Apollo

User specification

**confidential**

Revision number: 1

Date:

Name: P. van der Velde

# Disclaimer

This specification is not, by any stretch of the imagination, complete. It will need to be revised several times before it is complete. Currently several major parts are either missing or incomplete. This disclaimer will be updated to reflect any change in these sections. Finally a specification is supposed to be a ‘living’ document and therefore never complete. What were you thinking; better learn to live with this fact.

# Goal

The goal of Apollo is to provide engineers and designers with the capability to evaluate and change the behaviour of geometry based designs / objects in different physical situations. Apollo achieves this goal by allowing users to create different data sets on a design. These data sets can hold data describing one or more physical models. The data can either be generated by a simulation or an experiment. Apollo is able to process the data in the data set and finally store and display the processed data.

Figure 1 shows a mind map of the goals for the Apollo project.

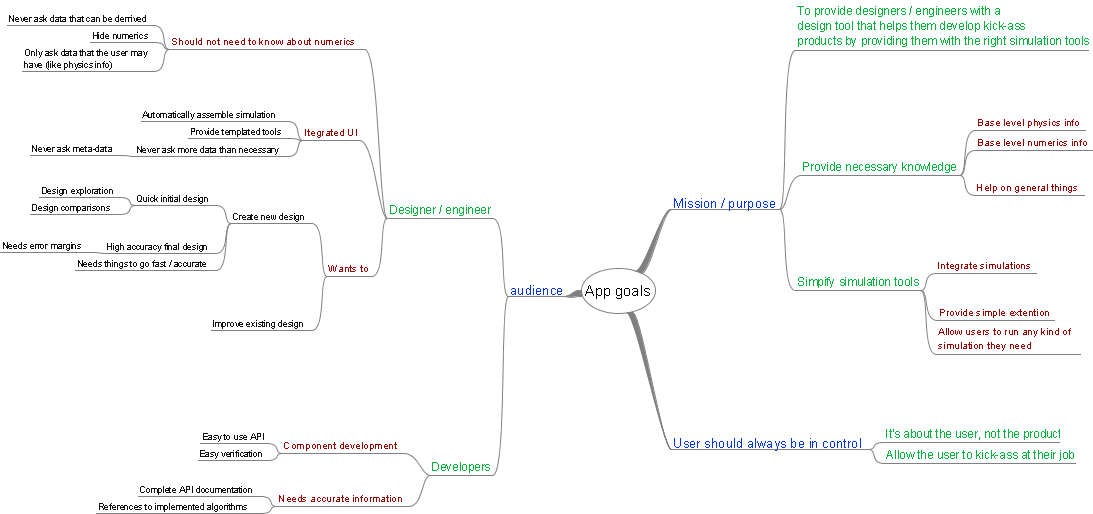


Figure : Mind map of the goals of Apollo

# Introduction

Apollo will support engineers, designers, architects and other professionals who need to evaluate and improve geometry based designs by providing them with a tool that allows them to study the physical behaviour of the design. Apollo will achieve this goal by providing the user with the ability to process data sets which are based on one or more physical models on the design. These data sets can hold virtual (i.e. simulations) and real data.

During the design process the user may need to investigate different kinds of physical behaviour, ranging from simple cases where only a single physical model plays a role to more complicated cases where multiple physical models interact. Apollo will provide the tools to create and store simple and complex data sets as well as providing as many different physical models as possible. However if a physical model is not known to Apollo it should be possible for the user (or others) to add tools that allow Apollo to work with the desired physical model.

Because data sets, both virtual and real, can quickly become complicated one of the main goals of Apollo is to simplify that part of the design process by providing easy setup and execution of simulation calculations and easy processing of data resulting from either simulations or experiments. The idea is to make simple things automatic and complicated things easy. The user interface should be as simple as possible (but no simpler) but still provide access to more powerful features for the complicated scenarios.

# Users

Apollo is aimed at two different groups of users. The first group consists of engineers, architects, designers and other design related professionals who will use Apollo to evaluate the behaviour of different designs. The second group consists of software developers who will develop new tools for Apollo.

## The designer

The design related professionals will use Apollo to create new designs, possibly based on existing designs. For this task they require that Apollo provides them with tools that support their design process. Apollo should allow users to create several concepts quickly and use an iterative design process to arrive at a final design. This means that Apollo should provide the capability to perform design explorations and comparisons between different concepts. Apollo should allow highly accurate evaluations of the final concepts, by providing tools that allow optimization of the final design while providing certainty levels during each step.

While Apollo provides users with powerful tools it should never the less take a backseat to the user’s final goal. Apollo should never get in the way of the design or analysis work. This means that Apollo should perform as many of the menial tasks automatically and without asking the user for any more information than strictly required. The user should only need to provide data that is rooted in the physical domain of the experiment. From there Apollo should be able to determine all the other required data.

Apollo should ensure that users who only have knowledge of the physical aspects of the modelling can still perform generate the required data successfully. This means that Apollo should hide as much of the calculation side of the data generation process as possible but still provide advanced users with the possibility to control these parts.

## The developer

Apollo will provide an API (Application Programming Interface) for developers who wish to provide additional tools for Apollo. The API will provide developers with the capability to extend Apollo in a few key areas. The extensibility points of Apollo should allow developers to create additional tooling that provides new data set sub-elements, data processing and visualization, data import & export filters and additional data sources.

In order to support development of new tools for Apollo an easy to use API should be provided. This API should guide developers to the best way to develop tools for Apollo, i.e. the API should make the developer ‘fall into the pit of succes’.

# What we’re not doing

While Apollo will be a powerful system there are a number of things it will, by design, not be capable of. Note that this list will probably grow longer and longer as time goes by. The idea for Apollo is to keep it concentrated on a specific area so that it may be good at everything in that specific area.

* Apollo will not define a geometry manipulation API. Geometry manipulation should be done by a Computer Aided Design (CAD) application. To provide geometry storage capability Apollo may be running inside the CAD application as plug-in or side-by-side with the CAD application.
* Apollo will not provide physics modelling for every possible situation. The current interest lies in the fields of heat, structures and fluids. At later stages additional physical models may be added or removed.
* Apollo will not support multiple concurrent users working on the same model. All projects and data sets will only be accessible by a single user at the time.
* Apollo will not allow users to make changes to a running data set. In order to change or update the setup for a data set the user will have to stop processing the data set and make the changes. Apollo may allow users to pause data set processing in order to make changes.

# Use cases

The following use cases will describe the way Apollo can be used by designers. Currently four different use cases have been prepared. These will describe the high level workflow experienced by the designer as they work with Apollo. Parts of the workflow will be described in more detail in later sections.

The different use cases provided below describe:

* Robust design. A workflow that allows the designer to design a product which is able to perform optimally while being tolerant to changes in geometry or environmental conditions.
* What-if research or design. A workflow that allows the designer to investigate the response of a design to changes in geometry or environmental conditions.
* Optimization of existing or new products. A workflow that allows designers to optimize an existing or new design. Note that this workflow is essentially a more simplified version of the robust design workflow.
* General research. A workflow that allows designers to perform general research into the response of a design. Note that this workflow is a more general version of the what-if workflow.

While the workflows are described as separate cases it should be possible for the designer to switch between the different workflows at any point during the design process. The use of workflows while allowing easy switching between workflows will be achieved by providing support for the different workflow scenario’s through suggestions in the user interface. No information will be stored about the actual workflow being followed by the designer.

## Robust design

In robust design the designer is interested in developing a product which is able to perform optimally within a specific range of geometric and environmental conditions. The main goals for the designer are:

* Create an optimal design with which is tolerant to changes in conditions.
* Control the data processing by specifying one or more parameters manually or by fixing the range of one or more parameters.
* Generate the final design as quick as possible with known accuracy of the final solution.

In response to these goals Apollo should be able to:

* Allow user to specify tolerances and error levels which are allowed for the final solution. Furthermore Apollo should allow users to specify the variance on the input values. These variances should be taken into account when processing the data. While the user is probably only interested in the final result there could be some interest in learning why this solution is the best so Apollo needs to store all the data generated during the processing of the model.
* Allow users to specify a range of values for the free parameters. Once the range is specified Apollo should be able to determine for which values the data processing shall be performed. The user may desire to have more or less control over this selection process so appropriate amount of control should be provided to the user.
* Allow the user to select the desired combination of physics models. Apollo should try to hide as much of the unnecessary details of the selected models. Users should normally not need to select the exact model even though this should be possible.
* Reach the final solution quickly and with a known accuracy, i.e. the user should know what the tolerances are on the solution.
* Allow the user to specify how the results should be presented, for instance in tabular format, on a graph or other method.
* Allow the user to add data from other projects or external sources for comparison purposes.

Further steps from robust design may be:

* ?

## What-if research

In what-if research the designer is interested in finding out what the effects of a certain change to the geometry or initial conditions will be. In this case the main goals for the designer are:

* To create or import the original geometry and starting conditions.
* To quickly assess the influences of changes to the original design.
* Put a design through different parameter studies to see what the differences between the different cases are. Normally the designer will only be interested in the result, i.e. the influences of the different parameters; however the calculation data may be used later on for further calculations.

In response to this Apollo should be able to:

* Allow the user to specify new parameters to track. A parameter maybe defined in the data set or it may be a new parameter with specific calculations to perform.
* Allow the user to specify the range for parameters.
* Allow the user to specify how the results should be presented, for instance in tabular format, on a graph or other method.
* Allow the user to add data from other projects or external sources for comparison purposes.
* Reach the final solution quickly and with a known accuracy, i.e. the user should know what the tolerances are on the solution.

Further steps from what-if research may be:

* An optimisation calculation where the unimportant parameters are left out
* A robust design calculation where only the important parameters are given tolerances
* A research calculation for cases which show unexpected behaviour

## Optimization

When performing an optimization the designer is interested in finding the most optimal design given a set of specific restrictions. This case is a special case of the robust design case.

## Research

When performing research the designer is interested in discovering the operating principles behind the performance of a given design. In this case the main goals of the designer are:

* To creation or import the original geometry and initial conditions
* To set up the model for:
  + General reviewing of a specific model in a specific situation
  + General reviewing of several models in a specific situation
  + General review of a specific model in multiple situations etc.
* Have exact control over the way the data set is processed

In response Apollo should be able to:

* Allow the user to specify exactly how the data is processed
* Allow the user easily to add, change or upgrade the geometry or initial conditions
* Allow the user to specify how the results should be presented, for instance in tabular format, on a graph or other method.
* Allow the user to add data from other projects or external sources for comparison purposes.
* Reach the final solution quickly and with a known accuracy, i.e. the user should know what the tolerances are on the solution.