

Modeling project requirements with Objectiver



www.cediti.be

Denis.Ballant@cediti.be
Cediti SA



www.objectiver.com

Requirements Engineering – Why ?



Project completed
on time, on budget,
and with all functions
originally specified

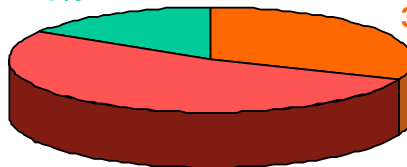
success
16%

failure
31%

Project cancelled
before completion
or never
implemented

challenged
53%

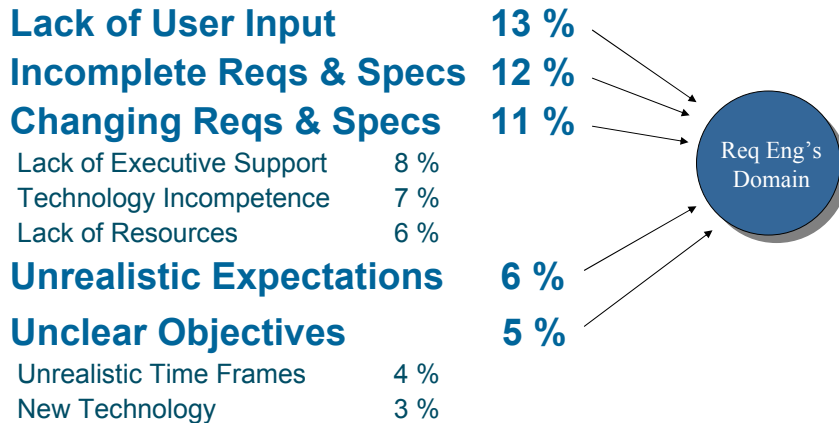
Project completed *but...* over
budget, late, and with fewer
functions than initially specified



Source: The Standish Group. Sample size was 365 companies and 8380 software projects.
http://www.standishgroup.com/sample_research/chaos_1994_1.php

Reasons for Project Failures

Objectiver

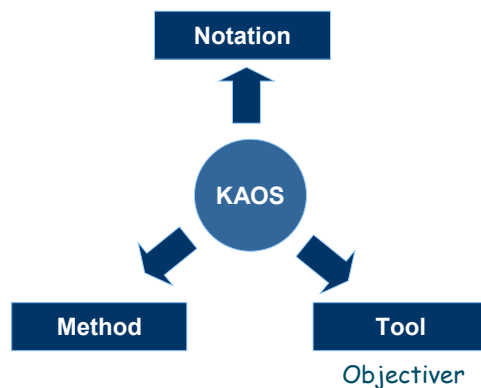


The KAOS approach

Objectiver

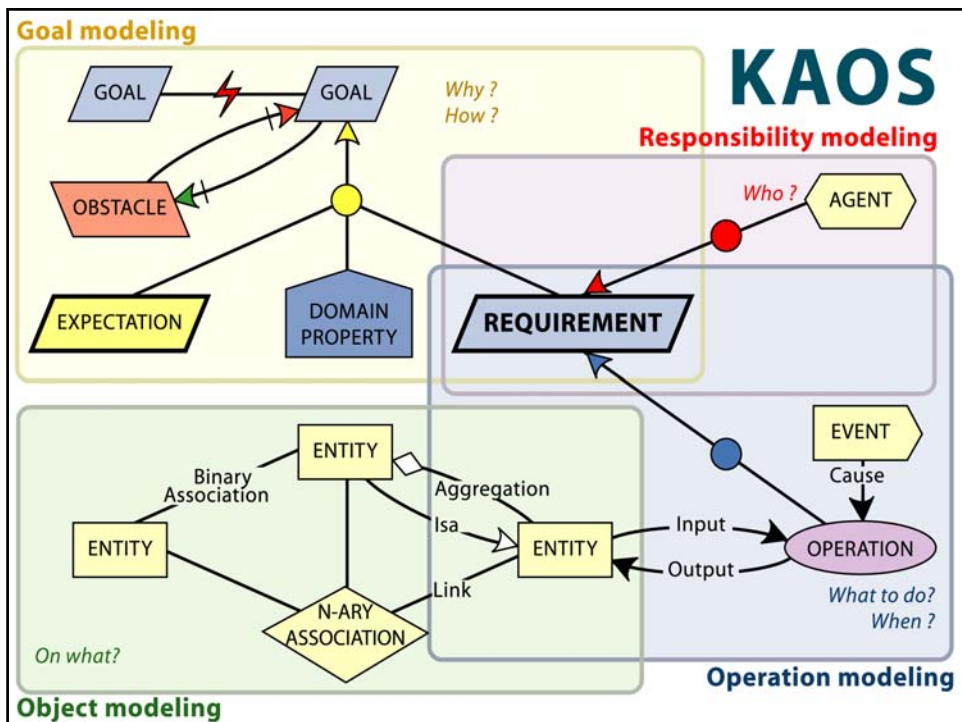
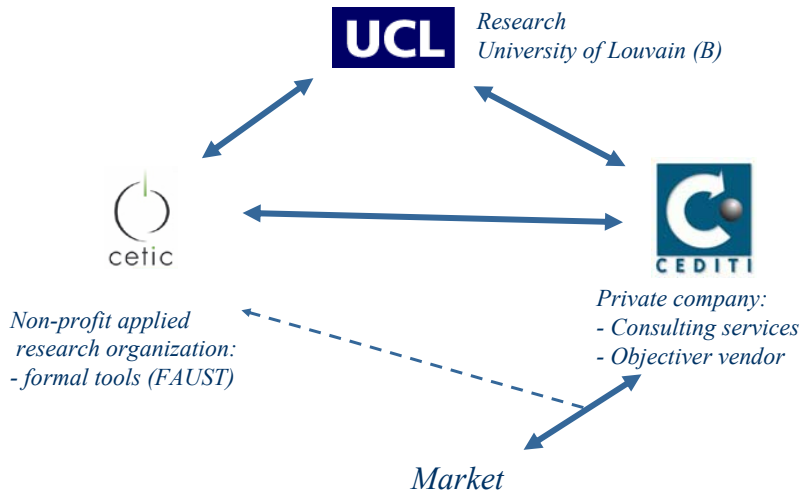
A goal-driven approach to requirements engineering

- Elicitation
- Structuring
- Specification
- Analysis
- Negotiation
- Communication
- Documentation
- Evolution



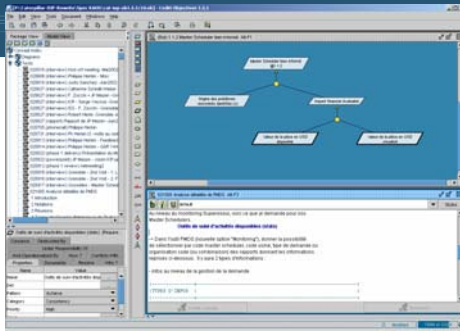
Involved organizations

Objectiver



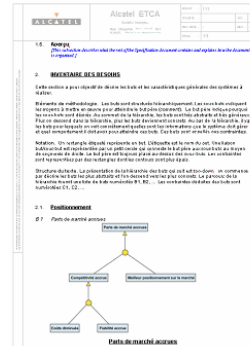
Main Views

Objectiver

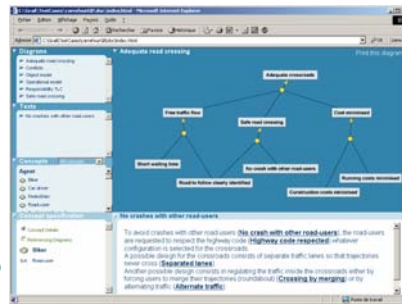


Model editor

Generated requirements document

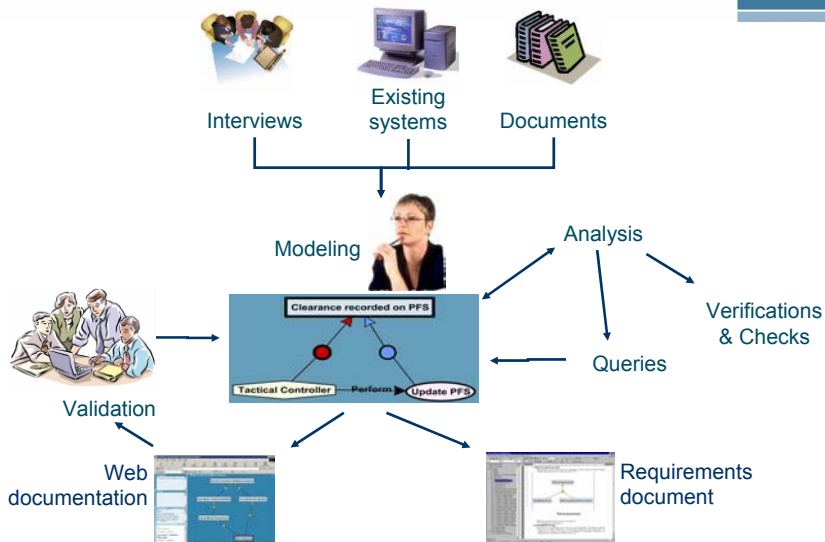


Generated web documentation



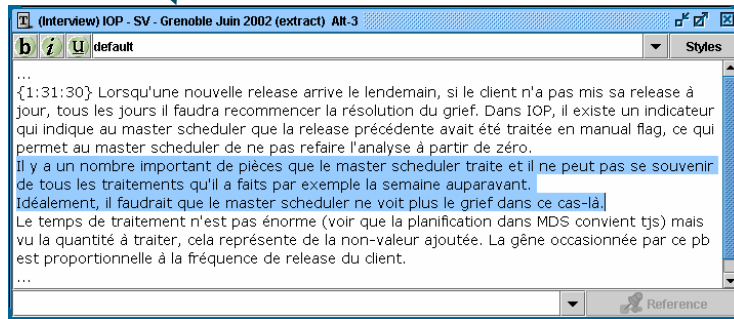
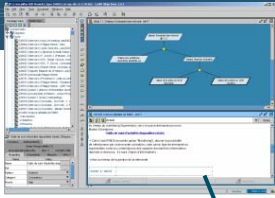
Process

Objectiver



1> Interviews

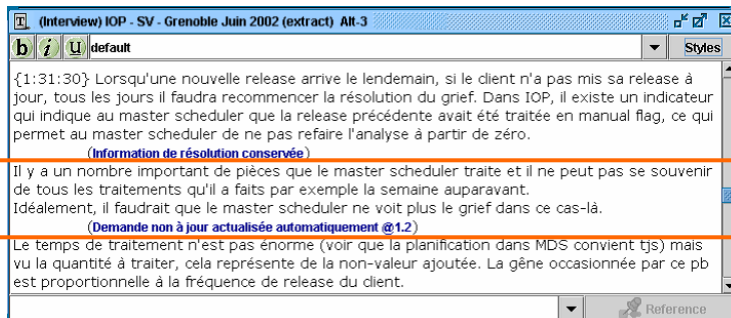
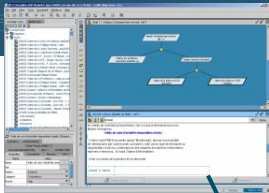
Objectiver



IOP Rewrite Project

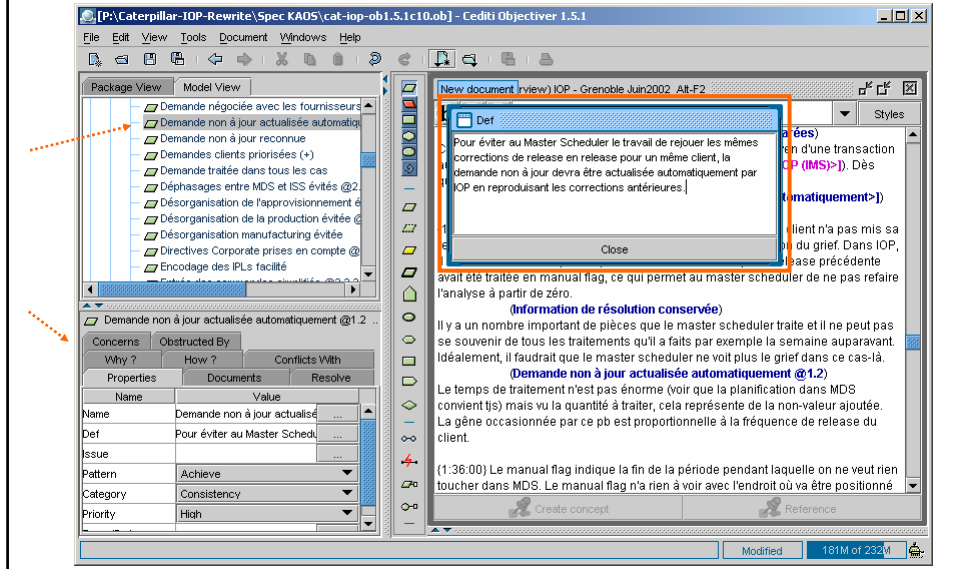
2> Identify Goals

Objectiver



3> Specify Goal Properties

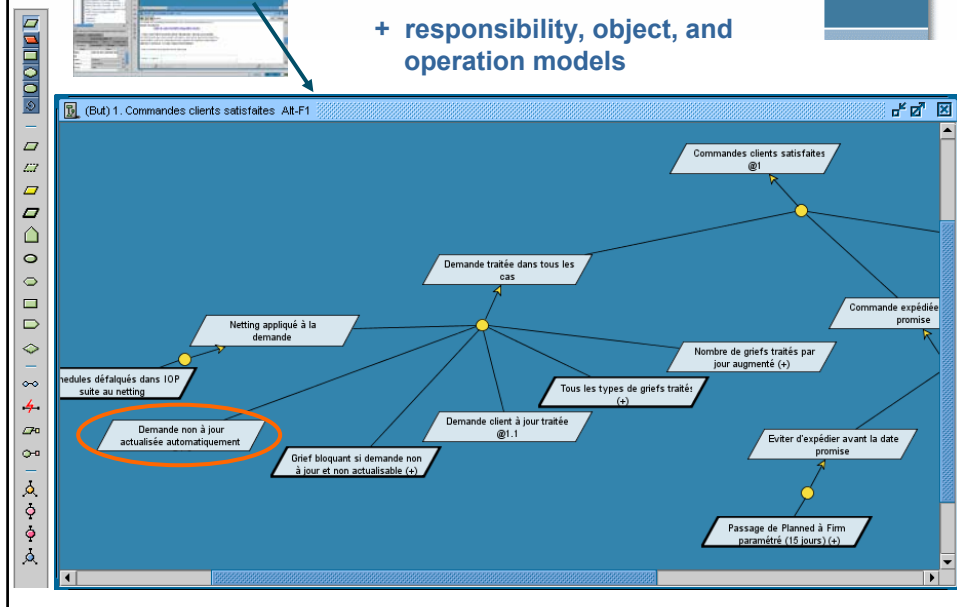
Objectiver



4> Build Goal Model

Objectiver

+ responsibility, object, and operation models



5> Generate Reqs Document

Objectiver

The screenshot shows a software interface titled '(CC) 1. Commandes clients satisfaites Alt-0'. It features a diagram with nodes and arrows. One node, 'Demande non à jour actualisée automatiquement @1.2', is circled in orange. An arrow points from this node to a text box below the diagram. The text box contains the following text:

Pour éviter au Master Scheduler le travail de rejouer les mêmes corrections de release en release pour un même client, la demande non à jour devra être actualisée automatiquement par IOP en reproduisant les corrections antérieures.

Below the diagram, there is a section titled 'Informal Def' with a 'Close' button. The text below this section reads:

Comme illustré par le diagramme (Objet 1. Commandes clients satisfaites),
 Commandes clients satisfaites @1 si (a) Demande traitée dans tous les cas, (b) Commande expédiée à la date promise et (c) Temps de traitement

a. Demande traitée dans tous les cas
 Avant de mettre à jour le système de gestion de la demande acceptée (MDS) sur base duquel le plan de production est établi

Le traitement de la demande doit comporter une phase de netting des nouveaux carnets de commande. Les releases client

R* Schedules défalqués dans IOP suite au netting
 IOP opérera le netting de la demande entrante. Le netting consiste à défalquer les schedules les plus récents à concurrence des quantités en transit. Les quantités en transit sont calculées par IOP comme étant la différence entre le cumul expédié (ACCUM SHP disponible dans MDS) et le cumul reçu par le client (ACCUM REC disponible dans la release client).

Pour éviter au Master Scheduler le travail de rejouer les mêmes corrections de release pour un même client, la demande non à jour devra être actualisée automatiquement par IOP en reproduisant les corrections antérieures (Demande non à jour actualisée automatiquement @1.2).

6> Amend Reqs Document

Objectiver

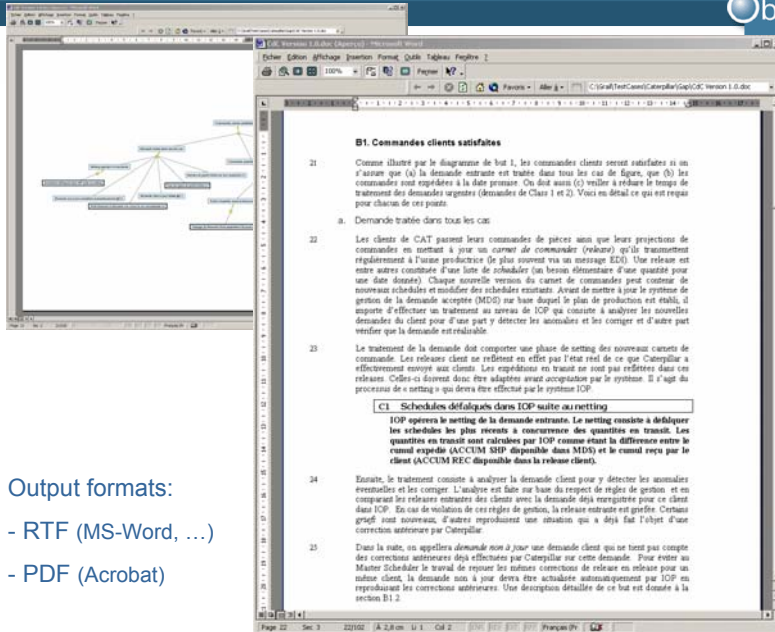
The screenshot shows a software interface titled '(CC) 1. Commandes clients satisfaites Alt-1'. It features a diagram with nodes and arrows. One node, 'Demande non à jour actualisée automatiquement @1.2', is circled in orange. An arrow points from this node to a text box below the diagram. The text box contains the following text:

Ensuite, le traitement consiste à analyser la demande client pour y détecter les anomalies éventuelles et les corriger. L'analyse est faite sur base du respect de règles de gestion et en comparant les releases entrantes des clients avec la demande déjà enregistrée pour ce client dans IOP. En cas de violation de ces règles de gestion, la release entrante est grieffée. Certains griefs sont nouveaux, d'autres reproduisent une situation qui a déjà fait l'objet d'une correction antérieure par Caterpillar.

Dans la suite, on appellera demande non à jour une demande client qui ne tient pas compte des corrections antérieures déjà effectuées par Caterpillar sur cette demande. Pour éviter au Master Scheduler le travail de rejouer les mêmes corrections de release en release pour un même client, la demande non à jour devra être actualisée automatiquement par IOP en reproduisant les corrections antérieures. Une description détaillée de ce but est donnée à la section (Objet 1.2 Demande client non à jour actualisée automatiquement).

7> Publish Reqs Document

Objectiver



Output formats:

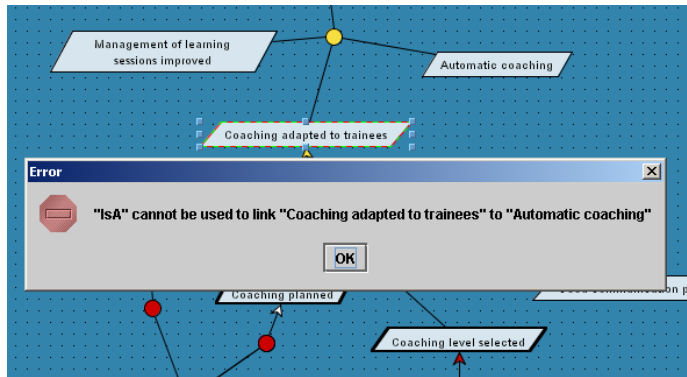
- RTF (MS-Word, ...)
- PDF (Acrobat)

Support for Model Checking & Validation

Objectiver

- Automatic Model Validation
- Extensible manually-triggered OQL queries
- Report & Diagram generation
- Validation based on the generated Web documentation

Conformance to KAOS syntax verified on the fly

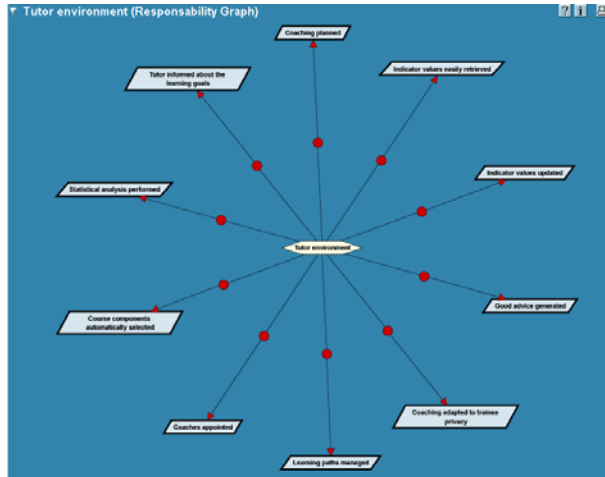


OQL-Supported Model Checking

- Query Language (OQL extension)
- Model completeness checking
 - goals not refined (wishful thinking)
 - requirements without a responsible agent
 - requirements not “operationalized”
 - unresolved obstacles
 - ...
- Model quality checking
 - concepts not used in any document
 - incomplete concept specification
 - concepts with unresolved *Issues*
 - ...

Diagram generation

Responsibility model
generated
based on the
goal model



Validation with experts : model browser

Indexes

Diagram viewer

Property viewer

Text viewer

The screenshot shows a web-based interface for a model browser. On the left, there are three panels: 'Documents' (listing goal diagrams, interviews, object model, and responsibility diagrams), 'Concept index' (listing concept index, goal diagrams, interviews, object model, and responsibility diagrams), and 'Concepts by type' (listing concepts by type, agent, association, conflict, entity, and goal). Below these is a 'Concept specification' panel with tabs for 'Concept Details', 'Referencing Diagrams', and 'Achieve Author environment'. The main area displays a diagram titled 'Learning efficiently organized' with a central node connected to four peripheral nodes: 'Trainee knowledge improved', 'Quality of learning improved', 'Education costs minimized', and 'Attractive learning environment provided'. A red star is placed on the 'Attractive learning environment provided' node. At the bottom, a 'Text viewer' panel shows a 'Summary' section with a paragraph about parts (Part) and a list of parts: 'lecture part', 'demonstration part', 'test part', and 'game part'.

Concept references are hyperlinks

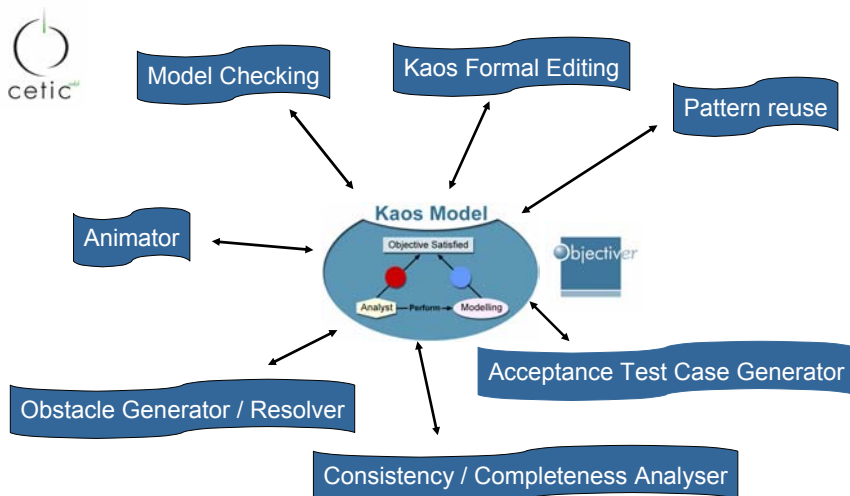
Should I use Objectiver ?

Objectiver

- Elicit and specify requirements in a **systematic** way
- Produce structured, motivated, easy to understand requirements documents
- Facilitate **communication**
- Simplify **validation** process
- Provide **traceability** from:
 - Requirements to goals
 - Operations to requirements

cetic
FAUST Project

Objectiver



Cediti References

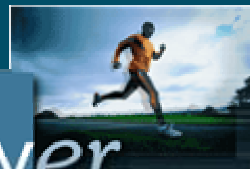
Objectiver

- Requirements analyses
& Reengineering
- Requirements documents
- Strategical plans & IT plans



A power tool
to engineer your business
and technical requirements

Objectiver



Get your free trial version now !

www.objectiver.com