

Glossar

<u>abstrakte Syntax</u>	Die abstrakte Syntax ist als darstellungsunabhängige algebraische Struktur, welche im Programmwort auftretende Konstrukte und Schachtelbeziehungen identifiziert, anzusehen. Die abstrakte Syntax kann bzw. wird für weitere Phasen der Compilierung, wie semantische Analyse und Codegenerierung benutzt.
<u>Arbeitspalette</u>	Leiste im Editor, die die einzelnen Tools enthält: Zoom, Select, Note sowie Creation Tools für Objekte und Beziehungen
<u>Beziehung</u>	Verbindung von zwei Objekten, besteht aus: -Node mit definierter grafischer Repräsentation (Pfeil mit Decorations) und -Name mit Label im gmfgraph -zugehörige eClass im eCore -Creation Tool im gmftool -Mappingeintrag in gmfmap
<u>Bundles</u>	sind Serviceanwendungen, die dynamisch in das OSGi-Framework eingespielt und auch wieder entfernt werden können (seit eclipse Version 2.2 sind alle eclipse-Plugins auch als OSGi-Bundles verfügbar)
<u>Dateiformat, spezielles</u>	Menge von Regeln, mit denen Informationen in einer Datei codiert werden. Bei uns: .gmfmap_diagram (s. grafische Daten) bei GMF: .ecore, .gmftool, .gmfgraph, .gmfmap
<u>Daten, grafische Editor</u>	Positionsdaten und Anordnung der Objekte in einer _diagram-Datei Unser Editor. Dessen Benutzeroberfläche besteht aus Zeichenfläche und Arbeitspalette
<u>Element</u>	jedes Artefakt, das auf der Zeichenfläche vorkommt: Beziehungen, Objekte, Notizen
<u>Endprodukt exportieren Framework</u>	der mit unserem Editor mittels GMF erzeugte GMF-Editor Speichern der Projektdateien in den workspace Als Framework bezeichnet man eine Menge von verknüpften Klassen, welche zusammen ein wiederverwendbares und erweiterbares Gerüst für die Entwicklung von Software eines bestimmten Typs bilden. Beispiel: eclipse oder GMF
<u>GEF</u>	Graphical Editing Framework (GEF) dient der Erstellung von grafischen Ansichten für Modelle
<u>GMF</u>	Graphical Modeling Framework (GMF) ist ein Werkzeug zur generativen Erstellung von metamodellbasierenden Editoren.
<u>GMF-Editor</u>	Tool zur Darstellung eines vorher mit GMF definierten Modells.
<u>GMF-Graph</u>	Definition der Editoroberfläche in GMFTool
<u>GMF-Map</u>	Hier wird die Verbindung der bisher erstellten eigenständigen Teildefinitionen spezifiziert: Metamodell, grafische und Tooling Definition.
<u>GMF-Tool</u>	Dient zum Erzeugen der Werkzeuge für die Palette
<u>gmfmap_diagram</u>	s. spezielles Dateiformat

<u>importieren</u>	Öffnen der Projektdateien
<u>Metamodell</u>	Modelle, die beschreiben, wie Modelle gebaut werden. Ein Metamodell enthält Methoden, die Syntax und Semantik der Modelle definieren. Diese Methoden selbst sind wieder reale Methoden, deswegen liegt es nahe, diese Methoden auch durch Modelle zu beschreiben.
<u>Metamodellierungssprache</u>	Eine Metamodellierungssprache die in ihrer Abstraktionsebene höher steht als Modellierungssprachen und wird deshalb zu deren Entwicklung genutzt.
<u>Modell</u>	vereinfachte und idealisierte Darstellung eines komplexeren Sachverhaltes
<u>Toolgroup</u>	Die einzelnen grafischen Objekte und Beziehungen zu einer Gruppe zusammengefasst , welche im Endprodukt angezeigt wird.
<u>Objekt</u>	Knotenpunkt auf der Zeichenfläche, besteht aus: -Node mit definierter grafischer Repräsentation (z.B. Rechteck, Ellipse, Kreis) und -Name mit Label im gmfgraph -zugehörige eClass im eCore -Creation Tool im gmftool -Mappingeintrag in gmfmap
<u>Öffnen</u>	Laden und verbildlichen der Projektdateien auf der Zeichenfläche
<u>OSGi</u>	Industriekonsortium, von dem die Entwicklung von Standards für herstellerunabhängige Java-Frameworks ausgeht
<u>OSGi-Framework</u>	modulierbare und offene Serviceplattformen für die Vernetzung von einzelnen Bundles - Beispiel: Equinox ist ein in eclipse eingebettetes OSGi-Framework
<u>OSGi-Manifest</u>	unter META-INF/MANIFEST.MF stehen die Daten, die das OSGi-Framework benutzt, um das Plugin als Bundle zu erkennen und auszuführen
<u>Plugin</u>	Ein Plugin wird als Komponente begriffen, die einen bestimmten Dienst in einem Oberprodukt zur Verfügung stellt
<u>Projekt</u>	Besteht aus den Dateien *.ecore, *.gmfgraph, *.gmfmap, *.gmftool., *.gmfmap_diagram Wobei „*“ für den jeweiligen Projektnamen steht.
<u>Projektdateien</u>	Die Dateien aus denen das Projekt besteht.
<u>Workspace</u>	Ordner in eclipse, in dem aktuelle Daten aller Projekte gespeichert werden
<u>Zeichenfläche</u>	Fläche im Editor, auf der die Elemente eingefügt werden