

  
**objectiF<sup>®</sup>**

## The Features of **objectiF<sup>®</sup>** Enterprise Edition

If you want to consistently design with UML in your company, but implement with a range of technologies and languages, then **objectiF<sup>®</sup>** Enterprise Edition is just what you're looking for!

**objectiF<sup>®</sup>** Enterprise Edition is more than a classic UML tool. It allows you to standardize your software development, and automate a large part of it. You can design the domain-oriented architecture of your applications and automatically create the technical architecture, as well as a lot of the code, using model transformations in **objectiF<sup>®</sup>**. This makes UML models more productive.

Whether you're developing Java, EJB, Web or SOA applications, client-server applications in .NET or technical applications in C++ - **objectiF<sup>®</sup>** Enterprise Edition provides the support you need: from business process modeling with BPMN to UML, right through to the code.

Take a look at the features in more detail.

Here is a selection of the functions and advantages of *objectiF*® Enterprise Edition. One look and you'll know: *objectiF*® is the right choice for you.

Function	Advantages	Support with <i>objectiF</i> ® Enterprise Edition
<b>Data Organization</b>		
Transaction secure repository with Backup/Restore	Secure data management	Yes <ul style="list-style-type: none"> <li>Full, <b>multi-user capable LAN repository</b> for efficient project work in a team</li> <li>Optimized for a high number of users and types</li> </ul>
Automatic assurance of consistency between model and code after every change	High quality results	Yes <ul style="list-style-type: none"> <li>Every change in the model or code is applied and made visible to all users</li> <li>Every change, such as re-naming types and methods, is automatically implemented in all relevant areas in the model and code</li> </ul>
Upwards compatability between repositories	Highly flexible during the start and extension of an <i>objectiF</i> ® application	Yes <ul style="list-style-type: none"> <li>Processing of all data created with alternative <i>objectiF</i>® Editions</li> <li>Processing of all data created with alternative <i>objectiF</i>® Trial Editions</li> </ul>
Import/Export	Great flexibility for project organization, as well as for acquirers and suppliers	Yes <ul style="list-style-type: none"> <li>Simple exchange and consistent integration of results in separate projects.</li> </ul>
<b>Version Management</b>		
Package-oriented versioning with standard configuration management systems	Work securely with version control	Yes
<b>Modeling Standards</b>		
UML (Unified Modeling Language)	Practical implementations of UML for increased result consistency, simple application and efficient development.	Yes
BPMN (Business Process Modeling Notation)	Extensive support of SOA development due to consideration of B2B integration and the application of Web services as well as WS-BPEL generation	Yes <ul style="list-style-type: none"> <li>Business process modeling with BPMN</li> <li>Orchestration with Web services at the model level</li> <li>WSDL and XSD Reverse Engineering for external Web service allocation on the model level</li> </ul>

Function	Advantages	Support with the <i>objectiF</i> ® Enterprise Edition
<b>Model-Driven Development with Model Transformations and Code Generation According to the Concepts of MDA/MDD</b>		
Instantly applicable and iterative model transformations to automate the development of <b>SOA applications</b> in Java	<p>Fast development with high degree of automation</p> <p>Simple domain-oriented modeling</p> <p>Comply with architecture standards</p> <p>Investment protection for specialist knowledge</p>	<ul style="list-style-type: none"> <li>▪ Automatic transformation of the business process model into WS-BPEL with WSDL and XSD. Automatic transformation of domain-oriented service models into Java Session Beans and WSDL</li> <li>▪ Automatic transformation of domain-oriented business services into business messages in Java classes and XSD</li> <li>▪ Automatic transformation of domain-oriented entity models into Java Entity Beans</li> <li>▪ Reverse Engineering of XSD files into business messages</li> <li>▪ Reverse Engineering of WSDL files into business services</li> <li>▪ The resulting Java classes and XML documents (WSDL, BPEL, PDD, and XSD files) can be modified and extended in the technical model. Manual entries remain intact during further transformations.</li> </ul>
Instantly applicable and iterative model transformations to automate the development of <b>Web applications</b> in Java	<p>Fast development with high degree of automation</p> <p>Simple domain-oriented modeling</p> <p>Comply with architecture standards</p> <p>Investment protection for specialist knowledge</p>	<p>Yes</p> <ul style="list-style-type: none"> <li>▪ Automatic transformation of domain-oriented service and entity models into Session Beans and Entity Beans</li> <li>▪ Automatic transformation of user-interface presentation models into Java Server Faces with backing beans and configuration files</li> </ul>
Ready-to-use, iterative model transformations for automating the development of <b>Web services and EJB</b>	<p>Fast development with high degree of automation</p> <p>Simple domain-oriented modeling</p> <p>Comply with architecture standards</p> <p>Investment protection for specialist knowledge</p>	<p>Yes</p> <ul style="list-style-type: none"> <li>▪ Automatic transformation of domain-oriented service models into Java Session Beans and WSDL</li> <li>▪ Automatic transformation of domain-oriented business messages into Java classes and XSD</li> <li>▪ Reverse Engineering of XSD files into business messages</li> <li>▪ Reverse Engineering of WSDL files into business services</li> <li>▪ The resulting Java classes and XML documents (WSDL, BPEL, PDD and XSD files) can be modified and extended in the technical model. Manual entries remain intact during further transformations</li> </ul>

## The Features of *objectiF*® Enterprise Edition

Function	Advantages	Support with the <i>objectiF</i> ® Enterprise Edition
Instantly applicable and iterative model transformations to automate the development of <b>Persistence Layer in .NET</b>	<p>Fast development with high degree of automation</p> <p>Simple, domain-oriented modeling</p> <p>Comply with architecture standards</p> <p>Investment protection for specialist knowledge</p>	<p>Yes</p> <ul style="list-style-type: none"> <li>▪ Automatic transformation of the domain-oriented into technical models</li> <li>▪ Automatic creation of the persistence layer with NHibernate</li> </ul>
<b>Higher Modeling Functions</b>		
Extending the UML language scope with profiles from stereotypes and user-defined properties	Adapt and extend and tailor <i>objectiF</i> ®, the way you want it	Yes
Integration of the test framework JUnit and NUnit at the model level.	High quality	Yes
Refactoring functions on architecture and class levels	Stable solutions	Yes

Function	Advantages	Support with the <i>objectiF</i> <sup>®</sup> Enterprise Edition
<b>Code Generation</b>		
Context sensitive code editor	Comfortable navigation between code and models High quality with minimum effort: changes to the model are instantly applied to the code	Yes
Script controlled, user specific changeable and extendable code generation in line with Eclipse, and with extensive language support for Java	High productivity thanks to code generation which you can tailor to your needs	Yes
Script controlled, user specific changeable and extendable code generation in line with Visual Studio .NET, and with extensive language support for C# and Visual Basic.	High productivity thanks to code generation which you can tailor to your needs	Yes
Script controlled, user specific changeable and extendable code generation of C++	High productivity thanks to code generation which you can tailor to your needs	Yes
Generation of WS-BPEL, XSD and WSDL from business process models in BPMN	Fast development of service-oriented applications	Yes
<b>Round Trip with Eclipse and Visual Studio</b>		
Round Trip Engineering with Eclipse, via automatic or user-controlled synchronization, according to your preference	It's like working with a single tool, thanks to seamless integration Easy to maintain systems where code = model and model = code	Yes
Easy navigation between models and code in <i>objectiF</i> and the code in Eclipse or Visual Studio .NET – both ways with just one click	Faster implementation through navigation between models and code	Yes
<b>Tool Support in the Development of User-Specific Model Transformations</b>		
Profile controlled generation of user-specific model transformations in a common program language	Automatic software development tailored to your needs	Yes
<b>Generating Documentation</b>		
Adaptable MS Word templates for describing diagrams and model elements	Standard documentation guaranteed	Yes
Requirements specifications and HTML documentation at the push of a button	Time saving thanks to document generation	Yes
Automatic update of all referenced results in the documentation	Extremely up-to-date documentation	Yes
Dialog controlled creation of JavaDoc and NDoc-commentaries (even at the model level), as well as completeness checks	Simple, high quality documentation	Yes

Function	Advantages	Support with the <i>objectiF</i> <sup>®</sup> Enterprise Edition
<b>Modeling Templates</b>		
Templates for setting up a new system	Quick and easy project start	Yes For SOA, Web and EJB applications and Web services in Java and for C#, VB.NET, client-server and C++ applications. When creating a new system, the following is already taken care of for you: <ul style="list-style-type: none"> <li>▪ The domain-oriented and technical structure of a system</li> <li>▪ The domain-oriented and technical architecture model types</li> <li>▪ The corresponding model transformations</li> <li>▪ The necessary <b>Eclipse workspaces</b> and Ant scripts for <b>Deployment</b></li> <li>▪ And the necessary Visual Studio Solutions and tips for assemblies</li> </ul>
Separation of domain-oriented context menus in the domain-oriented model, and technically oriented context menus in the technical model	A tool for the whole team – easy to use thanks to target-group oriented menus	Yes
Straight forward navigation between models and code in <i>objectiF</i> <sup>®</sup> and the code in Eclipse and Visual Studio .NET – bi-directionally	Faster implementation thanks to easy switching between models and code	Yes
<b>Quality Assurance</b>		
Automatic consistency between models as well as between the models and code	Consistent systems, effortlessly	Yes
Integrated model reviewer	Fast and simple result checks	Yes
<b>Interfaces</b>		
XMI Export	Straight-forward data exchange with generators	Yes
.NET and COM interface	Extend project specific tool functions as required	Yes
Connections with the project management software <i>in-Step</i> <sup>®</sup>	Secure project control	Yes
<b>Model Comparison</b>		
Structured comparison between <i>objectiF</i> <sup>®</sup> models and export files, displaying added, changed and deleted elements	Changes constantly traceable	Yes



## The Features of **objectiF**® Enterprise Edition

### **Contact**

Want to know more about **objectiF**®? Then feel free to contact us. We're always happy to hear from you.

**microTOOL GmbH**

**Voltastr. 5**

**D-13355 Berlin**

**Tel. (+49 30) 467 086-0**

**Fax (+49 30) 464 47 14**

**[info@microTOOL.de](mailto:info@microTOOL.de)**

Or, you can simply try out one of the **objectiF**® Editions. You'll find the free downloads here:

**[www.objectiF.biz/Download](http://www.objectiF.biz/Download)**