



BILINFO BRANDS AND PRODUCTS

Third party integration document

Abstract

Brands and Products are a required element of both a Case Plugin and Finance Offer On Platforms integration. This document describes the prerequisites and details for building a Brands and Products API for integrating into Bilinfo.

Version history

Version	Date	Authors	Comments
0.1.0	10/09/2018	Jacob Sønderskov (jsonderskov@ebay.com)	First draft version based on the <i>Bilinfo integration using the Case Plugin Architecture (2.4.0)</i> documentation.
1.0.0	12/09/2018	Jacob Sønderskov	First release version.
2.0.0	12/09/2018	Jacob Sønderskov	Document version aligned with current major version of Brands and Products API specification.
2.0.1	13/09/2018	Jacob Sønderskov	Change cover page and add abstract.

Contents

Version history.....	1
A. Disclaimer	3
B. Versioning and Deprecation Policy.....	4
1. Introduction.....	6
1.1 Purpose and Scope	6
1.2 References	6
1.3 Definitions and acronyms.....	6
2. Prerequisites.....	7
2.1 Security	7
2.2 Bilinfo Shared Services	7
3. Partner Server-Side.....	8
3.1 Brands & Products API.....	8
3.2 Recommended: Brands & Products Management System	12

A. Disclaimer

Information presented here might be altered by eBay from time to time. Inconsistencies across the document are to be expected and they will be addressed in updates.

Any update will be specified in Version history.

Data and system integrity

Abuse of the system is forbidden in any regard. If you find a security issue or exploitation outside the original intent of the system, you are expected to report the exploit or bug to the Bilinfo team.

B. Versioning and Deprecation Policy

Versioning in Bilinfo Services is essential to achieving our vision behind Partner integrations in Bilinfo. Using the versioning principles described below will allow for your Bilinfo integrations to remain stable and fully functional as the Bilinfo business continues to evolve and mature.

New Versions of the Bilinfo Services

The versioning principles employed in Bilinfo Services largely follow that of the *Semantic Versioning Specification*^[1]. The Semantic Versioning Specification, in short, specifies a version increment based on the backwards compatibility of the API or Web Service. A summary of the specification can be seen in the following Listing B-1:

Given a version number MAJOR.MINOR.PATCH, increment the:

1. MAJOR version when you make incompatible API changes,
2. MINOR version when you add functionality in a backwards-compatible manner, and
3. PATCH version when you make backwards-compatible bug fixes.

Additional labels for pre-release and build metadata are available as extensions to the MAJOR.MINOR.PATCH format.

Listing B-1: Semantic Versioning Specification 2.0.0 summary

The types of changes that are minor version changes and backward compatible are:

- Adding a new method (GET, POST etc.) to an API
- Adding a new property to the method response payload
- Adding a new *non-personal data*^[2] property to an `iframe` communication

The types of changes that are major version changes and not backward compatible are:

- Removing existing method (GET, POST etc.) from an API
- Renaming existing method path
- Changing request body or query string for existing method
- Changing method response structure and/or property names
- Removing a property from an `iframe` communication
- Renaming a property from an `iframe` communication
- Renaming a message in an `iframe` communication
- Adding a new *personal data*^[2] property to an `iframe` communication

¹ <https://semver.org>

² Personal data as defined in Regulation (EU) 2016/679 of 27 April 2016 (GDPR) and the Danish Data Protection Act.

In general, new major versions of Bilinfo Services will only be introduced, when existing interfaces does not allow for further evolution and improving of our Partner integrations without modification. Due to the need for Partner action, major versions are used as a last resort and are as such very rare. Minor version updates will require no Partner action.

Updating your Bilinfo integration

Updating your Bilinfo integration to support a new major version is non-optional as the existing integration paradigm is fundamentally changed. It is as such not possible to opt out without risking major problems with your Bilinfo integration. Minor versions, however, are fully optional, but may contain new fields, which may enrich the experience and value of your Bilinfo integration.

To assist Partners in upgrading their Bilinfo integration with minimal efforts, each major version will be associated with a *migration chapters* added to this document. Minor version changes are specified primarily in the Version History and is subject to the reader to adhere to the changes.

Deprecation Policy and Supported Versions

Bilinfo Services will support older versions for a grace period appropriate to the contractual obligations. After that time, integrations based on older versions may no longer work or experience severe operational issues.

1. Introduction

1.1 Purpose and Scope

This document will present the technical specifications for implementing a Brands and Products API for use in *Case Plugin* and/or *Finance Offer On Platforms* integrations into Bilinfo.net.

For more information pertaining to the integrations see the [Bilinfo Case Plugin \(Integration\)](#) and [Bilinfo Finance Offer On Platforms \(integration\)](#) documents, respectively.

1.2 References

Documents relevant to the reading of this document are listed here. Links and other external resources accessible via the internet are referenced via footnotes relative to the term or technology. You should have access to every document mentioned in this list. If that is not the case, contact Bilinfo.

Table 1-1: Document references

Document name	Description	Link
Bilinfo Auth Services (Integration)	Describes how to integrate into the Bilinfo Auth Services, SSO and OAuth2.0, needed to respectively authenticate and authorize access to Bilinfo.net.	TBA
Bilinfo Shared Services (Integration)	Describes a number of services made available to integrating party, enabling access to additional information to e.g. Dealers.	TBA
Bilinfo Case Plugin (Integration)	Describes the integration details for a Case Plugin integration into Bilinfo.net	TBA
Bilinfo Finance Offer On Platforms (integration)	Contains the specification for implementing a Finance Offer On Platforms integration to Bilinfo.net	TBA

1.3 Definitions and acronyms

The definitions and acronyms defined in Table 1-2 cover frequently used concepts, terms and acronyms used throughout this document. It is suggested that the reader acquaints him- or herself with the key concepts and refer to this list, when in doubt.

Table 1-2: Definitions and acronyms

Term/acronym	Definition
Bilinfo Services	Includes – but not limited to – Bilinfo Auth Services, Bilinfo Shared Services, Bilinfo Finance Offer On Platform integrations and Bilinfo Case Plugin integrations.
Partner	Synonym for the integrating party.
Case Plugin	Iframe integration in Bilinfo.net’s Case system allowing Finance and Insurance offer handling.
Finance Offer On Platforms	Infrastructure component which enables managing Finance Banners in Bilbasen and DBA platforms via Bilinfo.net. See
Bilinfo Data Services	Aggregate term for API and Data Services provided by Bilinfo to external Partners and integrations.

2. Prerequisites

This chapter describes the preliminary information and tasks you should get under way before starting any development. These tasks are dependent on external parties and may take some time to complete.

2.1 Security

All Bilinfo service API's use SSL/TLS (HTTPS) and OAuth 2.0 for authorization. It is expected that the Partner API's follow the same standards for Transport Layer Security and equivalent for Authorization (at least Basic authentication).

2.2 Bilinfo Shared Services

The Brands & Products API depends on Dealer identifiers relative to the Bilinfo.net context. These may be acquired through the Bilinfo Shared Services. To access the Bilinfo Shared Services, client credentials with proper authorization scopes must be acquired. For more on this refer to the [Bilinfo Auth Services](#) (Integration) document referenced in section [1.2 References](#).

3. Partner Server-Side

This chapter will cover the Partner Server-Side components that must be implemented in order to enable a Case Plugin or Finance Offer On Platforms integration.

3.1 Brands & Products API

The *Brands & Products API* is a primary component in both a Case Plugin or Finance Offer On Platforms integration as it enables Partner-controlled user selections on Bilinfo.net.

Upon user interaction with integration touch points, Bilinfo.net will poll Brands and Products from the Partner and present them to the user. It is important, that the integrating keeps this in mind as the same list will be used in different areas of Bilinfo.net

The API is a shared entity and as such contain fields that are relevant to specific integrations. These are marked in the following specification for the *expected* Input and Output of the API.

The Brands & Products API must provide a single HTTP GET endpoint.

3.1.1 Input

Table 3-1 shows the input fields required for the Brands & Products API. The input fields should be accepted as query strings.

Table 3-1: Brands and Products API interface input

Field	Type	Description
dealerId	string	The Bilinfo dealer identifier, ex: "42304136-fdca-df11-9e98-0025b3e6b7d8". Refer to the Bilinfo Shared Services (Integration) document for further details regarding Dealer mapping.
userId	string	The Bilinfo user identifier, ex: "10420". Refer to the Bilinfo Shared Services (Integration) document for further details regarding User mapping.

3.1.2 Output

Table 3-2 shows the top-most part of the API return message. The output should be in JSON format. Note, that the response is divided into separate sub-types (or models), each of which must fit their respective specification defined in Table 3-3 and Table 3-4, respectively.

Table 3-2: Brands and Products API output top-level

Field	Type	Description
brands	array	Array with elements of type Brand
version	string	The version of the integration document that this api is implementing. Formally this is "2.0.0"

Brand

Table 3-3 describes the fields required for a Brand.

Table 3-3: Brands and Products API output Brand member specification

Field	Type	Description
id	number	Brand identifier. At the moment only 32-bit integers are supported
name	string	Brand name
logoUrl	string	Url to a 133x35 brand logo
allowFinanceOfferOnMarketplaces	boolean	Must be set to TRUE if Brand and Products are used for a Finance Offer On Platforms integration.
products	array	Array with elements of type Product

Product

Table 3-4 describes the fields required for a Product.

Table 3-4: Brands and Products API output Product member specification

Field	Type	Description
id	string	The id of the product
name	string	The name of the product
isPrivateAllowed	boolean	Whether this product is allowed for private persons
isBusinessAllowed	boolean	Whether this product is allowed for businesses
isDefault	boolean	Indicates that this Product may be applied as a <i>default</i> finance offer be set to TRUE if Brand and Products are used for a Finance Offer On Platforms integration.
offeredServices	object	An object of type Service mapping the services that this product offers
pluginUrl	string	Specifies a URI to a webpage containing a Case Plugin form. This is only relevant for a Case Plugin integration

Service

Service is a Dictionary object that maps the offered services. Each map in the object should follow the convention listed below, where the key represents the [ServiceType](#) (e.g. “loan”, “leasing”, ...), and the value describes the offered service.

Note, that it is not recommended to mix different types of services within the same Brand as Brands are sorted into categories based on their Products’ ServiceType in the [Case Plugin](#) integration and filtered in a [Finance Offer On Platforms](#) integration.

It is allowed to specify multiple [ServiceType](#) on a single Product, creating an “all inclusive” product. It is however not allowed to combine the “Loan” and “Leasing” types as they are mutually exclusive, i.e. a finance offer can’t both include a loan and a leasing agreement.

Listing 3-1 shows an example of a pure Loan product and an “all inclusive” product.

```

{
  "brands": [
    {
      "id": 1,
      "name": "Loan Brand",
      ...
      "products": [
        {
          "id": "1",
          "name": "Test Loan",
          ...
          "offeredServices": {
            "loan": {
              "providerName": "Loan Finance Company"
            }
          }
        }
      ]
    },
    {
      "id": 2,
      "name": "Leasing brand",
      ...
      "products": [
        {
          "id": "1",
          "name": "Test Leasing",
          ...
          "offeredServices": {
            "leasing": {
              "providerName": "Leasing Finance Company"
            },
            "insurance": {
              "providerName": "Insurance Company"
            }
          }
        }
      ]
    }
  ]
}

```

Listing 3-1: One Brand with one Service type. Another wise "all inclusive"

ServiceType

Table 3-5 describes the possible values for ServiceType

Table 3-5: ServiceType member specification

Value	Type	Description
loan	string	Loan

leasing	string	Leasing
insurance	string	Insurance
roadSideAssistance	string	Roadside assistance

Listing 3-2 shows an example of a full Brands and Products response, listing one brand that contain two products, one being pure Loan, and the second being a bundle of Loan and Insurance. Please keep in mind that all possible services offered by the Plugin should be listed in the response.

```

{
  "brands": [
    {
      "id": 1,
      "name": "Finance company Loan",
      "logoUrl": "https://f.dk/logo.png",
      "allowFinanceOfferOnMarketplaces": true,
      "products": [
        {
          "id": "1",
          "name": "LoanContract",
          "isPrivateAllowed": true,
          "isBusinessAllowed": true,
          "isDefault": true,
          "pluginUrl": "https://f.dk/plugin/",
          "offeredServices": {
            "loan": {
              "providerName": "Loan Finance Company"
            }
          }
        },
        {
          "id": "2",
          "name": "LoanContract All Inclusive",
          "isPrivateAllowed": true,
          "isBusinessAllowed": true,
          "isDefault": true,
          "pluginUrl": "https://f.dk/plugin/",
          "offeredServices": {
            "loan": {
              "providerName": "Loan Finance Company"
            },
            "insurance": {
              "providerName": "Insurance Company"
            }
          }
        }
      ]
    }
  ]
}

```

Listing 3-2: Brands and Products JSON example

3.2 Recommended: Brands & Products Management System

It is suggested that the Partner implement a management system for the Brands and Products provided via the Brands & Products API.

The management system should accommodate the underlying integration, i.e. [Case Plugin](#) or [Finance Offer On Platforms](#).

Each call to the Brands & Products API is done with a `dealerId` and a `userId` relative to Bilinfo.net. In building the management solution, integrating towards the Dealer Lookup Service specified in the [Bilinfo Shared Services \(Integration\)](#) document will enable you to correlate Dealer CVR numbers with Bilinfo identifiers. Additionally, integrating towards the User Lookup Service will enable you to map specific rules for specific sales persons.