# The Donatus XML-RPC Interface
# (Donatus version 1.9)

Malcolm D. Hyman

April 7, 2004

## 1 Introduction

**The Donatus system** provides a unified frontend to a variety of morphological analysis software and databases. Morphological services are provided both through a Remote Procedure Call (RPC) interface that can be utilized by specialized user applications and through a CGI interface that is accessible in any web browser. Morphological data can be represented in XML, allowing them to be cached on client systems and to be processed by a wide range of software. In addition to providing access to pre-existing linguistic data, Donatus allows for the dynamic extension of morphological datasets by a user.

Donatus is intended to be one component in an evolving framework for network-accessible linguistic services. These services may be described as **linguistic middleware;** the idea is to provide a simple but powerful interface layer that will allow linguistic tools and data sources to be employed in end-user applications, such as browsers and editors (these we describe generally as **user agents**). Other services in this framework will provide segmentation or tokenization of natural language data and orthographic normalization.

## 2 The Donatus Architecture

Donatus comprises (1) a plugin architecture for integrating **backend** tools and data sources, using a simple programmatic interface (this interface will be described in a separate document); (2) a **hub** that invokes the appropriate backend plugins, passes linguistic data for analysis, and integrates data from the backends into a unified XML document that may be passed to user agents; and (3) public interfaces to the hub, including a CGI interface designed for direct human use, and an XML-RPC interface designed for programmatic use. The XML-RPC interface is described in this document.

Donatus employs a two-tiered lookup strategy for morphological analysis. First, forms are delegated for analysis to a backend that supports the appropriate language. If the backend fails to identify the form, a **fallback** database is queried. The fallback system allows new lemma/form/analysis triples to be added to Donatus through a single interface, with no need to extend the backend systems. This functionality is especially important, because backends may be proprietary or legacy systems that cannot be extended by users.

## 3 Donatus Servers

The standard Harvard Donatus server for the Archimedes Project is located at

```
http://archimedes.fas.harvard.edu/cgi-bin/donatus-rpc
```

Other servers may be set up for use by other projects and groups of users.

## 4 XML-RPC

**XML-RPC** is a specification that allows programs written in virtually any language to call procedures on remote machines that are connected in a network. XML-RPC is designed as a lightweight solution and uses XML to encode its procedure/method calls and responses. Information on XML-RPC is available at `http://www.xmlrpc.com/`. Open-source implementations are available for many programming languages, including C, Java, Lisp, Perl, Python, and Ruby.

## 5 API Methods

The current version of the Donatus XML-RPC API defines two methods:

### `donatus.analyze(`***`wtag-data`***`)`

Here ***`wtag-data`*** is a simplified XML format used to abstract the linguistic content of a document from its structure. Words are pretokenized and normalized in this format. The format is described in the next section of this document. The ***`wtag-data`*** must be base 64 encoded and passed to Donatus as a parameter of the XML-RPC datatype **base64.**

This method returns a **struct** with three members:

1. ***`morphData`*** (type: **base64**): morphological data of the **Donatus morphology document type** (described below), base 64 encoded.
2. ***`unparsedURL`*** (type: **string**): a URL from which unanalyzed forms may be obtained. These data will be of the **Arboreal termlist document type.** They should be retrieved immediately, because they are considered transient data; after a short time, the URL may no longer be valid, as the data will be

flushed from the server to conserve disk space. (The expiration time for these data is set by the server administrator.)

3. ***code*** (type: **i4**/**int**): result code for the operation, in the range `0..3`, with the following semantics:

   0 — success
   1 — invalid parameters (the method was not called correctly)
   2 — server misconfiguration (the Donatus server is not correctly configured)
   3 — transient server error

   A client may wish to try the call again in the case of result `3`, which may occur if Donatus times out in accessing certain resources. Result type `2` should be considered irrecoverable, and type `1` indicates a fault on the part of the client.

## donatus.addEntries(*lang, user-id,* $\{$*lemma, infl-form, morph-label*$\}$ [...])

Here ***lang*** is a language identifier, as assigned in ISO 639-1/639-2 "Codes for the Representation of Names of Languages."[1] The ***user-id*** parameter is used for the purpose of generating metadata that track revisions to the system. No validation will be performed on this ID. The remaining data consist of one or more **morphological triples** consisting of ***lemma, infl-form, morph-label***; that is, a lemma (or "basic form" or "citation form" or "headword), an inflected form, and a morphological labeling (or analysis) of the inflected form. (More abstractly, the morphological triple may be considered a mapping between an inflected form and the pair $\{$***lemma, morph-label***$\}$, where ***morph-label*** specifies the set of morphoseman-tic/morphosemantic features that are realized on ***infl-form***.) An instance of a morphological triple for Latin is (*muscipulum, muscipulo, N neut abl/dat sg*); i. e. the Latin form *muscipulo* is the ablative or dative singular form of the neuter noun *muscipulum* 'mousetrap'. Donatus makes no assumptions about the form of morphological labels (***morph-label***), and allows the structure of these to vary arbitrarily across various languages and backends.

In order to avoid the overhead of multiple method calls (a problem that is, needless to say, exacerbated when dealing with remote network calls), Donatus allows the bundling of entries; multiple morphological triples may be submitted in a single call. Note that, since a morphological triple involves three distinct parameters, a call to **donatus.addEntries** must always be accompanied by $3n + 2$ parameters, where $n \geq 1$.

This method returns a **struct** with the members ***code*** (see the explanation of result codes above) and ***message*** (a **string** describing the action taken by Donatus, suitable for display in an end-user application, e. g. as a message dialog).

[1] See `http://lcweb.loc.gov/standards/iso639-2/`. The United States Library of Congress is the registration authority (RA) for the ISO 639-2 standard.

# 6 The Donatus WTAG Document Type

Linguistic data sent to Donatus for analysis via the XML-RPC interface must be submitted in the WTAG format. This format is designed (1) to abstract from arbitrary document structure, (2) to provide data that will be needed for various statistical information processing purposes, (3) to allow for explicit segmentation (tokenization at the word level), and (4) to allow for orthographic normalization of the underlying data.

A WTAG document has four levels:

1. The root is the element **`<wtag>`**. This element takes an obligatory attribute, **`locator`**, which specifies a canonical URI for the source document. The scheme of **locators** (or abstract URIs) employed in the Archimedes project is described in the document "The Arboreal Catalog System."
2. The **language section** level: the document contains one section for each natural language that occurs in the document. Thus, a document that contains Latin text will possess the second-level tag `<section lang="la">`.
3. The **container** level: the container is a concept that represents the primary semantic unit of interest to users of a particular document type. This unit is discussed in section 2 of the document "The Arboreal Docspecs System." For most documents and users, the container will be a *sentence* or *sentence-like* unit. The most frequent container element in the Archimedes DTD is **`<s>`**, which tags a sentence. Any allowable container for the relevant document type may appear at the container level. Optionally, the container element may take an **`id`** attribute with its value as an XML ID value, which is unique within the scope of the document. These IDs may be useful to information processing tools other than Donatus. Donatus uses them in the case of backends that support contextual morphological identification; in this situation, Donatus needs an ID in order to construct an XPointer expression that refers to the container within which a contextually-sensitive form is identified.
4. The **word** level: beneath the container are any number of words, orthographically normalized, and tagged as **`<w>`**.

Example: `<s id="Lucr.1.1">Aeneadum genetrix hominum divomque voluptas...</s>` is represented in WTAG as:

```
<s id="Lucr.1.1">
  <w>Aeneadum</w>
  <w>genetrix</w>
  <w>hominum</w>
  <w>divomque</w>
  <w>voluptas</w>
  ...[2]
</s>
```

[2] The ellipsis dots (...) here are a metasymbol; punctuation tokens are normally not included in the WTAG document.

In the case that a container has embedded text in another language, that material must appear at the word level under the appropriate section for the language in question, with the appropriate container elements and IDs (possibly repeated from another language section).

## 7 The Donatus Morphology Document Type

Morphology documents have a root element **`<morphology>`**. Their namespace is `http://archimedes.fas.harvard.edu/ns/morphology/2`. (The final portion of the URI path indicates the version of the Morphology Document grammar.)

Two types entries appear below **`<morphology>`**; these correspond to context-free and context-sensitive morphological analyses.

1. Context-free: the element is **`<lemma>`**, which takes two attributes, **`form`** (the standard citation form) and **`lang`** (the ISO 639 language specifier). E. g.:

   ```
   <lemma form="actio" lang="la">
     <definition>a putting in motion</definition>
     <variant form="actio">
       <analysis desc="N fem nom/voc sg" xlink:type="simple"/>
     <variant form="actionem">
       <analysis desc="N fem acc sg" xlink:type="simple"/>
     <variant form="actiones">
       <analysis desc="N fem acc pl" xlink:type="simple"/>
       <analysis desc="N fem nom/voc pl" xlink:type="simple"/>
     <variant form="actionibus">
       <analysis desc="N fem abl pl" xlink:type="simple"/>
       <analysis desc="N fem dat pl" xlink:type="simple"/>
     <variant form="actionis">
       <analysis desc="N fem gen sg" xlink:type="simple"/>
   </lemma>
   ```

2. Context-sensitive: the element is **`<context-form>`**. Thus

   ```
   <context-form lang="de" xlink:href="dex.xml#s2">
     <tokens>
       <token count="1" form="baut"/>
       <token count="2" form="auf"/>
     </tokens>
     <analysis xlink:href="dex.morph.xml#de000029"/>
   </context-form>
   ```

   This format allows not just for context-sensitive morphology, but also for multiwords and lexical constituents that are realized discontinuously.

The entry contains two XLinks: (1) to the container of the source text in which the morphological form in context is displayed; (2) a link to the morphological analysis (either in this file, or in another). For the link to the morphological analysis, Donatus will assign a document-unique identifier to the **`<analysis>`** element. Linking to the **`<analysis>`** uniquely specifies both the **`<variant>`** and **`<lemma>`**, which will be the parent and grandparent respectively of the **`<analysis>`** element.

Words (and their parts) are referred to by a **token-of-a-type** counting scheme. Thus the XML fragment above refers to the boldfaced words in the text under the container with the ID `s2`: *Die Technik setzt die Natur vielfach voraus und **baut** auf ihr **auf.*** The analysis ID `de000029` links to the boldfaced element in the following XML fragment:

```
<lemma form="aufbauen" lang="de">
  <definition>to build</definition>
  <variant form="baut...auf">
    <analysis desc="3SIE,2PIE" id="de000029"/>
  </variant>
</lemma
```

Morphology files returned by Donatus will include an internal DTD that is the canonical definition of this document type.

# 8 Comments and Suggestions

Both Donatus and this documentation are evolving projects. Comments and suggestions on the Donatus API are appreciated and may be directed to:

`mhyman@fas.harvard.edu`