Enterprise Architect

Guide to Software Engineering

Guide to Software Engineering with Sparx Systems Enterprise Architect

Author: Sparx Systems

Date: 13/11/2023
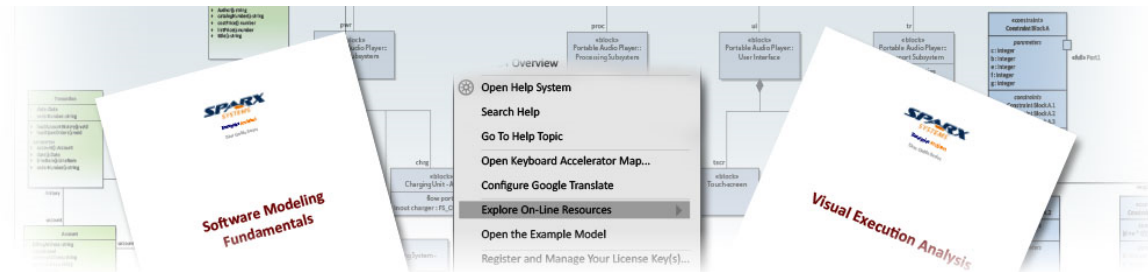
Version: 16.1

CREATED WITH ENTERPRISE ARCHITECT

Table of Contents

Guide to Software Engineering

This chapter provides background information, worked examples, advice and best practice information to help you leverage the complete set of tools and services built into the Enterprise Architect product suite for developing and working with software, source code, executables, databases, XSD and much more. These guides are provided in downloadable PDF format.

The Software Engineering Series Guidebooks

The guidebooks listed here catalogue Enterprise Architect's numerous features that are available to developers. The guidebooks are provided in four separate volumes, each covering a specific part of the tool's formidable feature set for software engineers. They are available in PDF format from the 'Resources' pages of the Sparx Systems website.

Software Modeling Fundamentals - Describes the Integrated Development Environment for working with models of static structure, and forward and reverse engineering programming code in a wide range of languages, including user defined ones.

[**Software Modeling Advanced**](#) - Describes database modeling using the Database Builder connecting to live databases, the Schema Composer for working with XML Schemas, Geospatial Geo-databases and integration with external Data Sources, and tools such as Jira and ServiceNow.

[**Simulation and Behavior**](#) - Describes Model Simulations and visualizations that bring business, systems and behavioral models to life. Automatically generate programming code for StateMachines and DMN based decision models - immediately available for inclusion into your projects and your code base.

[**Visual Execution Analysis**](#) - Describes the features for recording, profiling, testing and visualizing a code base and automatically generating diagrams that describe running code, allowing code hot spots to be analyzed, behavior to be documented and faults to be identified.