

LexEVS 6.x API Programmer's Guide

Contents of this Page

- [Introduction](#)
- [LexEVS 6.x API Programmer's Guide Sections and Related Documents](#)
 - [LexEVS 6.x API](#)
 - [LexEVS 6.0 CTS2 API](#)
 - [LexEVS 6.x CTS2 API Quick Start](#)
 - [LexEVS 6.2 CTS2](#)
 - [LexEVS 6.x Value Set and Pick List Definition Guide](#)
- [Related Documents](#)

LexEVS 6.x Programmers Links

- [Programmer's Guide Main Page](#)
 - [LexEVS API](#)
 - [LexEVS 6.0 CTS2 API](#)
 - [LexEVS 6.x CTS2 API Quick Start](#)
- [Value Set and Pick List Guide](#)
- [LexEVS 6.0 Main Page](#)
- [LexEVS Current Release](#)

Introduction

This document is intended for LexEVS developers, and explains how to use the core services and the APIs.

LexEVS 6.x API Programmer's Guide Sections and Related Documents

The following is a listing and description of the sections of the LexEVS 6.x API Programmer's Guide. Links to these documents are also found in the panel titled "LexEVS 6.x Programmers Links."

LexEVS 6.x API

This section describes the LexEVS interfaces for developers.

LexEVS 6.0 CTS2 API

This section describes programmatic access to LexEVS 6.0 implementation of pre-CTS 2 features and services. Developers can build custom applications, tools, and services that can make calls as per the CTS 2 specification against the LexEVS Pre-CTS 2 API.

LexEVS 6.x CTS2 API Quick Start

This section describes programmatic access to LexEVS 6.1 implementation of the OMG CTS2 final API specification. This is a REST-based architecture and is a significant departure from Java APIs featured in the CTS2 API in LexEVS 6.0.

LexEVS 6.2 CTS2

This section describes LexEVS 6.2 modifications made to the CTS2 services.

LexEVS 6.x Value Set and Pick List Definition Guide

The Value Set Guide describes the LexEVS 6.0 Value Sets and Pick List Definitions and use of the Developer's Value Set GUI.

Related Documents

- [LexEVS 6.x Installation Guide](#) for information about software requirements and configuring your environment
- [LexEVS 6.x Loader Guide](#) for information about loaders provided, mapping, and how to create your own loaders using the loader framework.