

Module descriptions:

<p>architecture model: <u>Representation of architecture models</u> as defined in reflexion modelling, i.e. modules / components connected via connectors indicating intended or unintended dependencies. Includes definition of a file format for <u>storing architecture models</u> and definition of architecture model change events.</p>	<p>reflexion modelling, modules, components connected, connectors, intended, unintended dependencies, file format, storing, change events</p>
<p>resource-mapping: This module encapsulates the data structures and functionality required for mapping elements of a system's implementation to elements of an architectural model of the system. For this purpose, it also defines an abstract resource model. It also defines events that are raised when the mapping is changed to propagate these changes to the architecture model if needed.</p>	<p>data structures, functionality required, mapping elements, system's implementation, elements, architectural model, system purpose, abstract resource model, events, mapping changed, propagate changes</p>
<p>impl-model: This module defines the data structure needed to store dependency information extracted from a codebase. In addition, a format to persist this information is defined in this module as well as events which can be used to propagate changes in the implementation to interested components, such as the architectural model.</p>	<p>data structure, store dependency information, extracted, codebase, format, persist, events, propagate changes, implementation, interested components</p>
<p>reporting: Container module to gather different report generators that convert the information (or a part of it) stored in architectural models, mapping, and implementation models into human-readable form.</p>	<p>Gather, different report generators, convert the information, mapping, implementation models, human-readable form</p>
<p>jdt-builder: This module contains a parser for Java based on Eclipse JDT to populate Jittac's internal implementation model.</p>	<p>jdt-builder: parser, Java, Eclipse JDT, internal implementation model</p>
<p>eclipse-resource-mapping: This module wraps the Eclipse-internal resource model for Jittac such that Eclipse resource can be mapped to components in architecture models</p>	<p>eclipse-resource-mapping: wraps, Eclipse, internal resource model, Eclipse resource, components</p>

eclipse-ui: Contains all user interface elements for the Eclipse version of Jittac, including, among others the architecture model editor and detailed reference view.	eclipse-ui: user interface, elements, eclipse, architecture model editor, reference view
eclipse-codesupport: Contains Jittac-specific extensions to the Java editor of Eclipse such as markers in the editor for architectural problems.	eclipse-codesupport: extensions, Java editor, Eclipse, markers, architectural problems
eclipse-main: Contains classes required by the Eclipse plugin architecture and plugin-wide utility functions.	eclipse-main: Eclipse, plugin architecture, plugin-wide, utility, functions

Mappings:

(1) architecture-model

se.kau.cs.jittac.model.am

(2) resource mapping

se.kau.cs.jittac.model.mapping

(3) impl model

se.kau.cs.jittac.model.im

(4) reporting

se.kau.cs.jittac.model.report

(5) jdt-builder

se.kau.cs.jittac.eclipse.builders.jdt

(6) eclipse-resource-mapping

se.kau.cs.jittac.model.mapping.eclipse

(7) eclipse-ui

se.kau.cs.jittac.eclipse.ui

(8) eclipse-codesupport

se.kau.cs.jittac.eclipse.codesupport

(9) eclipse-main

se.kau.cs.jittac.eclipse

se.kau.cs.jittac.eclipse.util

exclude all feature classes and model package