Goal-driven Requirements Analysis and Design for Process Control Systems

Islam A. M. El-Maddah and Tom S. E. Maibaum {elmaddah , tom}@dcs.kcl.ac.uk, Department of Computer Science, King's College London

KING'S College LONDON Founded 1829 University of London

Could You start me off, please?

Sure, I can offer ready templates (high-level goal-models) that accommodate your needs

What about the components of my system ?

Do not worry, you can import their low-level models from the GOPCSD library
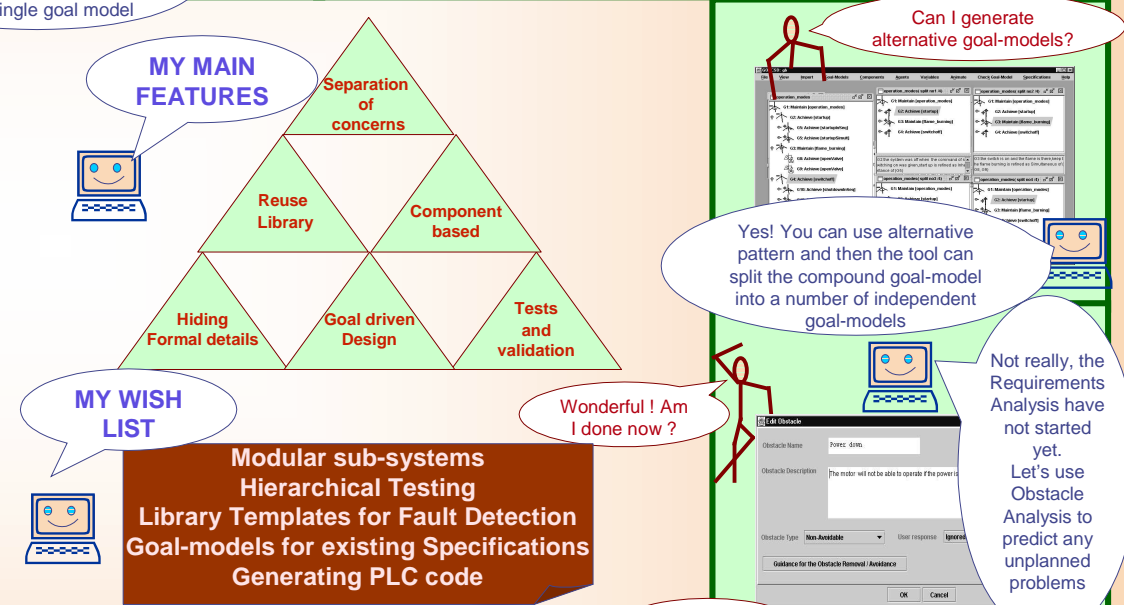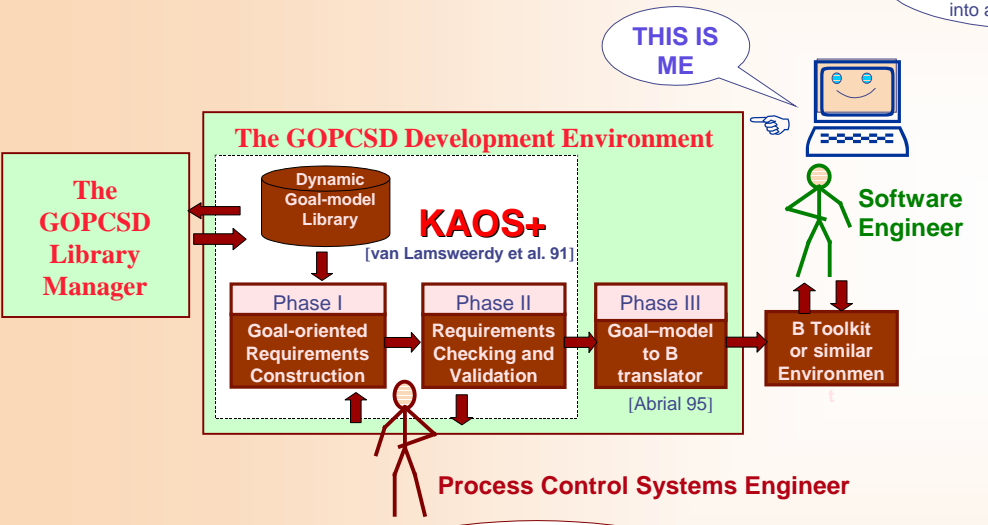
What do I have to do now?

Refine the high-level goals and combine them into a single goal model

Do I need much effort to formalise the goal-model?

No, I offer a predictive parser and spell checker

Can I check my goal-model as I am building it?

Sure, I can offer Reasoning and Investigation tools

THIS IS ME

MY MAIN FEATURES

Can I generate alternative goal-models?

The GOPCSD Development Environment

Dynamic Goal-model Library

KAOS+
[van Lamsweerdy et al. 91]

Software Engineer

The GOPCSD Library Manager

Phase I
Goal-oriented Requirements Construction

Phase II
Requirements Checking and Validation

Phase III
Goal–model to B translator
[Abrial 95]

B Toolkit or similar Environmen

Process Control Systems Engineer

Separation of concerns

Reuse Library

Component based

Hiding Formal details

Goal driven Design

Tests and validation

MY WISH LIST

Modular sub-systems
Hierarchical Testing
Library Templates for Fault Detection
Goal-models for existing Specifications
Generating PLC code

Wonderful ! Am I done now ?

Yes! You can use alternative pattern and then the tool can split the compound goal-model into a number of independent goal-models

Not really, the Requirements Analysis have not started yet. Let's use Obstacle Analysis to predict any unplanned problems

That sounds good. But after my requirements model is satisfactory, I do not know much about B.

No worry. I'll automatically generate B machines.

What next !

Keep the goal model for future access. You'll probably extend/maintain your system

The generated B machines are documented by tracing them to the informal Requirements.

Now, I can take over. The stage is just ready for me

Do you have any clue as how to remove the detected bugs?

I will highlight the suspect goals and guide you through what you could do

I wish there was a way for the tool to tell me how the system will behave at runtime.

Your wish comes true! A chance to animate and enhance the system

Any Formal Analysis ?

Do you not need to check the consistency and completeness of your requirements ?