AQUA
EXCEL
2020

AQUAculture infrastructures for EXCELlence
in European fish research towards 2020 —
AQUAEXCEL2020

Deliverable D5.4 First prototype flow field model established

Morten Omholt Alver

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 652831. This publication reflects only the view of the author, and the European Commission cannot be held responsible for any use which may be made of the information contained therein

Executive Summary

Objectives:
The purpose of this document is to describe the functionality and technical implementation of the flow field model. This numerical model prototype is one of the main components in the AQUAEXCEL^{2020} virtual laboratory, which will be developed in WP5: "Virtual laboratories and modelling tools for designing experiments in aquaculture research facilities".

The main components are:

- Growth, nutrition and waste production models for different fish species
- Water quality and water treatment modelling
- Modelling of hydrodynamic flow fields in tanks and cages

The objective of the flow field model is to represent the water currents within the production unit - fish cage or tank, presenting key information relating to the current to the other model components.

Rationale:
One of the main research activities in AQUAEXCEL^{2020} is to develop a virtual laboratory system that enables virtual experiments in aquaculture research facilities. This system will feature a framework that allows the integration of mathematical models of different subsystems in common simulations, replicating the system operation of research laboratories.

Main Results:
The flow field prototype model is developed and tested, and it is shown that this model component can be integrated with the other main components. The flow field model uses one approach for current in tanks – precomputed flow fields from a CFD model – and another for open sea cages – current profiles depending on ambient current conditions. The model interacts with the other model components either through providing the current speed and direction vector for given locations, or through providing descriptive numbers for the overall flow field in the production unit.

Authors/Teams involved:
Morten Omholt Alver (NTNU)
Martin Føre, Gunnar Senneset (SINTEF)
Štěpán Papáček (Univ. of South Bohemia)

Table of contents

1. BACKGROUND

This document is part of the AQUAEXCEL^{2020}, WP5/Joint Research Activity 1 – Virtual laboratories and modelling tools for designing experiments in aquaculture research facilities.

Experiments with fish usually involve extensive use of laboratory facilities and run for long periods of time. Both from an ethical perspective (3R's) and from a cost perspective, tools for design and planning of experiments are increasingly important. In aquaculture research as well as other domains, numerical models are increasingly used preparatory to the actual experiments.

One of the main research activities in AQUAEXCEL^{2020} is to develop a virtual laboratory system that enables virtual experiments in aquaculture research facilities. This system will feature a framework (see Bjørnson et al., 2016) that allows the integration of mathematical models of different subsystems in common simulations, replicating the system operation of research laboratories. The overall system architecture is shown in Figure 1.

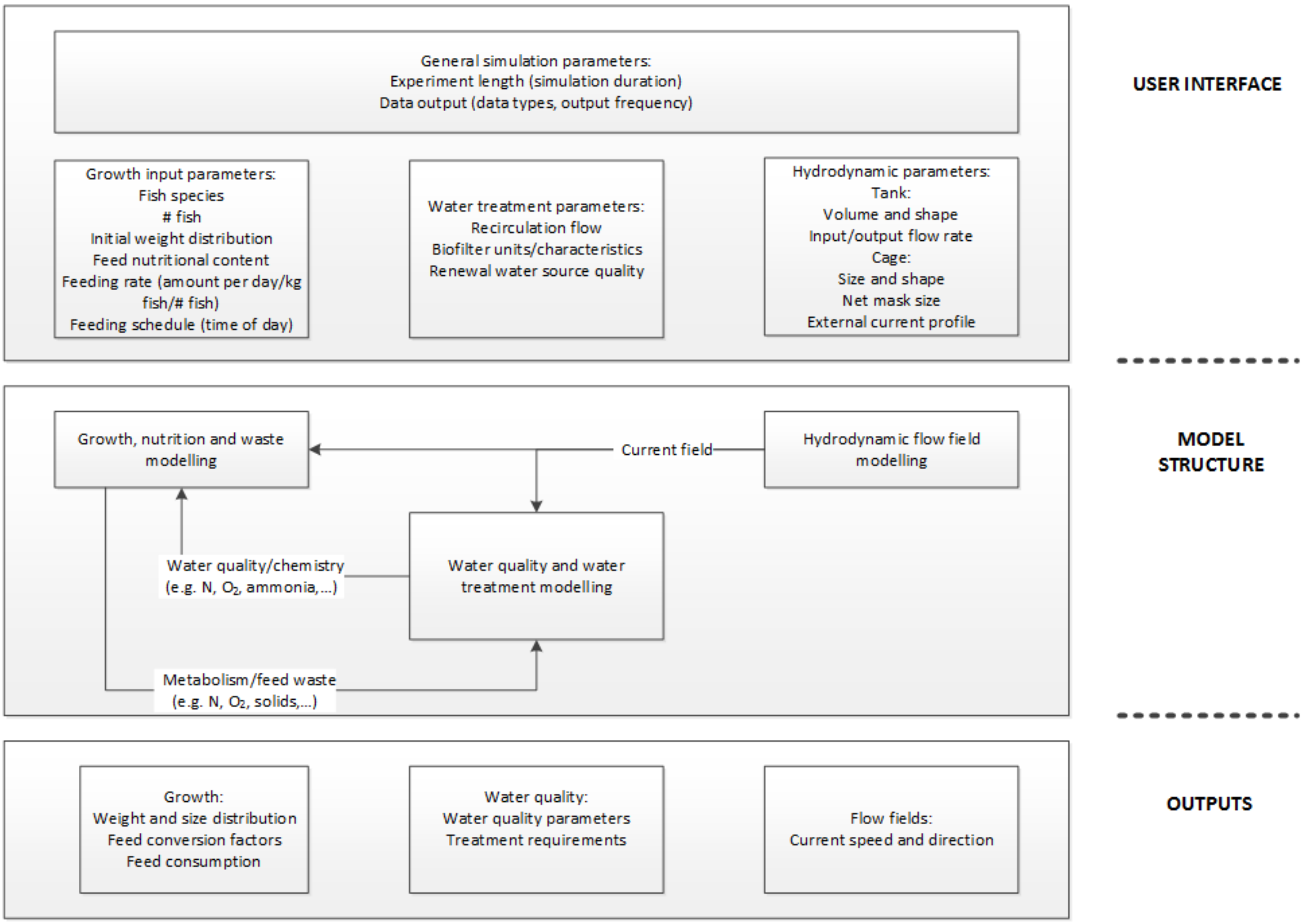

Figure 1 Virtual laboratory system architecture

This document describes the technical implementation and functionality of the flow field model.

2. MODEL DESCRIPTION

Background

The flow field of production units, either tanks or net cages, may affect the fish in a number of ways. Most obviously, the current at the location of each fish represents a motion superimposed on the fish's own active movement. The current also serves to transport and mix particulate and dissolved matter in the water, such as feed pellets and concentrations of oxygen and ammonia, all of which may have effects on fish welfare and performance.

The flow fields are very different in tanks and in net cages. In tanks, the walls are closed so water only enters and leaves the tank through designated water inlets and outlets. In net cages, water flows through the cage sides, causing a net transport across the cage, typically supplemented by complex turbulent variability caused by the cage netting and the fish movement. In case of strong current, the fish may be forced to swim up against the current to keep their position in the cage. Tanks and net cages need to be handled using different tools, and in both cases exact currents are hard to measure, and hard to model mathematically.

Flow fields in tanks

Flow fields in tanks, disregarding the effect of the fish, can be computed using Computational Fluid Dynamics models (e.g. Sayma *et al.* 2009). The fundamental basis of these models is the Navier-Stokes equations which are used to describe the motion of viscous fluids. CFD models solve a form of Navier-Stokes numerically after discretizing the simulation volume. CFD models typically use non-uniform discretization schemes where the spatial resolution varies over the simulated volume, with higher resolution around features such as inlets. The process of designing the model grid is a separate operation that may require a large number of computations. Numerical solution of the equations on the model grid is also computationally demanding, and models are typically parallelized and run on high performance computing centers. The output may be in the form of time-varying 3D flow fields. For the purpose of flow fields in tanks, we are looking for the steady-state field for a given tank setup and water in-flow rate, so the CFD model will be run to produce a constant 3D field representing steady state flow (see example flow field in Figure 2). The field may be accompanied by a computed turbulence kinetic energy field describing the level of turbulence estimated over the volume.

There is a wide selection of open source software (e.g. OpenFoam, Typhon and Reef3D) and commercial software (e.g. Ansys Fluent, CFX and COMSOL) that basically solve the same mathematical equations, but have different features and different learning curves.

CFD models do not have the option of simulating the effect of fish swimming activity on the flow field. However, by making assumptions of simplified fish swimming patterns (e.g. with the fish swimming in circular motion against the current direction at their preferred speed) one may include the effects of these in order to make the CFD model estimates more precise for a tank with fish.

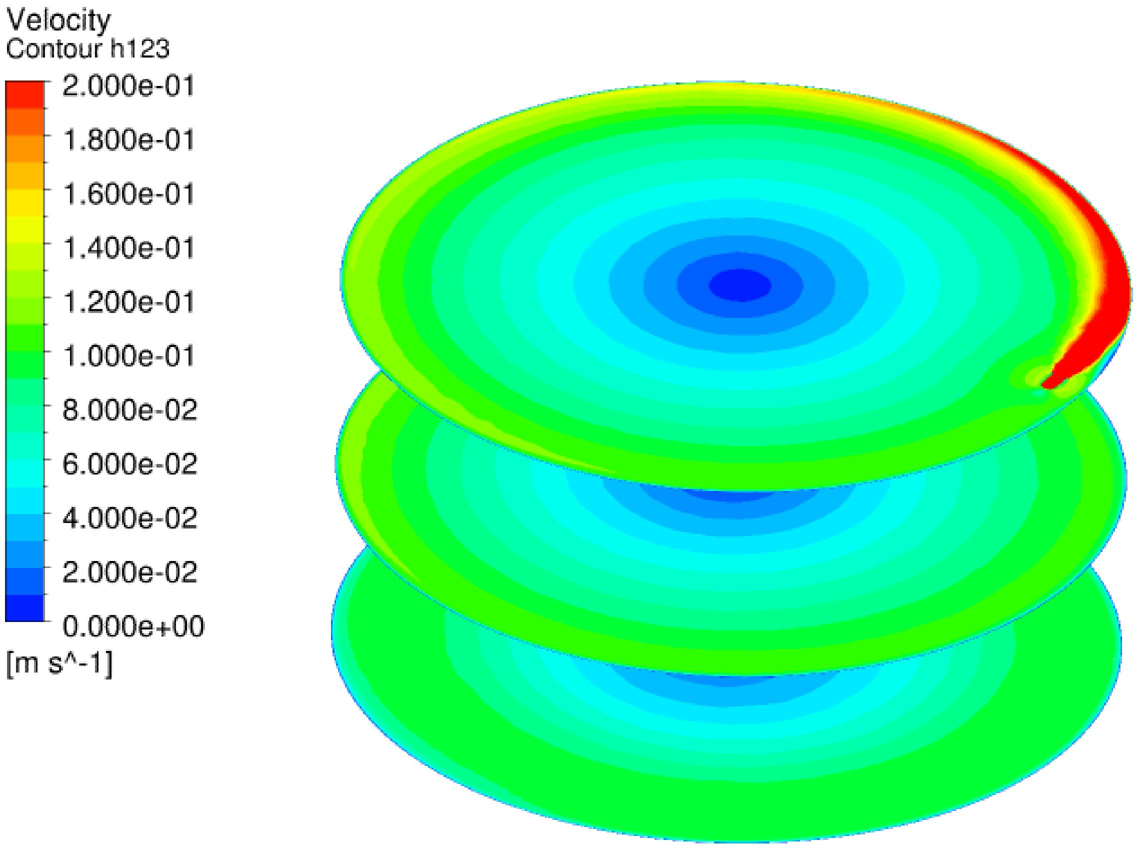

Figure 2: Example flow field in tank, computed using Ansys Fluent. The figure shows velocity contours in three planes above the tank bottom. The upper contour plot shows the higher velocities originating from a water inlet. The other dominating features are a lower current velocity closer to the centre of the tank, and a slight reduction of current near the tank wall except those areas affected by the water inlet.

Flow fields in cages

In the case of cages, no established models exist to simulate currents as function of outside current and the activity of the fish. The current outside the cage at a given location may be monitored using point or profiling current meters, and modelled using regional ocean models such as SINMOD (Slagstad & McClimans 2005) or ROMS (Shchepetkin & McWilliams 2005). The vertical current profiles observed at an aquaculture location depend on which processes dominate the dynamics at the location. Some of the relevant processes are tidal dynamics, wind induced currents and stratification caused by freshwater run-off. Aquaculture farms may generally be categorized according to the type of current and hydrography conditions they experience. In the Norwegian salmon industry, for instance, there are some sheltered fjord locations, with limited wind and waves, high stratification and mainly tidal currents and estuarine circulation, and more exposed locations with stronger wind and wave effects, less pronounced tidal currents and with current influenced by the Norwegian Coastal Current.

The current speed inside of the cage is dependent on outside current and the properties of the cage (Løland 1993). The outside current for each individual cage in a fish farm depends also on the cage's position, as the wake from upstream cages will affect the current speed. Cages deform in strong currents, making the dynamics more complicated (see Figure 3). Løland (1993) developed a method for calculating the current forces on a net structure and the resulting wake, including estimates of current inside the cage, and compares results to laboratory scale model test. An approximate inside current speed of 85% of the ambient current speed is found for a reasonable set of parameters. Endresen et al (2013) uses a net cage model implemented in SINTEF's FhSim modelling framework to estimate wake effects acting within an aquaculture net cage, comparing numerical estimates with experimental data. An approximate inside current speed of 80-85% of the ambient current speed is found.

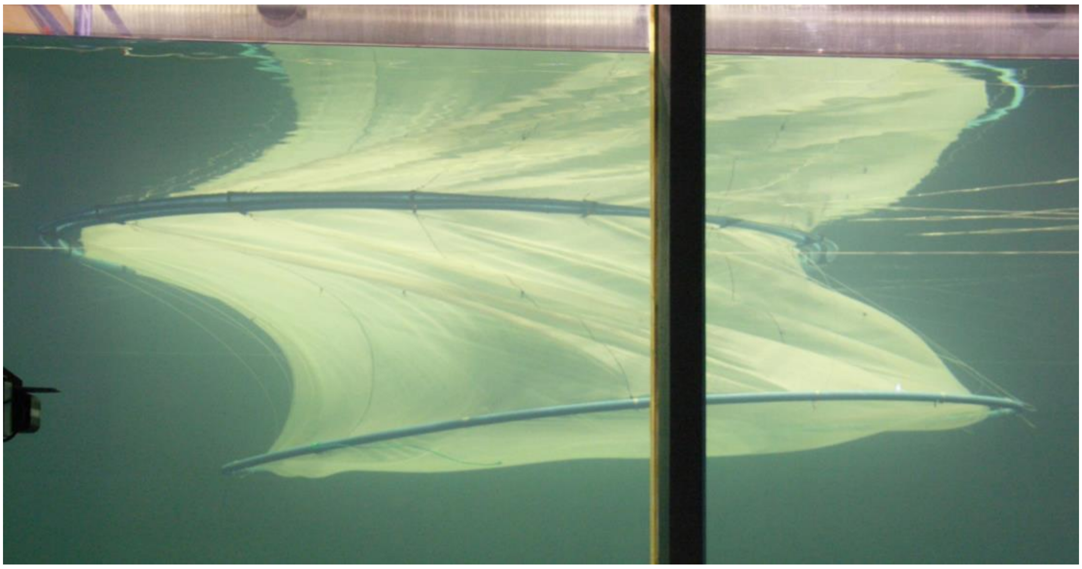

Figure 3: Photograph of downscaled sea cage model in a laminar current field (Photo: SINTEF Ocean).

As with CFD modelling of tanks, the effect of the fish is difficult to account for. Models describing the current flow field inside net cages in more detail may become available through ongoing research, but at present the best approach is to do a simple parameterization based on the ambient current profile to estimate the flow field inside.

Model functionality

In the case of tanks, due to the computational cost, computing a flow field based on user inputs is time consuming and requires high performance hardware. Therefore, it is not practical to run the current model on demand through the virtual laboratory web interface. The solution is to provide precomputed flow fields for a selection of tank designs and operating conditions. These precomputed fields are stored in NetCDF files[1] structured in a way suitable for the flow field model to read. As long as the output is stored or convertible to a suitable format, any CFD model may be used for this purpose.

The precomputed flow fields will be available to the user for selection. The web interface may include a mechanism for users to request runs of new tank designs.

In the case of cages, the flow field will be estimated based on outside current conditions in the form of vertical profiles of current direction and speed. The user will be able to either choose and adjust typical profiles, or input their own. The vertical current profile to use will be written in NetCDF format for access by the model.

In either case, the key functionality of the flow field model component is to provide information about the water current in the tank or cage. The model operates in two different modes – one that provides 3D current components (e.g. current upwards, towards the east and towards the north) at requested position, and another that provides overall descriptive values characterizing the flow field.

---

[1] For information about the NetCDF file format, see https://www.unidata.ucar.edu/software/netcdf/

Model interface

The model interface is different depending on in which mode it is running. However, in both modes the initialization of the model requires a path to a data file in NetCDF format containing the precomputed flow field or current profile. This file will be opened upon startup of the model, and will be accessed through the course of the simulation to extract the information needed to compute the requested model output values.

Table 1 lists the initialization parameters required by the model. This list may be expanded in future versions of the model.

Table 1: Model initialization parameters

Name	***Type***	***Data type***	***Description***
unitType	Input	Category	Tank or cage
mode	Input	Integer	Mode 1 (current at positions) or 2 (current descriptors)
filename	Input	String	Path to NetCDF file containing flow field or current profile

Mode 1: Current at positions
In this mode, the input values to the model are the x, y and z coordinates of the requested position, and the output is the current at that position. Input and output values are listed in Table 2. This list may be expanded in future versions of the model.

Table 2: Input and output values in mode 1

Name	***Type***	***Unit***	***Description***
xPos	Input	m	Requested position along the *x* axis
yPos	Input	m	Requested position along the *y* axis
zPos	Input	m	Requested position along the *z* axis
xVel	Output	m/s	Current speed along the *x* axis
yVel	Output	m/s	Current speed along the *y* axis
zVel	Output	m/s	Current speed along the *z* axis

Mode 2: Current descriptors
In this mode, no input values are required. The module outputs a selection of descriptive values for the flow field. Input and output values are listed in

Table 3. This list may be expanded in future versions of the model.

Table 3: Input and output values in mode 2. Values marked with (*) are only available if they can be read or calculated from the input NetCDF file.

Name	***Type***	***Unit***	***Description***
vAvg	Output	m/s	Average speed
vMin	Output	m/s	Minimum speed
vMax	Output	m/s	Maximum speed
minX	Output	m	X position of minimum speed
minY	Output	m	Y position of minimum speed
minZ	Output	m	Z position of minimum speed
maxX	Output	m	X position of maximum speed
maxY	Output	m	Y position of maximum speed
maxZ	Output	m	Z position of maximum speed
*avgTke**	Output	J/kg	Average turbulence kinetic energy
*minTke**	Output	J/kg	Minimum turbulence kinetic energy
*maxTke**	Output	J/kg	Maximum turbulence kinetic energy
*minTkeX**	Output	m	X position of minimum TKE
*minTkeY**	Output	m	Y position of minimum TKE
*minTkeZ**	Output	m	Z position of minimum TKE
*maxTkeX**	Output	m	X position of maximum TKE
*maxTkeY**	Output	m	Y position of maximum TKE
*maxTkeZ**	Output	m	Z position of maximum TKE

Technical implementation

The flow field model component is written in C++ and packaged as an FMU. The model utilizes the NetCDF 4 library[2] which provides functionality for reading and writing NetCDF files.

Model initialization
At startup, the FMU opens the NetCDF file and reads the metadata defining the grid or the vertical profile. The NetCDF format allows for metadata describing units and the geometry of the data contained. Provided the input NetCDF file is well formatted, no additional information is needed for the model to relate the flow field to the tank geometry. The layout of a NetCDF file can be shown using the CDL (network Common data form Description Language) format, in which file dimensions and variables with attributes are listed. The following listing shows the CDL of a file containing a three dimensional current flow field (given as components *u*, *v* and *w*), with metadata providing grid and variable units:

---

[2] NetCDF libraries are open source and can be downloaded at https://www.unidata.ucar.edu/downloads/netcdf/index.jsp.

```
netcdf datafile {
dimensions:
        xc = 400 ;
        yc = 400 ;
        zc = 200 ;
variables:
     float xc(xc) ;
          xc:units = "m" ;
     float yc(yc) ;
          yc:units = "m" ;
     float zc(zc) ;
          xc:units = "m" ;
     float u(zc, yc, xc) ;
          u:units = "m/s"
     float v(zc, yc, xc) ;
          v:units = "m/s"
     float w(zc, yc, xc) ;
          w:units = "m/s"
}
```

When running in mode 1, the vertical profile is read upon startup in the case of net cages, and stored in memory. In the case of tanks, no further data is read upon initialization of the model. When running in mode 2, the output values are calculated upon initialization of the FMU and stored for output in all subsequent time steps.

Calculation of output values in mode 1

When the FMI master calls for the model to advance to the next time step, new *xPos*, *yPos* and *zPos* values are provided. In the case of net cages, the *zPos* value is used to extract the appropriate current vector from the vertical profile, and the vector components *xVel*, *yVel* and *zVel* are returned. In the case of tanks, the appropriate data cell is found based on the given coordinates, and the current velocity of the cell is read from the NetCDF file. If the file provides vector components, these are returned directly. If the file provides axial, radial and vertical velocities, vector components are calculated from these and returned.

Calculation of output values in mode 2

To calculate current descriptor variables, the entire flow field is read from the NetCDF file, and the appropriate calculations are performed, such as finding the minimum, mean and maximum speeds. All of this is done at model initialization, and no further calculations are done when the model advances in time steps.

3. REFERENCES

Bjørnson, F. O. et al. (2016). *D5.1 Model development guidelines*, AQUAEXCEL^{2020} Report.

Endresen, P. C., Føre, M., Fredheim, A., Kristiansen, D., & Enerhaug, B. (2013). Numerical modeling of wake effect on aquaculture nets. In ASME 2013 32nd International Conference on Ocean, Offshore and Arctic Engineering (pp. V003T05A027-V003T05A027). American Society of Mechanical Engineers.

Løland, G. (1993). Current forces on, and water flow through and around, floating fish farms. Aquaculture International, 1(1), 72-89.

Sayma, A. (2009). *Computational fluid dynamics*. Bookboon

Shchepetkin, A. F., & McWilliams, J. C. (2005). The regional oceanic modeling system (ROMS): a split-explicit, free-surface, topography-following-coordinate oceanic model. *Ocean modelling*, *9*(4), 347-404.

Slagstad, D., & McClimans, T. A. (2005). Modeling the ecosystem dynamics of the Barents Sea including the marginal ice zone: I. Physical and chemical oceanography. Journal of Marine Systems, 58(1-2), 1-18.

Glossary

AQUAEXCEL^{2020}: AQUAculture Infrastructures for EXCELlence in European Fish Research towards 2020

CFD: Computational Fluid Dynamics

FMI: Functional Mock-up Interface. A standard to support model exchange and co-simulation of dynamic models using a combination of xml-files and compiled C-code (http://fmi-standard.org/).

FMU: Functional Mock-up Unit. A model packaged to support the FMI standard. The flow field model described in this report is packaged as an FMU.

NetCDF: A set of software libraries and self-describing , platform independent data formats for array-oriented data

Definitions

Document information

EU Project N°	652831	**Acronym**	AQUAEXCEL^{2020}
Full Title	AQUAculture Infrastructures for EXCELlence in European Fish Research towards 2020		
Project website	www.aquaexcel.eu		

Deliverable	**N°**	D5.4	**Title**	First prototype flow field model established
Work Package	**N°**	5	**Title**	JRA1 – Virtual laboratories and modelling tools for designing experiments in aquaculture research facilities

Date of delivery	**Contractual**	31/03/2018 (Month 30)	**Actual**	19/03/2018 (Month 30)
Dissemination level	PU	**PU Public, fully open, e.g. web**		
		CO Confidential, restricted under conditions set out in Model Grant Agreement		
		CI Classified, information as referred to in Commission Decision 2001/844/EC.		

Authors (Partner)	NTNU			
Responsible Author	**Name**	Morten Omholt Alver	**Email**	alver@itk.ntnu.no

Version log

Issue Date	**Revision N°**	**Author**	**Change**
01/03/2018	1	Morten Omholt Alver	First version for internal review
19/03/2018	2	Morten Omholt Alver	Final version

Annex 1: Check list

Deliverable Check list (to be checked by the "Deliverable leader")

	Check list		**Comments**
BEFORE	I have checked the due date and have planned completion in due time	X	*Please inform Management Team of any foreseen delays*
	The title corresponds to the title in the DOW	X	*If not please inform the Management Team with justification*
	The dissemination level corresponds to that indicated in the DOW	X	
	The contributors (authors) correspond to those indicated in the DOW	X	
	The Table of Contents has been validated with the Activity Leader	X	*Please validate the Table of Content with your Activity Leader before drafting the deliverable*
	I am using the AQUAEXCEL^{2020} deliverable template (title page, styles etc)	X	*Available in "Useful Documents" on the collaborative workspace*
The draft is ready			
AFTER	I have written a good summary at the beginning of the Deliverable		*A 1-2 pages maximum summary is mandatory (not formal but really informative on the content of the Deliverable)*
	The deliverable has been reviewed by all contributors (authors)		*Make sure all contributors have reviewed and approved the final version of the deliverable. You should leave sufficient time for this validation.*
	I have done a spell check and had the English verified		
	I have sent the final version to the WP Leader, to the 2nd Reviewer and to the Project coordinator (cc to the project manager) for approval		*Send the final draft to your WPLeader, the 2nd Reviewer and the coordinator with cc to the project manager on the 1st day of the due month and leave 2 weeks for feedback. Inform the reviewers of the changes (if any) you have made to address their comments. Once validated by the 2 reviewers and the coordinator, send the final version to the Project Manager who will then submit it to the EC.*