

# MPS User's Guide

## MPS User's Guide

Welcome to MPS. This User Guide will navigate you through the many concepts and usage patterns that MPS offers and will give you a hand whenever you need to know more details about any particular aspect of the system.

First, the Introduction section will offer a high-level overview of the basic notions and their roles. In the second section, named Using MPS, you'll get familiar with the interface through which you'll communicate with MPS. Although very small, there still are some differences between how you interact with MPS and how you typically use other common programming tools.

In the third section, called Defining Languages, we'll get to the meat of MPS. We'll show details on how to define the many aspects of your custom languages - their structure, editors, generators and type systems rules. The IDE integration section will then provide some additional context necessary to help you improve the IDE aspect of your languages and integrate them nicely into MPS.

The Platform languages section gives you details on all languages bundled with MPS including the corner stone language of MPS - the BaseLanguage. Whatever didn't fit the mentioned scheme was placed into the last Miscellaneous section.

## Tutorials and cookbooks

Don't forget to check out our tutorials and focused cookbooks listed in the Tutorials and Cookbooks sections, to learn more about individual aspects of MPS

1. Introduction to MPS
  - [Basic notions of MPS: nodes, concepts, and languages](#)
  - [MPS project structure](#)
2. Using MPS
  - [Commanding the editor](#)
  - [Version control notes](#)
  - [Using Debugger](#)
3. Defining Language in MPS
  - [Structure](#)
    - [Constraints](#)
    - [Behavior](#)
    - [SModel language](#)
    - [Pattern](#)
  - [Editor](#)
    - [Editor Actions](#)
  - [Generator](#)
  - [Typesystem](#)
    - [Using the typesystem](#)
  - [Data flow](#)
  - [TextGen](#)
4. Languages for IDE Integration
  - [Plugin](#)
  - [Intentions](#)
  - [Find Usages](#)
  - [Stubs](#)
  - [Debugger](#)
  - [Run Configurations](#)
  - [Changes highlighting](#)

- 5. Platform Languages
  - Base Language
    - Base Language Extensions Style Guide
  - Closures
  - Collections language
  - Tuples
  - Dates language
  - Regexp language
  - GText language
  - Unit test language
  - Type Extension Methods
  - Builders
  - XML language
  - Other languages
- 6. Miscellaneous
  - Language tests language
  - Build languages
    - Build Language
    - Packaging Language
    - Custom MPS language
  - Generating MPS models from Ant
  - GWT Support
  - FAQ
- 7. Tutorials
  - JetBrains MPS tutorial
  - Markus Voelter's tutorial
  - The beginners screen-cast series on JetBrains TV
  - The MPS channel on JetBrains TV
- 8. Cookbooks
  - Type System
  - Dataflow
  - Regular expressions
- 9. Copyrights



Did not find the answers?  
Post your questions to our discussion forum!