



# Generating an IDE using Rascal

- **Goal:** create an IDE for your DSL
- **Example:** state machine DSL

- **Requirements:**

- Syntax highlighting
- Content assist
- Refactoring
- Builders
- Error marking
- Hyperlinking
- Outlining
- Folding

The screenshot shows the Rascal IDE interface with the following components and callouts:

- Highlighting and folding:** A callout bubble pointing to the code editor.
- Graphical visualization:** A callout bubble pointing to the state transition diagram.
- Hierarchical outline:** A callout bubble pointing to the Outline view.
- Error marking:** A callout bubble pointing to an error message: "Unreachable state Press 'F2' for focus".
- Context menu:** A callout bubble pointing to a context menu with the option "Visualize Rename...".

The code editor shows the following DSL code:

```

1 state Opened
2   close => Closed
3
4 state Foo
5
6 state Closed
7   open => Opened
8   lock => Locked
9
10 state Locked
11   unlock => Closed
  
```

The state transition diagram shows states Opened, Closed, Locked, and Foo. Transitions are: Opened to Closed (close), Closed to Opened (open), Closed to Locked (lock), Locked to Closed (unlock), and Closed to Opened (open).

The Outline view shows the following structure:

- States
  - Closed
  - Foo
  - Locked
  - Opened
- Transitions
  - close => Closed
  - lock => Locked
  - open => Opened
  - unlock => Closed

- **Implement IDE services:**

- IDE services = Rascal functions
- Register call-backs to Eclipse

