Eclipse UFaceKit Project
# *Creation Review*

# Table Of Contents

# Introduction

The UfaceKit's goal is to:

- improve adoption of Eclipse Core technologies (like Eclipse Data Binding) outside RCP and SWT (e.g. Swing, GWT, QT)

- improve usability of Eclipse Data Binding by providing a high-level Widget-API

## *Project Naming*

**Descriptive Name:** Eclipse UFaceKit Project
**Acronym:** UFK
**Nickname:** UfaceKit
**URL:** http://www.eclipse.org/ufacekit
**Newsgroup:** eclipse.ufacekit
**Root Package:** org.eclipse.ufacekit
**Mailing List:** ufk-dev@eclipse.org

## *Links*

- Creation Review Comments/Votes: news://news.eclipse.org/ufacekit

- Project Proposal: [http://www.eclipse.org/proposals/ufacekit/](http://www.eclipse.org/proposals/ufacekit/)

# Scope

UFacekits scope is to develop highlevel API and framework modules to develop applications in a widget-toolkit independent fashion which are small and easy to extend (e.g. create an implementation for other widget-toolkits, use different widget types, Nebula-Grid instead of SWT-Table). UFacekit differs in this scope with SWT as it won't provide access to lowlevel drawing operations like GC but only provides the minimal needed subset of methods needed to develop applications which makes the task of extending and porting to new technologies easier.

Besides the widget-toolkit neutral API UFaceKit will provide established eclipse technologies/modules for currently not support platforms/runtimes (EMF for GWT, Databinding for GWT, Databinding for Swing, ...) and implementation of Databinding-Observables and JFace-Viewer implementations for currently not support model-technologies and widget-toolkits.

UFaceKit will provide its bundles with as few dependencies as possible so that they can used stand alone.

## *Widget-Toolkit independent highlevel API for Databinding*

UFaceKit will provide a highlevel API to bind a model-attribute to an UI-attribute without the need to create observables for them. UFaceKit's UI-Elements are highlevel widgets backed by native framework widgets (SWT, Swing, ...) and provide a model-technology independent possibility to bind the business-model attribute.

Besides that UFaceKit provides conversion, validation and decoration support out of the box.

Currently planned Tookits:

- SWT
  - Plain SWT
- SWT Forms
- Swing
- QT
- GWT
  - Plain GWT
- GXT

### *Observables for other Widget-Toolkits*

UFaceKit will provide observable implementations for widget technologies other than SWT.

Currently planned Toolkits:

- Swing
- QT
- GWT
  - Plain GWT
  - GXT
- SWT-Forms

### *Lightweight-Model-Technology*

UFaceKit provides a light-weight model implementation named UBean which is completely self-contained providing a generic set(int,Object)/get(int)-interface and an accompanying observable implementations. UFaceKit is internally using this model-technology e.g. all highlevel UIElements provided by UFaceKit use.

### *Application Framework*

- Provide an application framework to ease development of GUI-Applications. Probably the model developed in E4 is reused.
- Provide a Declarative Syntax based upon EMF to describe the UI

### *Other UI-Technologies*

- Android
- Eclipse – Draw2d/GEF
- OpenGL

## Out of Scope

- UFaceKit will not provide access to low-level widget APIs like Graphic-Context, ...
- UFaceKit will not provide any tooling for the Eclipse-IDE e.g. to visually develop applications

## Relationship with other projects

UFaceKit will reuse opensource technologies from projects developed in and outside the Eclipse Ecosystem

- Eclipse
  License: http://www.eclipse.org/legal/epl-v10.html
  - Platform – Equinox, Databinding, JFace, SWT
  - EMF – EMF for GWT, EMF as a supported model technology, the application model
  - E4 – Application-Model, Declarative UI, CSS-Styling
- SwingX
  License: LGPL – https://swingx.dev.java.net/
- GWT
  License: various http://code.google.com/webtoolkit/terms.html
- Mig-Layout
  License: BSD – http://www.miglayout.com/
- QT
  License: GPL + EPL-Exception – http://doc.trolltech.com/main-snapshot/license-gpl-exceptions.html

## Code Contribution

The initial code contribution is coming from the UFacekit project located at http://code.google.com/p/uface/ which is already licensed under EPL.

Until now used the following policies:

- for committers:
  http://code.google.com/p/uface/wiki/CommitterAgreement
- for contributors: http://code.google.com/p/uface/wiki/Contributing

The following people contributed to the code base:

- Tom Schindl – Initial Committer
- James Strachan – Initial Committer
- Angelo Zerr – Committer
  Agreement:
  http://groups.google.com/group/uface/browse_thread/thread/9c7d9cbe7083fc13#
- Kenneth Westelinck – Committer
  Agreement:
  http://groups.google.com/group/uface/browse_thread/thread/4285dbe607fbc6cb#
- Daniel Spiewak – djspiewak@gmail.com – Contributor
  Contribution: http://code.google.com/p/uface/issues/detail?id=8&can=1

## Mentors

- Boris Bokowski – IBM – Platform UI and e4 committer, member of the

Eclipse Architecture Council
- Chris Aniszczyk – Code 9 – PDE co-lead and Technology PMC member

# Initial Participants

### Tom Schindl - BestSolution Systemhaus Gmbh - tom.schindl@bestsolution.at (component lead)

Tom is self-employed and CEO of BestSolution.at Systemhaus Gmbh a software company building applications (RCP, J2EE) for companies around the world. Besides implementing solutions their own BestSolution.at consulted companies to introduce Eclipse Technologies into their software stack by providing its knowledge about Eclipse Technologies and Software Design experience.

Tom is one of the Platform-UI and Nebula committers working on JFace-Viewers, Nebula-Grid and contributed patches to other eclipse projects (EMF, ...). He is part of the E4 project team and wrote the EMF based platform prototype used as the starting point for the implementation of the next generation of the Eclipse-Platform.

He is a regular contributor to the eclipse newsgroups and received the top contributor award in 2007 for his work on Jface-Viewers.

### James Strachan - Progress Software - james.strachan@gmail.com

James works in the FUSE group at Progress and is a long time contributor to open source projects mostly at Apache and works on projects like ActiveMQ, Camel and ServiceMix

### Angelo Zerr - Independent - angelo.zerr@gmail.com

Angelo is the developer of TK-UI, a toolkit to manage CSS Styling/Declarative UI (XUL, XHTML) which renders to Swing and SWT. He has also developed Akrogen, an Eclipse plugin for code generation where you can describe Eclipse Wizard with XUL and Javascript and link it with template (Freemarker, Velocity, XSL...) or Ant Task to generate code.

### Kenneth Westelinck - Ebit NV - kenneth.westelinck@gmail.com

Kenneth is a senior Java Software Engineer at Ebit NV specialized in building custom J2EE, Hibernate and Spring based applications for disparate types of clients. In the past, his main focus has been solving application integration problems for clients, using both Commercial and Open Source software (Mule). Kenneth is a regular contributor to the Mule ESB platform and active on the mailing lists. Currently he's working on technical projects for an engineering company involving Eclipse RCP GUIs for monitoring and administration purposes.

# Community Support

The current google group has 36 members.

# Roadmap

- Creation Review Dec, 10th 2008
- Q1 - 2009: 1.0.0 Milestone 1 (all objectives implemented (SWT & Swing))
- Q2 - 2009: 1.0.0 Milestone 2 (all objectives for an initial version implemented)