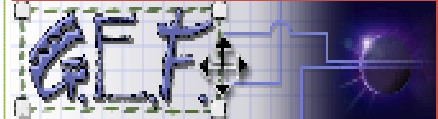


# GEF COURSE CONTENT



The Eclipse Graphical Modeling Framework (GMF) provides a generative component and runtime infrastructure for developing graphical editors based on **EMF** and **GEF**. The project aims to provide these components, in addition to exemplary tools for select domain models which illustrate its capabilities.

Annamalai C, ANCIT Consulting

# Graphical Editing Framework Training Program

## Course Overview

The Graphical Editing Framework (GEF) provides a foundation for building rich, interactive user interfaces, which are not easily built using native widgets found in the base Eclipse platform. GEF is based on MVC architecture. Every GEF-based application uses a model to represent the state of the diagrams being created and edited.

## Audience

This course is meant for Developers (Team Members, Project Leads and Managers) with advanced programming knowledge in Java.

## Prerequisites

- Strong knowledge of Java and XML.
- Should have hands on developing Eclipse plugins.
- Knowledge of EMF would be helpful.

## Duration and Additional Information

- It is a 2 Day Program and extends upto 8hrs each day. Therefore a 16hrs Courseware.
- The format is 60% Lab and 40% Theory.
- Max 15Nos per Class.

For details of the Course please refer to the attached Annexure.

## Annexure – Table of Contents for GEF Training

### Graphical Editing Framework

- 1 Creating a simple Empty Graphical Editor
2. Palette on the Editor
  - 2.1 Simple Palette
  - 2.2 Stack on Palette
  - 2.3 Drawer on Palette
3. Node
  - 3.1 Placing a canvas on the editor
  - 3.2 Drawing Node on the Editor – Figure
  - 3.3 Moving and Resizing of the Node - LayoutEditPolicy
  - 3.4. Creation of Node from Palette - LayoutEditPolicy
  - 3.5. Deletion of Node - ComponentEditPolicy
  - 3.6. Direct Editing on the Node – DirectEditManager
  - 3.7. Selection Feedback on Node - SelectFeedbackPolicy
  - 3.8. Placing Node within Node - ContainerEditPolicy
  - 3.9. Enabling ScrollBar on a Node - ScrollPane
4. Connection
  - 4.1 Drawing Simple Connection - PolylineConnection
  - 4.2 Selecting the Connection Line- ConnectionEndPointsRole
  - 4.3 Creating Connection Line from Palette - GraphicalNodeRole
  - 4.4 Reconnection to Other Node - GraphicalNodeRole
  - 4.5 Connection Deletion - ComponentEditPolicy
  - 4.6 Decoration on Connection -Polygon Decoration
  - 4.7 Routing Algorithm – Router (Ex: ManhattanConnectionRouter)
  - 4.8 Anchor Point -ChopboxAnchor
  - 4.9 Bend Point - BendPoinEditPolicy
  - 4.10 Line Style
5. Add Ons
  - 5.1. Property View
  - 5.2. Zoom
  - 5.3. Grid
  - 5.4. Background image to Editor
  - 5.5. Print your Editor
  - 5.6. Export your Editor as image
  - 5.7. Outline
    - 5.7.1 Outline view
    - 5.7.2 Tree View
  - 5.8. Context Menu
  - 5.9. Preference
  - 5.10. Cut, Copy and Past