

Inferapp

IT Discovery Service

Specification for Software Detection Results

VERSION 1.3

Revision Date: 18 February 2015

Author: Artur Kornatowski

Table of Contents

- 1. Publisher 3
- 2. Product..... 4
- 3. Version 5
- 4. Discovery Rules 6

1. Publisher

Publisher has PublisherID, PublisherName and Webpage fields.

PublisherName is the unique current name of the owner/developer of the applications. It can usually be found in the Webpage's footnote. Incorporation affixes (Inc., Corp., Company, Ltd., GmbH etc.) are omitted in when possible in PublisherName to help avoid duplicates.

When PublisherName changes, it is also changed in the library and retains the same PublisherID.

When a publisher is acquired by another, all of its products are moved to the new publisher, as explained in the Product section, and the publisher name is affixed with "acquired by *new publisher*". The old publisher then holds no products.

Webpage is a unique domain name prefixed with "www" even if the prefix is not used by the website.

Consistency in Webpage format is important in that it helps to avoid publisher duplicates, which we also seek to do in PublisherName.

2. Product

Product has ProductID, PublisherID, ProductName, Category and Licensable fields.

ProductName represents the publisher-wide unique current name of the software application as listed on the Publisher's website. The PublisherName prefix is omitted when possible.

ProductName may change repeatedly over a product's lifetime. In our approach, all product versions, including older ones, are reported under the current product name. It may be argued that this falsifies the software detection results of older versions. However, it is the lesser evil because otherwise different versions of same product would be scattered over different library products, which would prevent intuitive reporting of all product versions and make maintenance of downgrade logic more difficult.

The software product may also change hands. It is then reassigned to a new PublisherID and retains the same ProductID. ProductName is often prefixed by the old publisher name, or part of it, depending on how the new publisher presents the acquired product on its web site.

There can be only one ProductName assigned to a given Publisher.

Category describes the application's major functionality areas including business reporting, DBMS, operating system, etc.

The Licensable yes/no field is determined by its use in the commercial environment. Only products that can (or could, if discontinued) actually be bought are marked Licensable. In other words, children applications of licensable applications are marked as not licensable unless they can be bought separately. For example, Microsoft Office and Microsoft Word are both marked as licensable because they can be bought separately, even if Word is usually a child application of Microsoft Office.

3. Version

Version has VersionID, ProductID, UniqueVersion, Build, Major, Minor, Edition, Variation and LicenseVersion fields.

UniqueVersion: This field defines product-wide uniqueness of a version, mostly for signature management purposes, and by default consists of Build + Edition.

Build: Value is usually found in ProductVersion fields of files/addremoves/pkginsts, which virtually never contains edition information. If it does, it is moved to separate Edition field.

Major: The character(s) of Build up to the first dot or whitespace, if any.

Minor: The character(s) of Build from the second dot/whitespace up to the next dot/whitespace, if any.

Edition: This value is usually found either in addremove's DisplayName or application installation path or deduced from some other source.

Variation: The marketing name of the version, which is usually just another name for Major.

LicenseVersion: This field is maintained on the version resolution of purchasing data (SKUs) for quick license mapping, i.e., it groups all versions on the same licensing level and eliminates the need to use wildcards on version for license reconciliation purposes. If Variation is empty, LicenseVersion usually equals Major + Edition. If Variation is not empty, LicenseVersion equals Variation + Edition. Occasionally Minor has to be used when a new license has to be purchased to upgrade to a higher Minor.

4. Discovery Rules

For reporting purposes, discovery rule has two fields, Source and Key, which combined with DetectionPath for file source, uniquely identify the element that has triggered the discovery of each application version on each machine.

Version can be discovered via different sources, the most important of which are files, addremoves and pkginsts. There is also a hybrid collection source which relies on application-specific logic, e.g., by querying databases or proprietary application APIs.

Key is defined as file name for file source, display name for addremove source and pkginst name for pkginst source. ProductName is the key for the hybrid source.