "shipment_message":"",
          "is_using_delivery_email":"0",
          "delivery_email_address":"",
          "delivery_message":"",
          "is_showing_telephone_on_mail":"0",
          "shipper_code":"アイウエオカキクケコ",
          "shipper_telephone":"724621795",
          "shipper_telephone_display":"724621795",
          "shipper_telephone_ext":"15",
          "shipper_name":"ご依頼主テスト※※※※※※※15",
          "shipper_title":"様",
          "shipper_zip_code":"5980005",
          "shipper_address":"大阪府泉佐野市市場東1丁目250",

```

[illegible]

```

},
"error":[
  {
    "error_property_name":"shipment_date",
    "error_code":"ES003001",
    "error_description":"出荷予定日は本日～30日後までの範囲で指定して下さい。"
  },
  {
    "error_property_name":"invoice_code",
    "error_code":"ES006002",
    "error_description":"請求先が存在しません。"
  }
],
"id":"1",
"link":[
  {
    "__href":"1",
    "__rel":"self"
  }
]
],
{
  "shipment":{
    "tracking_number":"2",
    "shipment_number":"16",
    "service_type":"7",
    "is_cool":"0",
    "shipment_date":"20160830",
    "delivery_date":"",
    "amount":"0",
    "tax_amount":"0",
    "is_printing_lot":"",
    "invoice_code":"076522056601",
    "invoice_code_ext":"002",
    "invoice_freight_no":"01",
    "invoice_name":"",
    "payment_flg":"",
    "payment_number":"",
    "payment_receipt_no1":"",
    "payment_receipt_no2":"",
    "payment_receipt_no3":"",
    "closure_key":"",
    "search_key_title1":"",
    "search_key1":"",
    "search_key_title2":"",
    "search_key2":"",
    "search_key_title3":"",
    "search_key3":"",
    "search_key_title4":"",
    "search_key4":"",
    "search_key_title5":"",
    "search_key5":"",
    "sorting_code":"",
    "sorting_ab":"",
    "is_printing_date":"0",
    "input_system_type":"api",
    "package_qty":"3",
    "delivery_time_zone":"0000",
    "is_using_shipment_email":"0",
    "shipment_email_address":"",
    "input_device_type":"",
    "shipment_message":""
  },

```

[illegible]

```

    "cons_deli_post_message": "",
    "is_using_shipper_deli_post_email": "0",
    "shipper_deli_post_email_address": "",
    "shipper_deli_post_input_device_type": "",
    "shipper_deli_post_message": "",
    "display_flg": "1",
    "reissue_count": "",
    "is_printing_logout": "0",
    "error_flg": "9",
    "is_agent": "0",
    "cooperation_number": "",
    "notification_email_address": ""
  },
  "error": [
    {
      "error_property_name": "shipment_date",
      "error_code": "ES003001",
      "error_description": "出荷予定日は本日～30日後までの範囲で指定して下さい。"
    },
    {
      "error_property_name": "invoice_code",
      "error_code": "ES006002",
      "error_description": "請求先が存在しません。"
    }
  ],
  "id": "2",
  "link": [
    {
      "__href": "2",
      "__rel": "self"
    }
  ]
}
]
}
}
}

```


9-2.発払い(複数口)の入力チェックについて

発払い(複数口)で複数口くりキーに同じ値を設定された場合、出荷データを跨いだ入力チェックを行う。

①発払い(複数口)選択時、同一くり内で以下の項目が1つでも異なる場合、エラーとする。

項目名	備考
出荷予定日	
お届け予定日	
請求先・運賃管理番号	
請求先・顧客コード	
請求先・分類コード	
配達時間帯区分	
お届け先・電話番号(表示用)	
お届け先・電話番号枝番	
お届け先・名	
お届け先・郵便番号	
お届け先・入力住所	
お届け先・都道府県名	お届け先・入力住所が指定された場合は対象外
お届け先・市区郡町村名	お届け先・入力住所が指定された場合は対象外
お届け先・番地	お届け先・入力住所が指定された場合は対象外
お届け先・建物名	
お届け先・部門名1	
お届け先・部門名2	
営業所コード	

②出荷予定個数口は同一くりキー内の合計が2個から99個でなければ、エラーとする。

③クロネコwebコレクトの指定は先頭の出荷データのみで指定可能とする。
同一くりキー内で複数の指定がある場合、エラーとする。

10.制限事項

画面API利用あり

画面API利用なし

10_1.出荷データ登録件数について

出荷データの登録時に一度に実行できる件数は最大1000件となります。

画面API利用あり

画面API利用なし

10_2.メンテナンス時間について

定期 .. 毎日(25:00~7:00)

画面API利用あり

画面API利用なし

10_3.出荷データの保存期間と情報取得可能期間について

出荷データ(/spool)の参照可能時間は発行後5分間となります。

画面API利用あり

10_4.セッションID(SID)の有効期間は、最後の操作を行ってから60分間有効です。

画面API利用あり

画面API利用なし

10_5.ワンタイムパスワード(RXID)の利用は1回限りで発行後60分間有効です。

画面API利用あり

10_6.API検証環境をご利用の場合、発行される伝票はサンプルPDFとなります。 お客様のシステムで入力された情報は反映されません。

画面API利用あり

画面API利用なし

10_7.ダウンロードした送り状PDFは、取り決めた時間有効です。 API検証環境をご利用の場合、有効時間はございません。

画面API利用なし

10_8.推奨環境

画面API利用あり

画面API利用なし

・OS/ブラウザの推奨環境

日本語版オペレーティングシステム

<Windowsをご利用の場合>

OS/Windows10・11

ブラウザ/Edge chromium版、Google Chrome

<Macintoshをご利用の場合>

macOS 13.2 ブラウザ/Google Chrome 最新版

11.免責事項

画面API利用あり

画面API利用なし

1.当社のインターフェースのご利用について

このデータ交換規約は当社の提供するインターフェースに関する規約です。

なお、お客様がこのインターフェースを利用して自ら開発したシステムについてはお客様が一切の責任を負うことになります。

2.利用上の制約について

当社は、お客様への通知を行うことなく、自己の裁量に基づいていつでもソフトウェアや特定の機能に関して利用の制約(たとえば、ソフトウェアを介して当社のサービスへのアクセス回数、アクセス時間などについて制限することなど)を行うことができるものとします。

また、ソフトウェアに関するサポートや修正版(アップデート版を含みます)の提供は、当社の裁量に基づいて行い、またあらかじめお客様へ通知を行うことなく、それらを**終了する場合があります。**

3.禁止行為について

本サービスのご利用に際し、次に掲げる行為を禁止します。

- ①インターフェースを販売するなど営利目的で利用する行為
- ②不正にアクセスする行為
- ③不正なデータを送受信する行為
- ④管理及び運営を妨害する行為
- ⑤改変、改造、再配布する行為
- ⑥公序良俗に反するサイトで利用する行為
- ⑦その他当社が不適切と判断する行為

4.保証について

当社は提供するインターフェースについて、エラーやバグ、論理的誤り、不具合、中断その他の瑕疵(かし)がないこと、信頼性、正確性、完全性、有効性について一切保証しておりません。インターフェースは明示または黙示の有無にかかわらず、当社がその提供時において保有する状態で提供するものであり、特定の目的への適合性、有用性(有益性)、セキュリティ、権原および非侵害性について一切保証しておりません。

本ソフトウェアをご使用になりますと、お客様は上記事項に同意されたものとみなされます。