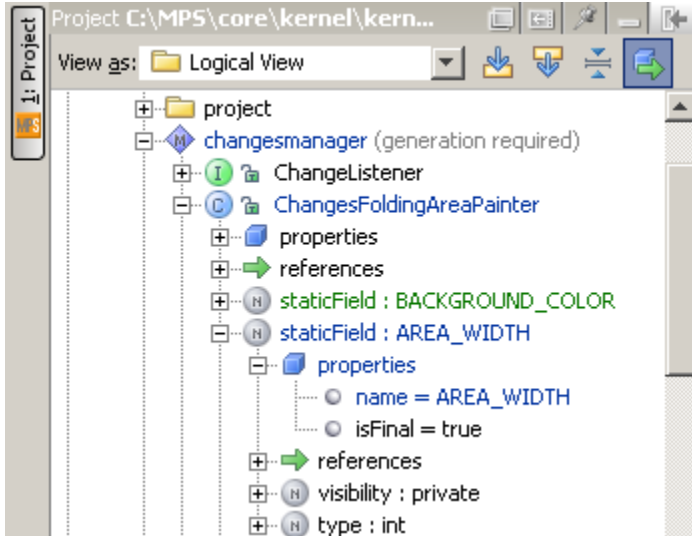


Changes highlighting

Changes highlighting is a handy way to show changes which were made since last update from version control system. The changes in models are highlighted in the following places:

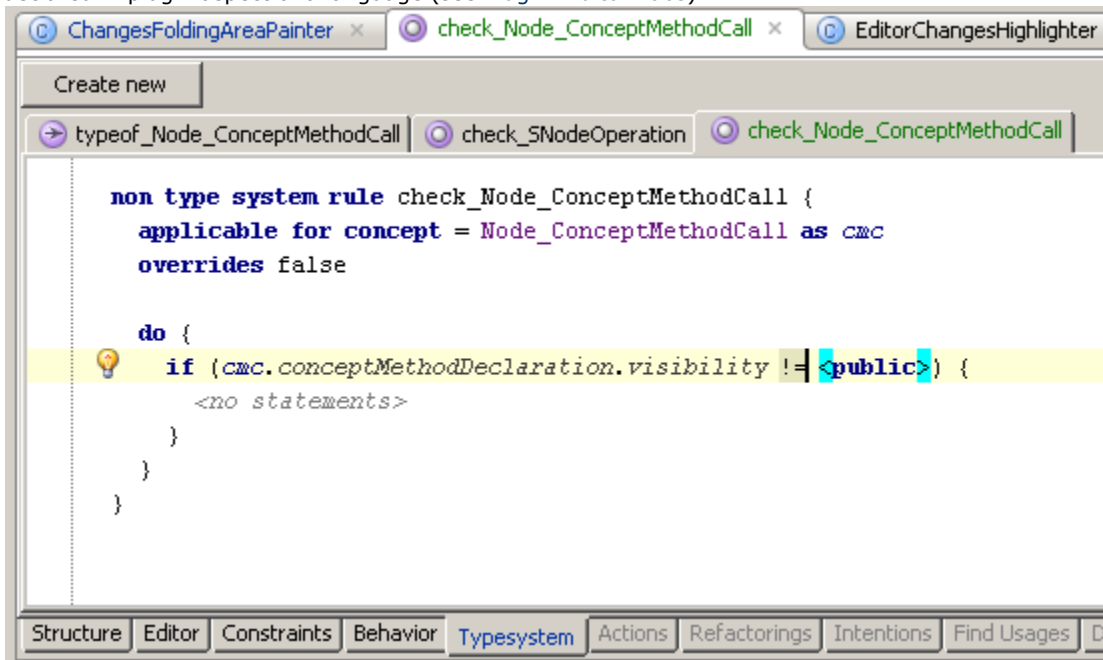
Project tree view

Models, nodes, properties and references are highlighted. green means new items, blue means modified items, brown means unversioned items.



Editor tabs

Highlighting appears for all of the editor tabs: for language aspect tabs of a concept and also for custom tabbed editors declared in plugin aspect of a language (see Plugin: Editor Tabs).



Editor

Every kind of changes are highlighted in MPS editor: changing properties and references, adding, deleting and replacing nodes.

```

private void reinitMessageGroups() {
    myMessageGroups = new arraylist<MessageGroup>;
    EditorComponent editorComponent = getEditorComponent();

    list<EditorComponentChangesHighlighter.ChangeEditorMessage> messagesWithCells
        = myEditorComponentChangesHighlighter.getEditorMessages().where({~m =
            toList;
    messagesWithCells = messagesWithCells.sort({~aMsg,~bMsg =>
        Rectangle a = getMessageBounds(editorComponent, aMsg);
        Rectangle b = getMessageBounds(editorComponent, bMsg);
        // First compare by y, then by x, and, after all, by height
        if (a.y == b.y) {
            if (a.x == b.x) {
                return a.height - b.height;
            } else {
                return a.x - b.x;
            }
        } else {
            return a.y - b.y;
        }
    }, asc).toList;
}

```

If you hover mouse cursor over the highlighter's strip on the left margin of editor, the corresponding changes become highlighted in editor pane.

If you want to have your changes highlighted in editor pane all the time (not only on hovering mouse cursor over highlighter's strip), you can select "Highlight Nodes With Changes Relative to Base Version" option in IDE Settings Editor.

```

private void reinitMessageGroups() {
    myMessageGroups = new arraylist<MessageGroup>;
    EditorComponent editorComponent = getEditorComponent();

    list<EditorComponentChangesHighlighter.ChangeEditorMessage> messagesWithCells
        = myEditorComponentChangesHighlighter.getEditorMessages().where({~m =
            toList;
    messagesWithCells.sort({~aMsg,~bMsg =>
        Rectangle a = getMessageBounds(editorComponent, aMsg);
        Rectangle b = getMessageBounds(editorComponent, bMsg);
        // First compare by y, then by x, and, after all, by height

```

Added node in 'statement' role

If you click on highlighter's strip on the left margin, there appears a panel with three buttons: "Go to Previous Change", "Go to Next Change" and "Rollback".

```

private void reinitMessageGroups() {
    myMessageGroups = new arraylist<MessageGroup>;
    EditorComponent editorComponent = getEditorComponent();

    list<EditorComponentChangesHighlighter.ChangeEditorMessage> messagesWithCells
        = myEditorComponentChangesHighlighter.getEditorMessages().where({~m =
            toList;
    messagesWithCells = messagesWithCells.sort({~aMsg,~bMsg =>
        Rectangle a = getMessageBounds(editorComponent, aMsg);
        Rectangle b = getMessageBounds(editorComponent, bMsg);
        // First compare by y, then by x, and, after all, by height

```

Rollback

If you click "Rollback", all the corresponding changes are reverted.

This feature allows you to freely make any changes to the MPS model in the editor without fear, because at any moment you can revert your changes conveniently right from the editor.

Previous Next