This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No 727689

Document Title:

D2.3 – Environmental Conditions Database

Document No:

RLT-WP2-3-PDL-001-03

Status Code	Description
A	Accepted
B	Issued for Acceptance
C	Issued for Review
D	Information Only Approval not required
E	Cancelled

03	B	Issued for Acceptance	Peter McCallum 30/09/21	Brian Sellar 30/09/21	
02	B	Issued for Acceptance	Chris Old, Peter McCallum, Brian Sellar 28-29/09/21	Brian Sellar 29/09/21	
01	C	Draft Completion	Chris Old, Peter McCallum 20/09/2021	Brian Sellar 21/09/21	Stéphane Paboeuf
00	D	Document Creation	Chris Old 19/03/2019		
Rev No	Status	Revision Description	Prepared By / Date	Reviewed By / Date	Approved By / Date

European Commission
H2020 Programme for Research & Innovation

Advanced monitoring, simulation and control of tidal devices in unsteady, highly turbulent realistic tide environments

Grant Agreement number: 727689

Project Acronym: RealTide

Project Title: Advanced monitoring, simulation and control of tidal devices in unsteady, highly turbulent realistic tide environments

Deliverable 2.3
Environmental Conditions Database: Collation, Demonstration and Dissemination

WP 2
Realistic Tidal Environment

WP Leader: The University of Edinburgh

Dissemination level: Public

Summary: This reports forms Deliverable 2.3 and details the work of Task 2.3 of RealTide. It provides a description of the development and construction of a database for serving a range of different classes of data relevant the MRE sector in an integrated format. Examples of how to query the construct as both a data library and as a means to sub-setting the underlying data based on key state parameters that represent environmental conditions and machine states. A description of a prototype front-end service is provided explaining the rationale behind the separation of the interface service, database service and a dedicated data processing service. Feedback on preliminary versions of the database is presented. Datasets generated in RealTide Work Package 2 (in-situ data and hydrodynamic model data) and Work Package 3 (tank-test data from scale model turbine testing) have been identified and integrated in the demonstrator database.

Objective:
To develop and make publicly available a dedicated database the contains data primarily comprising

i. Re-analysed publicly available metocean data
ii. Existing UEDIN environmental datasets
iii. Environmental data [and additional related datasets] generated during RealTide that has been cleared for external release.

Table of Contents

List of Figures

List of Tables

Abbreviations & Definitions

BV	Bureau Veritas
BV M&O	Bureau Veritas Marine & Offshore
HO	HydrOcean
UEDIN	The University of Edinburgh
EO	EnerOcean
SAB	Sabella
1-T	1-Tech
IFR	Ifremer (Institut Français pour la Recherche et l'Exploitation de la Mer
ISSA	Ingeteam Power Technology
GA	Grant Agreement
PMP	Project Management Plan

3D	3 Dimensional
ADCP /ADP	Acoustic Doppler Current Profiler / Acoustic Doppler Profiler
ADV	Acoustic Doppler Velocimeter
BEMT	Blade Element Momentum Theory
CADP	Convergent-Acoustic Doppler Profiler
CFD	Computational Fluid Dynamics
DC	Data Controller
DOF	Degrees of Freedom
ED	Edinburgh Designs
EDC	Energy Data Centre
EIDF	Edinburgh International Data Facility
EPCC	Edinburgh Parallel Computing Centre
EULA	End User License Agreement
EMEC	European Marine Energy Centre
FloWave	FloWave ocean energy research facility, The University of Edinburgh.
FloWTurb	Flows, Waves and Turbulence
GA	Grant Agreement
HATT	Horizontal Axis Tidal Turbine
Lat	Latitude
Lon	Longitude
MRE	Marine Renewable Energy
ORE	Offshore Renewable Energy
SBD	Single-Beam Doppler Profiler
TEC	Tidal Energy Converter
UKERC	UK Energy Research Centre
WP	Work Package

References

[1] RealTide, "Grant Agreement ID: 727689 – RealTide Consortium & European Commission – December, 2017."

[2] RealTide, "Consortium Agreement (Rev.3, FINAL) – RealTide Consortium – November, 2017."

[3] RealTide, 2018, "Document Numbering Procedure (RLT-WP6-1-PRO-000-00)."

[4] RealTide, 2018, "Technical Report (Internal): Increased Reliability of Tidal Rotors (RLT-WP1-5-PDL-000-01)."

[5] RealTide, 2019, "Technical Report: Deployment and Instrument Specification for Advanced Flow Characterisation (RLT-WP2-1-PDL-000-03)."

[6] ReDAPT, 2016, "Technical Report MD3.8: Tidal Energy Site Characterisation at The Fall Of Warness, EMEC, UK."

[7] ReDAPT, 2016, "An Introduction to the ReDAPT Tidal Project Environmental Data Set, v2.0."

[8] Sellar, B. G., Sutherland, D. R. J., Ingram, D. M., and Venugopal, V., 2017, "Measuring Waves and Currents at the European Marine Energy Centre Tidal Energy Test Site: Campaign Specification, Measurement Methodologies and Data Exploitation," OCEANS 2017 - Aberdeen, doi:10.1109/OCEANSE.2017.8085001.

[9] Sellar, B. G., Wakelam, G., Sutherland, D. R. J., Ingram, D. M., and Venugopal, V., 2018, "Characterisation of Tidal Flows at the European Marine Energy Centre in the Absence of Ocean Waves," Energies, 11(1), pp. 1–23, doi:10.3390/en11010176.

[10] "University of Edinburgh: Interactive Data Archive for ReDAPT Data." https://data.ukedc.rl.ac.uk/browse/edc/renewables/marine/redapt/search-redapt.html.

[11] Wilkinson, M. D., et al., 2016, "The FAIR Guiding Principles for Scientific Data Management and Stewardship," Sci. Data, 3, doi:10.1038/sdata.2016.18. https://www.go-fair.org/fair-principles/.

[12] Integrated Ocean Observing System (IOOS), 2019, "Manual for Real-Time Quality Control of Stream Flow Observations (Qartod) v2.1," (July), doi:10.25923/sqe9-e310. https://repository.oceanbestpractices.org/handle/11329/1004.

[13] Eaton, B., Gregory, J., Drach, B., Taylor, K., Hankin, S., Blower, J., Caron, J., Signell, R., Bentley, P., Rappa, G., Höck, H., Pamment, A., Juckes, M., Raspaud, M., Horne, R., Whiteaker, T., Blodgett, D., Zender, C., and Herlédan, S., NetCDF Climate and Forecast (CF) Metadata Conventions, Version 1.9, 10 September, 2021.

[14] ReDAPT, 2015, "Technical Report MC7.3: Public Domain Report: Final.", http://redapt.eng.ed.ac.uk/library/eti/reports/MC7.3 Operations Final Report.pdf.

[15] Draycott, S., Payne, G. S., Steynor, J., Nambiar, A., Sellar, B., Davey, T., Noble, D. R., and Venugopal, V., 2019, "Environmental & Load Data: 1:15 Scale Tidal Turbine Subject to a Variety of Regular Wave Conditions," Data Br., 23, p. 103732, doi:10.1016/j.dib.2019.103732.

[16] Draycott, S., Nambiar, A., Sellar, B., Davey, T., and Venugopal, V., 2019, "Assessing Extreme Loads on a Tidal Turbine Using Focused Wave Groups in Energetic Currents," Renew. Energy, 135, pp. 1013–1024, doi:10.1016/j.renene.2018.12.075.

[17] Nambiar, A., Draycott, S., Payne, G. S., Sellar, B. G., and Kiprakis, A., 2021, "Influence of Tidal Turbine Control on Performance and Loads," Appl. Ocean Res., 114(July), p. 102806, doi:10.1016/j.apor.2021.102806.

[18] Draycott, S., Steynor, J., Nambiar, A., Sellar, B., and Venugopal, V., 2020, "Tidal Turbine Load Variability in Following and Opposing Irregular Wave Conditions," International Conference on Offshore Mechanics and Arctic Engineering, p. V009T09A004, doi:10.1115/OMAE2020-18701. https://asmedigitalcollection.asme.org/OMAE/proceedings-abstract/OMAE2020/84416/V009T09A004/1093105.

[19] Draycott, S., Steynor, J., Nambiar, A., Sellar, B., and Venugopal, V., 2019, "Experimental Assessment of Tidal Turbine Loading from Irregular Waves over a Tidal Cycle," J. Ocean Eng. Mar. Energy, 5(2), pp. 173–187, doi:10.1007/s40722-019-00136-9.

[20] Noble, D., Draycott, S., Nambiar, A., Sellar, B., Steynor, J., Lennon, M., Davey, T., and Kiprakis, A., 2020, "Flow Data around Three SuperGen UKCMER Tidal Turbines in a Closely Spaced Staggered Array at FloWave.", https://datashare.ed.ac.uk/handle/10283/3564.

[21] Draycott, S., Steynor, J., Nambiar, A., Sellar, B., and Venugopal, V., 2020, "Rotational Sampling of Waves by Tidal Turbine Blades," Renew. Energy, 162, pp. 2197–2209, doi:10.1016/j.renene.2020.10.037.

[22] Noble, D., Draycott, S., Nambiar, A., Sellar, B., Steynor, J., Lennon, M., Davey, T., and Kiprakis, A., 2020, "Turbine Loading and Performance Data for Three Supergen UKCMER Tidal Turbines in a Closely Spaced Staggered Array at FloWave.", https://datashare.ed.ac.uk/handle/10283/3563.

[23] Draycott, S., Payne, G., Steynor, J., Nambiar, A., Sellar, B., and Venugopal, V., 2019, "An Experimental Investigation into Non-Linear Wave Loading on Horizontal Axis Tidal Turbines," J. Fluids Struct., 84, pp. 199–217, doi:10.1016/j.jfluidstructs.2018.11.004.

[24] Gaurier, B., Ordonez-Sanchez, S., Facq, J. V., Germain, G., Johnstone, C., Martinez, R., Salvatore, F., Santic, I., Davey, T., Old, C., and Sellar, B. G., 2020, "MaRINET2 Tidal Energy Round Robin Tests-Performance Comparison of a Horizontal Axis Turbine Subjected to Combined Wave and Current Conditions," J. Mar. Sci. Eng., 8(6), doi:10.3390/JMSE8060463.

[25] Martinez, R., Gaurier, B., Ordonez-Sanchez, S., Facq, J. V., Germain, G., Johnstone, C., Santic, I., Salvatore, F., Davey, T., Old, C., and Sellar, B. G., 2021, "Tidal Energy Round Robin Tests: A Comparison of Flow Measurements and Turbine Loading," J. Mar. Sci. Eng., 9(4) , doi:10.3390/jmse9040425.

[26] RealTide, 2021, "Technical Report: Synthetic Load Spectra and Time Series of Tidal Turbines(RLT-WP3-5-PDL-001-02).", https://realtide.eu/realtide-project-deliverables.

[27] RealTide, 2018, "Technical Report: Inter-Comparison of BEMT, Blade-Resolved CFD, and BEMT-CFD Hybrid Models of Scale Turbines (RLT-WP3-4-PDL-000-01).", https://realtide.eu/realtide-project-deliverables.

[28] "DataSync Cloud Service (University of Edinburgh).", https://www.ed.ac.uk/information-services/computing/desktop-personal/datasync

[29] Sellar, B. G., Harding, S., and Richmond, M., 2015, "High-Resolution Velocimetry in Energetic Tidal Currents Using a Convergent-Beam Acoustic Doppler Profiler," Meas. Sci. Technol., 26, , doi:10.1088/0957-0233/26/8/085801.

[30] Harding, S., Dorward, M., Sellar, B., and Richmond, M., 2021, "Field Validation of an Actuated Convergent-Beam Acoustic Doppler Profiler for High Resolution Flow Mapping," Meas. Sci. Technol., 32(4), doi:10.1088/1361-6501/abd5ef.

[31] Jourdain de Thieulloy, M., Dorward, M., Old, C., Gabl, R., Davey, T., Ingram, D. M., and Sellar, B. G., 2020, "Single-Beam Acoustic Doppler Profiler and Co-Located Acoustic Doppler Velocimeter Flow Velocity Data," Data, 5(3), pp. 1–11, doi:10.3390/data5030061.

[32] Jourdain de Thieulloy, M., Dorward, M., Sellar, B. G., Old, C., Davey, T., Gabl, R., and Ingram, D., 2020, "Experimental Flow Data from Co-Located Single Beam Acoustic Doppler Profiler and Acoustic Doppler Velocimeter in the FloWave Ocean Energy Research Facility.", https://datashare.ed.ac.uk/handle/10283/3654.

[33] Payne, G. S., Stallard, T., and Martinez, R., 2017, "Design and Manufacture of a Bed Supported Tidal Turbine Model for Blade and Shaft Load Measurement in Turbulent Flow and Waves," Renew. Energy, 107, pp. 312–326, doi:10.1016/j.renene.2017.01.068.

EXECUTIVE SUMMARY

Deliverable D2.3 within work-package 2 (WP2) of the RealTide project was designed to ensure that publicly disclosable arising results from measurement and modelling campaigns together with reprocessing of collated pre-existing datasets could be efficiently shared. It is essential to consider the data sharing methodology and end-use cases from project outset, as these not only drive specification of measurement (i.e., capturing user requirements) but also ensure that data providers generate appropriate and robust meta-data as they proceed. D2.3 forms part of an overall Data Management strategy that seeks to speed up uptake and exploitation of arising datasets through improved processes and systems, and specifically via a dedicated database design that is fit-for-purpose. [1–5]

Original Objectives:

To develop and make publicly available a dedicated database the contains data primarily comprising

i. Re-analysed publicly available metocean data
ii. Existing UEDIN environmental datasets
iii. Environmental data [and additional related datasets] generated during RealTide that has been cleared for external release.

The RealTide project proposed that a holistic approach to data specification, measurement and use (e.g., in engineering design tools) be adopted. This approach was implemented and it identified that in addition to in-situ field measurements additional datasets should be considered for inclusion in the final database, since having access to multiple types of data opens up new use-cases and potential to develop tools that can improve tidal energy reliability. Four classes of data have therefore been incorporated into the prototype database system.

- In-situ field measurements
- Tidal site hydrodynamic numerical simulation data
- Physical simulation data
- Numerical simulation data e.g., Computation Fluid Dynamic (CFD) simulations

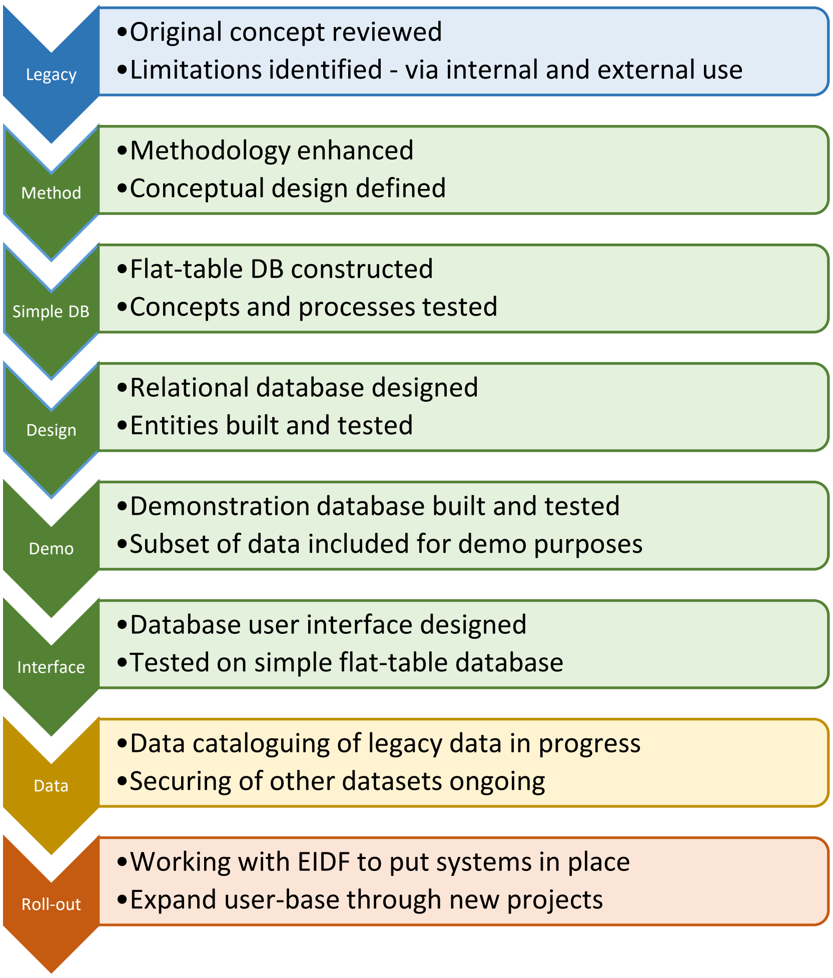
RealTide Database Design and Implementation

A solution has been developed that offers multiple advantages over existing solutions. It uses open-source tools that have a long history of stable operation. Where proprietary systems (e.g., MATLAB) are utilised alternate methods (e.g., python data-handling toolboxes) are under continued development in ongoing related projects. This document reports on a design and implementation that is secure, stable and scale-able in the immediate term and relatively easy to manage and populate with datasets. The work has, however, revealed, that for a long-term, fully scale-able and integrated solution expert dedicated service providers need to be engaged. This process has already started at the University of Edinburgh and is discussed in the report.

Datasets across the four classes have been identified and categorised. Re-processing of legacy and new datasets to match the specification of the implemented database structure is approaching completion. New datasets are still being generated from completed RealTide data generation activities (i.e., model runs and field campaigns) and aligned tidal energy research projects, and these will be processed and added to the database beyond the RealTide programme of work.

Development Process

The development process followed for this work package task is summarised in Figure Exec 1. The starting point for the work was the legacy methodology and data set developed in a previous flagship tidal energy project, the ReDAPT project. From this starting point a phased approach was taken to develop a scalable relational version of the database that supports the addition of new datasets, and allows the inclusion of new data classes that may arise in the future. The colour coding gives a simple indication of the level of completion achieved. Blue represent existing work prior to the start of the RealTide project, green indicates completed phases, yellow indicated incomplete phases of the development, the orange indicates the next phase of work that has been initiated from the work to date. Each phase will be summarised below.



Legacy
- Original concept reviewed
- Limitations identified - via internal and external use

Method
- Methodology enhanced
- Conceptual design defined

Simple DB
- Flat-table DB constructed
- Concepts and processes tested

Design
- Relational database designed
- Entities built and tested

Demo
- Demonstration database built and tested
- Subset of data included for demo purposes

Interface
- Database user interface designed
- Tested on simple flat-table database

Data
- Data cataloguing of legacy data in progress
- Securing of other datasets ongoing

Roll-out
- Working with EIDF to put systems in place
- Expand user-base through new projects

Figure Exec 1: Summary of tasks with level of completion indicated by the colour: green is completed, yellow is ongoing and soon-to-complete, orange is ongoing strategic activities to enable long-term impact.

Methodology

The concept of collating extracted state parameters from field data to enable sub-setting large data sets on a particular system state was demonstrated at the end of the ReDAPT project, in an attempt to make the information content of the extensive data set more accessible. There is real value in this approach to data control, as it provides parameters that non-experts can access directly, while facilitating enhanced searching of a complex data. The original demonstrator was based on a flat-table structure that could not be easily extended by either adding new data records or new instruments without rebuilding the database. The data were also inefficiently stored, as there were more NULL values than data as the structure had to be sparse. The new relational database was designed around the core principle of connecting datasets to extracted information, but allowing the database to scale and have new data added without needing to rebuild the database.

To keep the database small and easily maintained, the full data records are not stored in the database, only a sub-set of parameters and relevant metadata for identifying the data files requested. The searchable information in the database must include references to the actual data files with a means to identify the required data fragments corresponding to a data request. To retrieve the requested data an intermediate process is required that takes the result of query and extracts the data fragments into a standardised out that can be sent to the user making the request. In this way it is possible to secure data protection and manage data sensitivity through firewalls and user permissions.

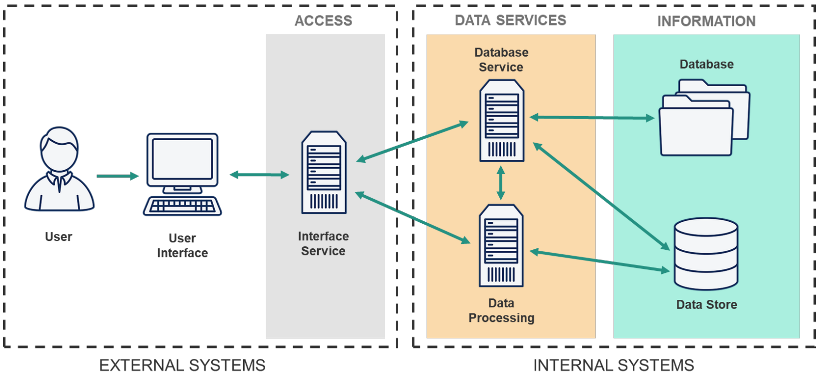


Figure Exec 2: Conceptual structure for the data service components.

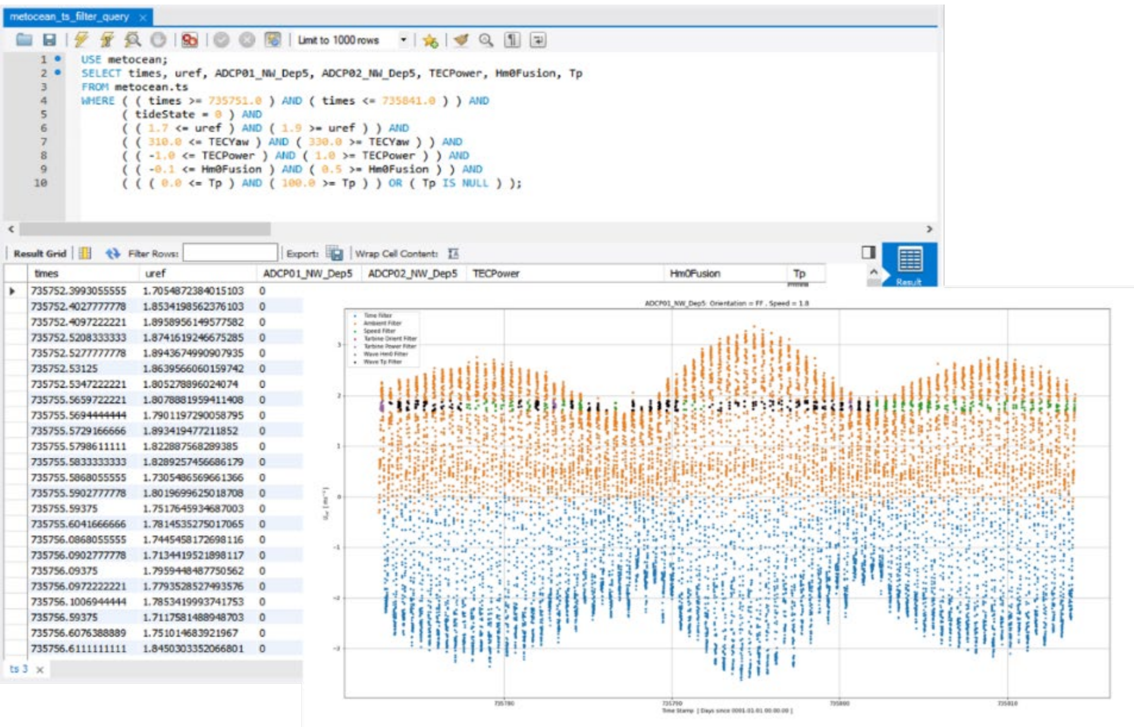
Simple Database

The overall structure was trialled by developing a simple flat-table database in MySQL from an archive of test tank data that were needed for input to numerical simulation work in WP3. This simple database was served internally on the UEDIN Eleanor Cloud computing service through a virtual machine. Users who have University IS accounts can access this service. The interface to the database was initially through the MySQL Workbench app. This allowed queries to be made and the results captured. This information was then used to write a set of MATLAB scripts that took the output from the query and used the information to find the requested data records, extract the relevant information and store the results in standardised NetCDF files. This process was then modified so that the full process was carried out from within MATLAB. Within MATLAB, a connection was made to the database, a query passed and the result capture and passed to the data processing routines. The resulting files were manually check for correctness, then passed to the modellers.

A second database was setup using the flat-table from the ReDAPT construct, and served from Eleanor. This allowed the testing of data sub-setting based on the extracted parameters. A simple python app was built that replicated the process of the original service. Graduate students were given access to these services to test the system, understand the process concepts, and to provide feedback.

Database Design

The approach taken to designing the full relational database was to review the different sources of data that are relevant to MRE developers and researchers, and sort the common data classes based on common features. For class of the key state parameters that are typically used by the sector and that would aid data sub-setting where identified. For each class a set of core entities were defined that support the conceptual design requirements. These entities were then populated with the fields extracted from the metadata relevant to the data class they represent. The entities for the different classes where then modified to include fields that allowed linking information across classes. This process allows the original flat table construct to be recreated by joining the tables through common key fields.



```sql
USE metocean;
SELECT times, uref, ADCP01_NW_Dep5, ADCP02_NW_Dep5, TECPower, Hm0Fusion, Tp
FROM metocean.ts
WHERE ( ( times >= 735751.0 ) AND ( times <= 735841.0 ) ) AND
      ( tideState = 0 ) AND
      ( ( 1.7 <= uref ) AND ( 1.9 >= uref ) ) AND
      ( ( 310.0 <= TECYaw ) AND ( 330.0 >= TECYaw ) ) AND
      ( ( -1.0 <= TECPower ) AND ( 1.0 >= TECPower ) ) AND
      ( ( -0.1 <= Hm0Fusion ) AND ( 0.5 >= Hm0Fusion ) ) AND
      ( ( ( 0.0 <= Tp ) AND ( 100.0 >= Tp ) ) OR ( Tp IS NULL ) );
```

times	uref	ADCP01_NW_Dep5	ADCP02_NW_Dep5	TECPower	Hm0Fusion	Tp
735752.3993055555	1.7054872384015103	0				
735752.4027777778	1.8534198562376103	0				
735752.4097222221	1.8958956149577582	0				
735752.5208333333	1.8741619246675285	0				
735752.5277777778	1.8943674990907935	0				
735752.53125	1.8639566060159742	0				
735752.5347222221	1.805278896024074	0				
735755.5659722221	1.8078881959411408	0				
735755.5694444444	1.7901197290058795	0				
735755.5729166666	1.893419477211852	0				
735755.5798611111	1.822887568289385	0				
735755.5833333333	1.8289257456686179	0				
735755.5868055555	1.7305486569661366	0				
735755.5902777778	1.8019699625018708	0				
735755.59375	1.7517645934687003	0				
735755.6041666666	1.7814535275017065	0				
735756.0868055555	1.7445458172698116	0				
735756.0902777778	1.7134419521898117	0				
735756.09375	1.7959448487750562	0				
735756.0972222221	1.7793528527493576	0				
735756.1006944444	1.7853419993741753	0				
735756.59375	1.7117581488948703	0				
735756.6076388889	1.751014683921967	0				
735756.6111111111	1.8450303352066801	0				

Figure Exec 3: Example of a data query output based on a conditional search of the ReDAPT flat-table database

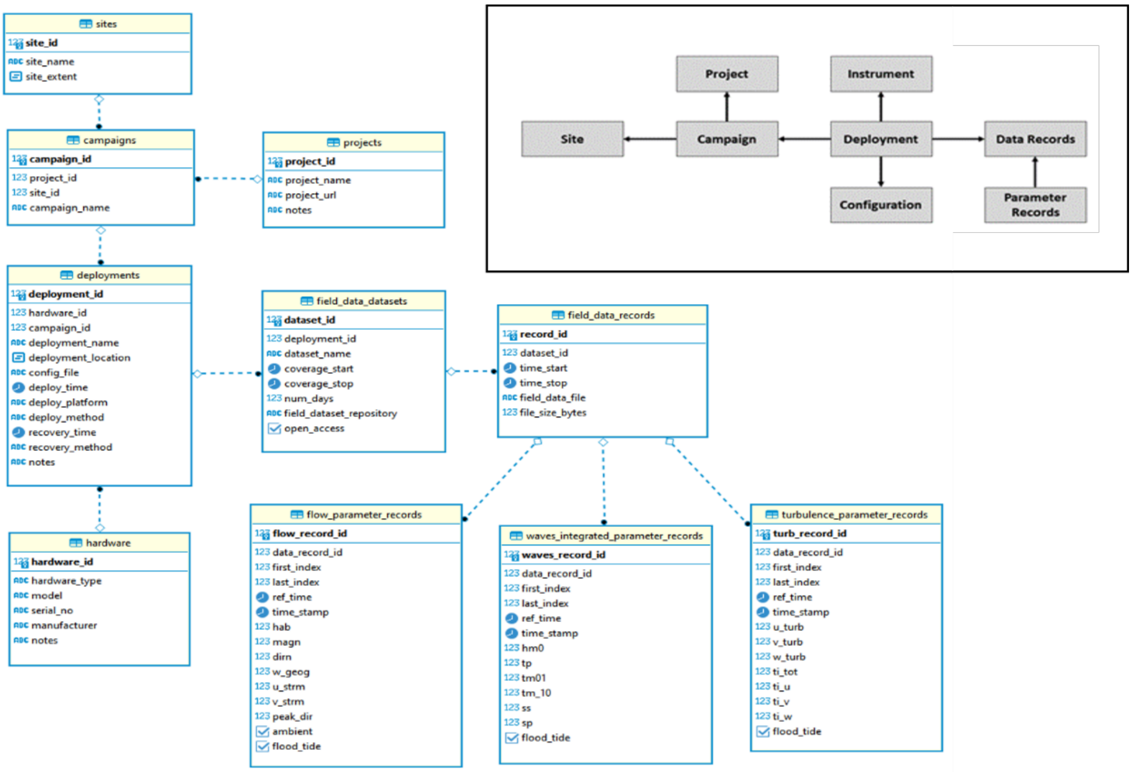
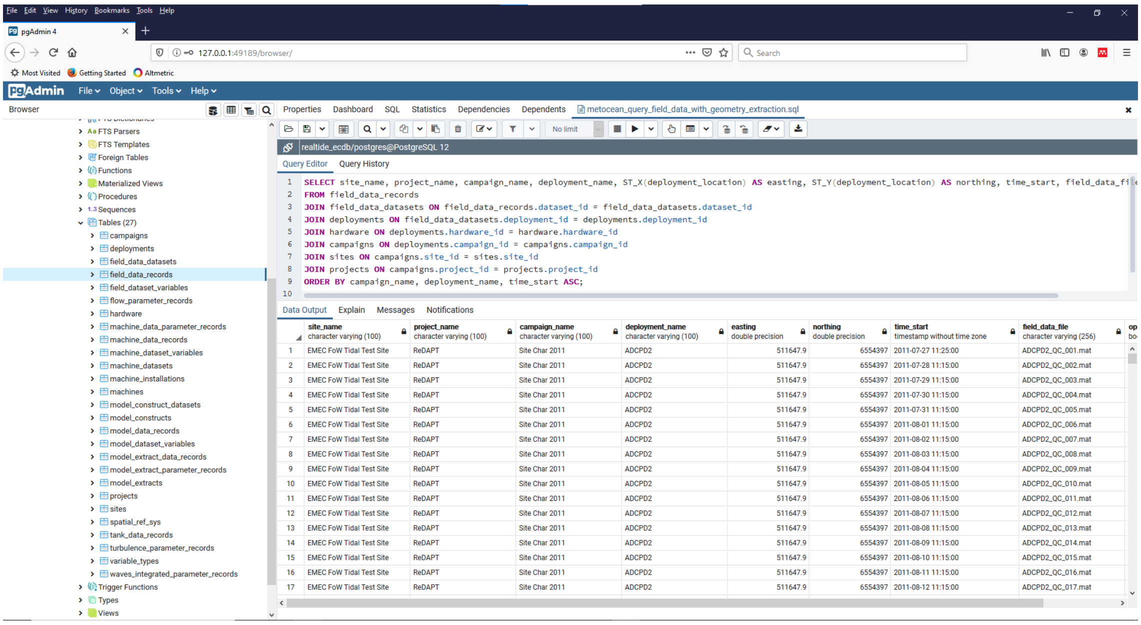


Figure Exec 4: Core entities and the associated entity tables defined for the field data class

Demonstrator Database

The demonstrator database has been constructed and tested on a PostgreSQL server currently running on a local machine, and is in the process of being ported to an interim service on the UEDIN Eleanor Cloud. PostgreSQL was chosen as the server software as there was a requirement to support geo-spatial data formats to future-proof the database and increase accessibility. Both MySQL and PostgreSQL support geo-spatial data types, PostgreSQL was chosen because the UEDIN team has more experience with these tools. The entity tables for the Field, Machine, Model and Tank data have been implemented and populated with demonstration datasets taken from the existing archived and processed data. The relational logic has been tested, and example queries trialled to ensure that the end-use requirements function. The addition of new datasets, data records and parameters sets has been tested and meets the requirements for database extensibility. SQL scripts have been written that can be used to auto-generate the demonstrator database and populate it with the initial data chosen for the demonstrator. The data for entry are stored in a set of ASCII *.csv files. Included in the SQL scripts are a set of example queries that were used to test the database construct.



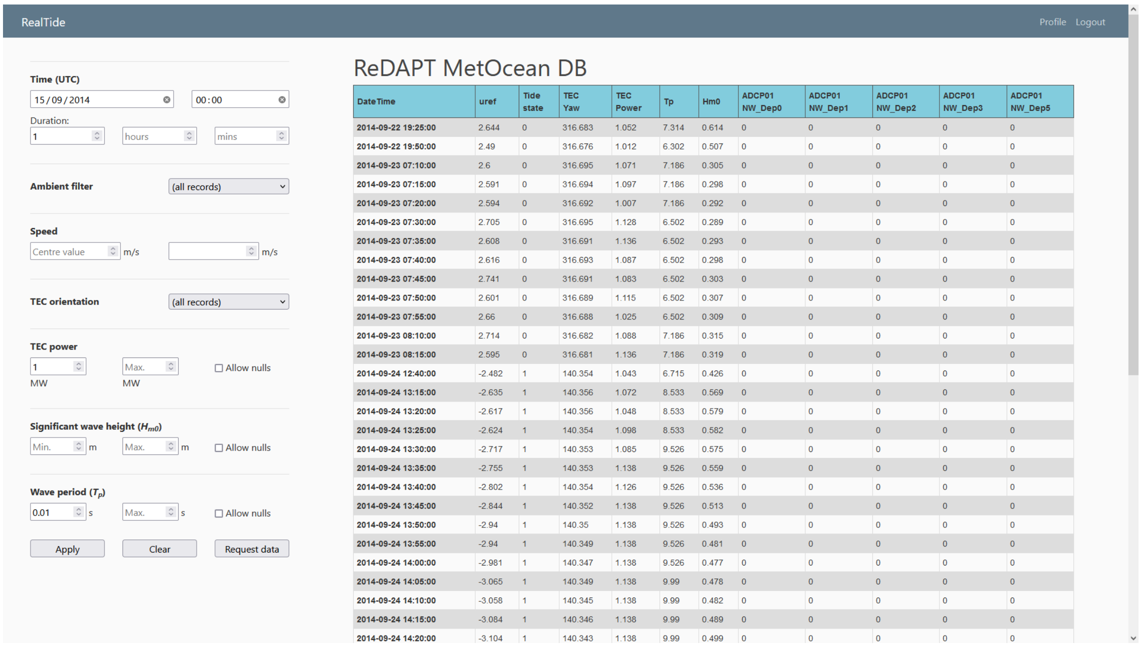
```sql
SELECT site_name, project_name, campaign_name, deployment_name, ST_X(deployment_location) AS easting, ST_Y(deployment_location) AS northing, time_start, field_data_fi
FROM field_data_records
JOIN field_data_datasets ON field_data_records.dataset_id = field_data_datasets.dataset_id
JOIN deployments ON field_data_datasets.deployment_id = deployments.deployment_id
JOIN hardware ON deployments.hardware_id = hardware.hardware_id
JOIN campaigns ON deployments.campaign_id = campaigns.campaign_id
JOIN sites ON campaigns.site_id = sites.site_id
JOIN projects ON campaigns.project_id = projects.project_id
ORDER BY campaign_name, deployment_name, time_start ASC;
```

	site_name character varying (100)	project_name character varying (100)	campaign_name character varying (100)	deployment_name character varying (100)	easting double precision	northing double precision	time_start timestamp without time zone	field_data_file character varying (256)	op bo
1	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-07-27 11:25:00	ADCPD2_QC_001.mat	
2	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-07-28 11:15:00	ADCPD2_QC_002.mat	
3	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-07-29 11:15:00	ADCPD2_QC_003.mat	
4	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-07-30 11:15:00	ADCPD2_QC_004.mat	
5	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-07-31 11:15:00	ADCPD2_QC_005.mat	
6	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-01 11:15:00	ADCPD2_QC_006.mat	
7	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-02 11:15:00	ADCPD2_QC_007.mat	
8	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-03 11:15:00	ADCPD2_QC_008.mat	
9	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-04 11:15:00	ADCPD2_QC_009.mat	
10	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-05 11:15:00	ADCPD2_QC_010.mat	
11	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-06 11:15:00	ADCPD2_QC_011.mat	
12	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-07 11:15:00	ADCPD2_QC_012.mat	
13	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-08 11:15:00	ADCPD2_QC_013.mat	
14	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-09 11:15:00	ADCPD2_QC_014.mat	
15	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-10 11:15:00	ADCPD2_QC_015.mat	
16	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-11 11:15:00	ADCPD2_QC_016.mat	
17	EMEC FoW Tidal Test Site	ReDAPT	Site Char 2011	ADCPD2	511647.9	6554397	2011-08-12 11:15:00	ADCPD2_QC_017.mat	

Figure Exec 5: Testing of the database construct using the PgAdmin interface

User Interface

A prototype user interface has been designed and developed using Django as the interface between the front-end web site and the back-end connection to and querying of an external database server. The system has been tested on the interim flat-table database created for the ReDAPT datasets. The principles directly port to the more complex relational database demonstrator, the main change being more complex SQL queries are required between Django and the database server. The interface manages the User interactions including registration, verification and login controls. Once the User is registered and given permission to login, they can explore the available datasets, browse the data, and submit a data request. For the production service, visibility of datasets with commercial (or other) sensitivities is automatically managed through the back-end, according to the User's specific permissions and affiliations, as defined by the End User License Agreement (EULA) specific to each dataset. The back-end systems manage the data request by recording the User's acceptance of the EULA (all datasets have the permissive MIT License as a minimum), then passing the request to the Data Controller for formal verification. The Data Controller then launches an offline process on a separate secure server which extracts the requested data subset from the full archive. The offline process will initially be a manual operation, but when rolled-out this will be a scripted process with appropriate checks on data restrictions, volume and correctness of data returned. The data package containing the query results are uploaded to temporary data store where the user can login and download the data within a fixed timeframe. Once the upload package is validated by the Data Controller, they should then close the request via the Django back-end, at which point an email is automatically sent to the User informing them on the results of the data request approval process, along with the hyperlinks to the data files. This has been built to provide an interim service until a dedicated host service can be secured.



Time (UTC)
15 / 09 / 2014 — 00 : 00
Duration: 1 — hours — mins

Ambient filter: (all records)

Speed: Centre value m/s — m/s

TEC orientation: (all records)

TEC power: 1 MW — Max. MW — □ Allow nulls

Significant wave height (H_{m0}): Min. m — Max. m — □ Allow nulls

Wave period (T_p): 0.01 s — Max. s — □ Allow nulls

ReDAPT MetOcean DB

DateTime	uref	Tide state	TEC Yaw	TEC Power	Tp	Hm0	ADCP01 NW_Dep0	ADCP01 NW_Dep1	ADCP01 NW_Dep2	ADCP01 NW_Dep3	ADCP01 NW_Dep5
2014-09-22 19:25:00	2.644	0	316.683	1.052	7.314	0.614	0	0	0	0	0
2014-09-22 19:50:00	2.49	0	316.676	1.012	6.302	0.507	0	0	0	0	0
2014-09-23 07:10:00	2.6	0	316.695	1.071	7.186	0.305	0	0	0	0	0
2014-09-23 07:15:00	2.591	0	316.694	1.097	7.186	0.298	0	0	0	0	0
2014-09-23 07:20:00	2.594	0	316.692	1.007	7.186	0.292	0	0	0	0	0
2014-09-23 07:30:00	2.705	0	316.695	1.128	6.502	0.289	0	0	0	0	0
2014-09-23 07:35:00	2.608	0	316.691	1.136	6.502	0.293	0	0	0	0	0
2014-09-23 07:40:00	2.616	0	316.693	1.087	6.502	0.298	0	0	0	0	0
2014-09-23 07:45:00	2.741	0	316.691	1.083	6.502	0.303	0	0	0	0	0
2014-09-23 07:50:00	2.601	0	316.689	1.115	6.502	0.307	0	0	0	0	0
2014-09-23 07:55:00	2.66	0	316.688	1.025	6.502	0.309	0	0	0	0	0
2014-09-23 08:10:00	2.714	0	316.682	1.088	7.186	0.315	0	0	0	0	0
2014-09-23 08:15:00	2.595	0	316.681	1.136	7.186	0.319	0	0	0	0	0
2014-09-24 12:40:00	-2.482	1	140.354	1.043	6.715	0.426	0	0	0	0	0
2014-09-24 13:15:00	-2.635	1	140.356	1.072	8.533	0.569	0	0	0	0	0
2014-09-24 13:20:00	-2.617	1	140.356	1.048	8.533	0.579	0	0	0	0	0
2014-09-24 13:25:00	-2.624	1	140.354	1.098	8.533	0.582	0	0	0	0	0
2014-09-24 13:30:00	-2.717	1	140.353	1.085	9.526	0.575	0	0	0	0	0
2014-09-24 13:35:00	-2.755	1	140.353	1.138	9.526	0.559	0	0	0	0	0
2014-09-24 13:40:00	-2.802	1	140.354	1.126	9.526	0.536	0	0	0	0	0
2014-09-24 13:45:00	-2.844	1	140.352	1.138	9.526	0.513	0	0	0	0	0
2014-09-24 13:50:00	-2.94	1	140.35	1.138	9.526	0.493	0	0	0	0	0
2014-09-24 13:55:00	-2.94	1	140.349	1.138	9.526	0.481	0	0	0	0	0
2014-09-24 14:00:00	-2.981	1	140.347	1.138	9.526	0.477	0	0	0	0	0
2014-09-24 14:05:00	-3.065	1	140.349	1.138	9.99	0.478	0	0	0	0	0
2014-09-24 14:10:00	-3.058	1	140.345	1.138	9.99	0.482	0	0	0	0	0
2014-09-24 14:15:00	-3.084	1	140.346	1.138	9.99	0.489	0	0	0	0	0
2014-09-24 14:20:00	-3.104	1	140.343	1.138	9.99	0.499	0	0	0	0	0

Figure Exec 6: User interface main browser page for public queries of the parameterised data

Data

While that database was being constructed work was on-going in collecting, cataloguing and processing datasets as they became available. The processing methods developed in Task 2.2 of this work package were used to extract parameters from the various data sets. A start has been made on defining the standardized forms for the archived data, and tools are in place to semi-automate the process but this has not yet been implemented. We need to ensure that our final data formats and vocabularies are compatible with the EIDF requirements before converting the large volume of data. There are still datasets to be secured, and the cataloguing and processing of secured is still on-going. A sub-set of data for a range of the data classes have been catalogued, and post-processed for inclusion in the demonstrator database.

Roll-Out

The aim is to use then new Edinburgh International Data Facility (EIDF) as the host. This facility was built to fill an identified gap in the UEDIN data services. EIDF had planned to be operational in 2020, but installation of infrastructure was delayed by the COVID-19 pandemic. The work from this project has been identified as a development case that will be used to scope out requirements and the types of interface capabilities that need to be supported. Discussions with EIDF are on-going, with the aim of starting data transfers to their systems in early 2022.

1 INTRODUCTION

The purpose of this Work Package Task was to bring together data from a variety of sources and integrated them into a database construct that would improve the dissemination of these data to the wider community, and simplify the extraction of targeted information content within the data. This distinction between the data collected, i.e. sensor measurements, and the information content is key to improving data accessibility. Many of the target end-users of the measurements do not have the skills or resources to extract the relevant information from the raw data. Through the development of verified data post-processing methods (carried out in Task 2.2), reliable information can be extracted and served along-side the measurements. This information can in turn be used to sub-set the data based on conditions selected from the available information parameters. This document describes the development of a database construct that meets these aims.

Scope

The document covers the design, construction and initial testing of a relational database designed to integrate a wide range of data relevant to the tidal stream energy sector. This document covers:

- The rationale behind the database structure.
- The relevant data classes and corresponding groups use to capture the data.
- The design of the database entity tables.
- Implementation of the demonstration database construct.
- How to query the database using the relational keys.
- The proposed user interface method demonstrated on a simple flat-table database.
- A discussion of EULA.
- Informal feedback on the database structure and features from internal users.

Background

Restricted data availability and accessibility and difficult extraction of information content are important barriers to engineers, academic researchers, TEC designers and site developers. Data are collected from a wide range of sources, all with unique data formats, and these data generally require expert knowledge to post-process and extract relevant information. An initial attempt to bridge this gap was tested in the closing months of the 5-year UK ReDAPT project (2010-2015) [6,7], where a simple flat-table database construct was built and served by UKERC with the purpose of archiving the large data set generated by the project and to trial a method for extracting data fragments from the archive that correspond to specific environmental conditions and machine states [5–9]. A frozen version of this is available at the UKERC Energy Data Centre via the following web link [10].

To allow the selection of data fragments based on user-defined conditions, the data were post-processed to generate a set of metrics for every 5-mintue time window from the beginning of the initial data collection campaign through to the end of the last data set collected in final data collection campaign. These metrics were collated into a flat table where the values for each metric were stored in a column of the table against the relevant time window. A column for every instrument was included with an index against the 5-minute window time stamp indicting which data fragment from the instrument data records corresponds to the time window. Where there was no metric calculated or where there was no corresponding data from a particular instrument deployment, the table column was set to a null value. This approach creates a large table that is predominantly sparse (i.e. there are a large number of null values throughout the table), which is an inefficient method for storing data. This method is not amenable to adding new records. Similarly, this structure does not allow for modifications to either the underlying data sets or the metrics extracted without a complete rebuild, in essence this is a frozen database structure that is limited to the data state at the time of creation.

The RealTide project will collate and generate multiple complimentary datasets that are relevant to the development of the tidal energy sector, are often interrelated, and have significant differences in their underlying data structures and information content. However, by taking the concept used to serve the ReDAPT data (and building on the lessons-learned from that process and use-case) and extending it into a relational database construct, it is possible to join these disparate data sets in a way that related data fragments can be extracted based on end-user criteria, while still supporting the most basic form of data archiving and full-record recovery. This is the purpose of this work package deliverable.

Conceptual Design for Database Construct

The data to be integrated into the data base cover in situ field measurements, tidal energy converter (TEC) machine data, regional hydrodynamic modelling data, test tank experimental data base on targeted environmental and machine states, and numerical simulations (BEMT, CFD) of turbine response to targeted environmental conditions. These can be identified as different classes of data. These data classes can be separated into two distinct groups: (1) real-world data, and (2) artificial construct data. The real-world data are the *in situ* field measurements, the (TEC) machine data, and regional hydrodynamic model data. These data are link in space and time, *i.e.* they correspond to an actual location on the Earth's surface and a real time period that can be given a date and time stamp. The artificial construct data are the test tank experimental data and the simulation data. These data are linked through the machine construct and the environmental state conditions being modelled. There is no common time stamp or physical location that directly joins the two groups of data together. However, the input environmental conditions for the artificial constructs are informed by real-world data, and the artificial machine constructs are based on real machines. Therefore there is a weak relationship that could be exploited for the purpose of data fragment selection.

Based on the given description of the data classes to be included in the database, separate sets of related tables will need to be generated for each data class to allow cataloguing and archiving and to allow a relational link to the derived search parameters that will be used to interrogate the data. The real-world data will be sampled at different frequencies across the range on sensors used. To allow these data to be searched contemporaneously based on system state parameters, a common reference time will need to be generated as part of the corresponding data class parameter set. To allow collocation all data sets will be tagged with a geo-spatial location. The artificial construct data need to be related via environmental conditions, machine model, and a set of parameters that are typically requested by the end-users of these classes of data.

The full content of the data for any given data class will not be stored directly in the database structure, these will be stored externally in a managed data archive. The data will be linked in the data via a reference to the file name, the archive location, and indices into the file records that match the output from the query request. An interface to the database will be provided that allows the data archive to be queried by a user, the result from a user query will be passed to a post-processing application that will extract the identified data and package it in a standardized data format that will be sent back to the user. The user will be given the option of what level of data is to be returned, e.g. full original records or just the 5-minute extracted parameters.

To provide confidence in data stored the archive, some form of data traceability needs to be adhered to. This is achieved through the inclusion of data quality assurance information, data quality control information, data provenance information, verification of the extraction software used to generate the searchable parameter data, and verification of the database function. For the purpose of this demonstrator system, full QA, QC and verification will not be implemented. The intent is to adhere as closely as possible to the FAIR data principles that are becoming a recognized standard for ensure that data are managed in a way that ensures they can be accessed through ever change data mining methods.

For the purposes of this database construct, not all of the conditions will be met, but this will not nullify the construct build to demonstrate the methods. The actual service will be provide from a platform where many these conditions will be managed by the service, so there is limited value in implementing everything in the demonstrator, but it is worth describing the requirements.

2 DATA MANAGEMENT

The purpose behind the development of this database is to increase the accessibility of open-access data generated through academic research to the end-users within the sector and the wider MRE community. To do this in a way that is both traceable and future proofed, appropriate data management principles should be adhered to. These principles cover every step of the data gathering process from initial data collection through to final archiving and integration into a database or data service. The FAIR Data Principles identify the criteria data should meet to both provide good data provenance and future proofing. The data management steps that need to be considered to meet the FAIR data requirements include quality assurance, cataloguing methods, quality control systems, parameter extraction software verification, and data standardization. This section summarizes these requirements.

FAIR Data Principles

Fundamentally we aim to follow the FAIR data principles. This is to future-proof the data archive and to maintain data traceability. The FAIR data principles [11] are as follows:

- **Findable**
  The first step in (re)using data is to find them. Metadata and data should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of datasets and services, so this is an essential component of the FAIRification process.

- **Accessible**
  Once the user finds the required data, they need to know how they can be accessed, possibly including authentication and authorisation.

- **Interoperable**
  The data usually need to be integrated with other data. In addition, the data need to interoperate with applications or workflows for analysis, storage and processing.

- **Reusable**
  The ultimate goal of FAIR is to optimise the reuse of data. To achieve this, metadata and data should be well-described so that they can be replicated and/or combined in different settings.

Quality Assurance

This is a step of the data collection process that is often missed, but can have a significant impact on the use and re-use of a given dataset. Quality assurance (QA) is the process that should be applied when designing and implementing a data collection exercise. The purpose of QA is to ensure that the data gather meet the intended end-use requirements, and that the data are collected to the best quality possible within the usual constraints of environmental conditions over which there is no control. This includes both the calibration and configuration of the instrument prior to deployment and the design of the deployment method and mitigations against known problems that can occur. The information given in RealTide deliverable D2.1 is aimed at identifying and mitigating the most common issues that will impact data QA. A simple example might be velocity data are being collected that are required to resolve the level of wave and turbulent process at a location. To do this the data need to capture the required time and length scales and cover the relevant sections of the water column. This needs to be factored into the instrument configuration and choice of both mooring method and suitability of the intended measurement location. Getting any of these wrong will undermine the data QA.

This QA information needs to be propagated through with the data, and made available to the end-users before they request data, to allow them to decide whether data in the archive is fit for their intended purpose. For example, if an end-user is interested in studying turbulence, they will want to know the sampling frequency and bin size associated with a given dataset to decide whether the data are suitable for their end-use. If they wish to extract information in Earth coordinate directions (e.g. east, north and vertical) they need to know whether the instrument compass has been calibrated correctly prior to deployment.

The QA metadata need to capture:

- Project name
- Instrument type
- Instrument identifier
- Pre-deployment calibration information
- Deployment site
- Deployment location
- Deployment time stamp
- Mooring method (including height of instrument above the bed)
- Deployment method / Vessel
- Recovery time stamp
- Recovery method / Vessel
- Sensor drift at recovery

This information is required for data quality control, post-processing, and database entity table creation.

Cataloguing

The purpose of cataloguing is to collate all relevant metadata for a given dataset to support the data archiving process and to align with the FAIR data principles. The dataset should be given a unique DOI that allows the set to found by data search engines. This will be linked with the QA metadata to ensure data traceability. The key metadata for every data file in the data set are generated and stored in a searchable table. The data file metadata need to capture:

- The data archive path

- Instrument location
- Instrument configuration parameters
- List of all filenames
- Time stamp for first and last record (temporal coverage) for every file
- The file size in bytes for every file

These data are integrated into the database tables and used provide a list of files available, to calculate total data volume in bytes, to link search results to the files containing the measurements, to estimate data volumes of extracted data subsets, the instrument location data are used to join data sets from different data classes together.

Quality Control

Very rarely are all data of best quality, even when the best possible quality assurance measures have been put in place. Before any data processing is carried out, the data quality needs to be checked and all potentially low quality or bad data need to be flagged. The identified data should not be removed from the original data, instead the quality flags should be used in the post-processing of the data. This approach allows different levels of QC to be used in the post-processing dependant on the end-use of the data. This also allows different gap-filling methods to be applied during post-processing. A version of the QUARTOD QC [12] standards that have been refined for marine instrument data are applied to the data prior to post-processing.

Standardisation

To simplify the process of cataloguing, archiving and data extraction, the underlying datasets should be standardized in both format and vocabulary. There are currently multiple “standards” used for marine data, meaning there is not yet a globally accepted standard. The different standards have been created by data services to meet the internal needs of their systems. There are, however, two common features: (1) the use of the NetCDF data format to store the data, and (2) the use of the NetCDF Climate and Forecast (CF) Metadata Conventions to generate variables names and standardise variable attributes.

NetCDF stands for network common data format (https://www.unidata.ucar.edu/software/netcdf/) [13]. The NetCDF format is self-describing, portable, scalable, appendable, sharable, and archivable. It was developed at UCAR for the management of very large atmospheric data sets. The NetCDF structures are optimised for data management, so has become the most widely used data format. The CF conventions (https://cfconventions.org/) were developed alongside NetCDF, with the goal of managing the output from global ocean-atmosphere-biosphere coupled climate forecast models, which have hundreds of state variables data fields of varying dimensions. The CF data also includes extracted parameters and quality control information. NetCDF has been chosen as the data format that will be used to standardise the data.

The definition of variables names is based on the CF convention, but with guidance from the various marine data standards that have been generated. A draft document has been generated with an over view of the proposed standard forms. This has been trialled for the extraction of tank test data based on user queries, and has been used to store output from numerical simulations carried out in Work Package 3. The development of the standardised format is on-going as data are secured and new requirements are identified.

Parameter Extraction

Essential to the data sub-setting process is a set of parameters derived from the sensor measurements that describe the system state and are commonly used by the community for design and development work. Through discussions with members of the RealTide consortia and other potential end-users of these data, core parameters from the various datasets have been identified. These will be listed in the next sub-section with their associated data source. The software tools to extract these parameters have been developed in Task 2.2 of this Work Package. Parameter extraction is the final step in the post-processing of the data archive that supports the database construct. It should be noted that the database is populated from parent to child, so the parameter data are the last records to be added for a given dataset, therefore it is possible to populate the database in stages, and the parameters can be modified as new methods and variables are identified without have to change any other part of the database.

3 DATA CLASSES

We intend to include data from a range of sources that were identified in D1.5 as of most relevance to the tidal energy sector for turbine design and development, and site selection, characterisation and development. There are five classes of data identified, as follows:

1. **Field Data:** The most fundamental data are the site measurements. These represent the real-world data that drives design and decision making.

2. **Machine Data:** If an operating TEC is in place, then there will instrumentation data, and turbine power production data.

3. **Regional Model Data:** Regional-scale fluid models compliment the *in situ* measurements. These data provide further information on spatial variability, providing more parameters for flow classification and site characterisation.

4. **Test Tank Data:** Next are the physical scale models deployed in test tank facilities. The data collected from the tank experiments are used in the design process, where there is some controls on conditions, and allow potential loads to be estimated.

5. **CFD Simulation Data:** Finally there are the CFD simulations used to investigate machine-fluid interactions in more detail. These data are also used for turbine design and loading estimation.

These five sources of data will be considered as separate data classes for the purposes of designing and building the database. Each data class will have their own set of entities and relational links, but the entity tables will be collected in a single overarching database construct that enables the overall aim of this database, the conditional selection of data subsets based on user-defined system states.

3.1.1 Field Data

Field data are collected from a variety of instruments, often on multiple occasions (i.e. field campaigns) and at multiple locations across a given site. Each instrument will have a specific configuration for a given deployment, and associated with this configuration there will be a collection of data files that contain the measurement records. Specific information can be extracted from these measurement records through advanced post-processing; it is this information content that is relevant to the majority of end-users.

The purpose of the field measurements are to collect range of system state data using a variety of instruments configured to collect data that meet the required technical specifications for the intended end-use. At tidal energy development sites, guidelines on various data-capture methods are provided by the IEC TS 626000:200 document. In general, data covering multiple tidal cycles are required (typically at least 30 days of data), with a sampling frequency high enough to resolve the largest range of time-scale associated with the keys physical processes being parameterized.

There a four main types of measurements that are used for site characterization:

1. Velocity profile time series
2. Surface elevation time series
3. Surface wave measurements
4. Meteorological measurements

The first 3 sets are specific to the fluid environment. These data are collected at the point location where an instrument is deployed. Some spatial variation can be inferred from a spatial distribution of instruments. Fundamentally, the data contains information on all interacting physical processes that can be resolved by the spatial and temporal scales the instrument has been configured for. Advanced data processing techniques are required to decompose the various complex signals into the component processes for the purpose of flow classification and/or site characterization. The meteorological data is auxiliary required to interpret wave-current process related to locally generated wind waves, and can be used to correct for surface pressure effects in marine pressure sensor data.

The key parameter that links all of these data together is time. Therefore, time must be core piece of metadata used to relate the information. Given that instruments will generally have different sampling frequencies, the data need to be sorted into common time fragments that can be used to extract synchronized information and parameters. A similar process will apply to machine specific instrumentation. All data from a given data collection campaign need to be synchronized to allow advanced data querying.

The time fragmentation length needs to be one that contains a sufficient number of samples to generate statistical parameter, while being short enough to allow the assumption of pseudo-stationarity in the dominant physical process, the tidal forcing. From the literature (citation?), for tidal sites 5 minutes is a reasonable upper limit on sample length to assume stationarity. This is the time fragmentation length that will be used throughout to construct the parameter data sets.

3.1.2 Instrument Specific Data

The oceanographic instruments used are as follows:

- Acoustic Doppler Current Profiler (ADCP or ADP) – these measure profiles of the fluid velocity at predefined ranges from the transducer head. These instruments are typically mounted pointing vertically through the water column, so measure the vertical velocity shear profile as a function of time. There are auxiliary sensors that are used to convert the data to Earth coordinates, these pitch and roll sensors, a flux-gate compass, and a pressure sensor.

- Tide gauge or High-frequency Pressure Sensor – these measure surface elevation relative to either a defined datum or the position of the sensor above the sea-bed. These are simple time series of either range or pressure.

- Wave buoys or Acoustic Wave Recorders – these measure 3-D surface motion (i.e. elevation and slopes in the horizontal directions as a function of time at a fixed location. The acoustic wave gauges also measure the wave orbital velocities as a function of time at a number of locations in the water column.

3.1.3 Instrument Specific Parameters

The following table provides a non-exhaustive lists of parameters extracted from the 5-minute time fragments for each instrument provide the basis for both entity definition and metadata collation.

Table 3-1: Instrument specific parameters for 5-minute time fragments.

Instrument	Parameter Description	Parameter(s)
ADCP / ADP / SBD	1-D velocity data in beam polar coordinates	$\bar{u}_{beam}(r,t)$
	1-D velocity data in the Earth coordinate reference frame	$\bar{u}_{beam}(r,\theta,\varphi,t)$
ADCP / ADP	3-D velocity in the Earth coordinate reference frame	$\bar{u}_{geo}(x,y,z,t)$ $\bar{v}_{geo}(x,y,z,t)$ $\bar{w}_{geo}(x,y,z,t)$
	3-D velocity in the dominant stream-wise coordinate reference frame	$\bar{u}_{strm}(x,y,z,t)$ $\bar{v}_{strm}(x,y,z,t)$ $\bar{w}_{strm}(x,y,z,t)$
	3-D velocity fluctuations in the Earth coordinate reference frame	$u'_{geo}(x,y,z,t)$ $v'_{geo}(x,y,z,t)$ $w'_{geo}(x,y,z,t)$
	3-D velocity fluctuations in the dominant stream-wise coordinate frame	$u'_{strm}(x,y,z,t)$ $v'_{strm}(x,y,z,t)$ $w'_{strm}(x,y,z,t)$
	3-D flow acceleration in the Earth coordinate reference frame	$\frac{\partial u_{geo}}{\partial t}(x,y,z,t)$ $\frac{\partial v_{geo}}{\partial t}(x,y,z,t)$
	3-D flow acceleration in the dominant stream-wise coordinate frame	$\frac{\partial u_{strm}}{\partial t}(x,y,z,t)$ $\frac{\partial v_{strm}}{\partial t}(x,y,z,t)$
	3-D Turbulence Intensity in the Earth coordinate reference frame	$TI_{geo}(x,y,z,t)$ $TI_{geo}(x,y,z,t)$ $TI_{geo}(x,y,z,t)$
	3-D Turbulence Intensity in the dominant stream-wise coordinate frame	$TI_{strm}(x,y,z,t)$ $TI_{strm}(x,y,z,t)$ $TI_{strm}(x,y,z,t)$

Tide gauge / Pressure Sensor	Surface Elevation Parameters	$MSL(x, y)$ $h(x, y, t)$
	Wave Parameters	$H_{m0}(x, y, t)$ $T_p(x, y, t)$ $T_{m01}(x, y, t)$
Wave Buoy / Acoustic Wave Recorder	Surface Elevation Parameters	$MSL(x, y)$ $h(x, y, t)$
	Wave Spectra	$S(x, y, f, \theta)$ $S(x, y, f)$ $S(x, y, \theta)$ $S_{swell}(x, y, f)$ $S_{wind}(x, y, f)$
	Wave Parameters	$H_{m0}(x, y, t)$ $T_p(x, y, t)$ $T_{m01}(x, y, t)$ $\theta_p(x, y, t)$

Machine Data

This data class is specific to a physical energy conversion device deployed at a given site. For the purposes of this demonstrator database we will only consider tidal energy converters, but the concepts can be readily extended to include wave energy converters, off-shore-floating wind turbines, *etc*. There are a variety of different types of TEC's in operation, with varying levels of mechanical complexity. This information has been capture in the "Reliability Database" developed in WP1 of the RealTide project. For demonstration purpose we will only consider the DeepGen IV [14] turbine deployed at the Falls of Warness during the ReDAPT project.

3.1.4 Source Specific Data

The machine data will have three main sources:

- Operational data giving the turbines status, *e.g*. operating, locked, under commission, under maintenance, retrieved, yaw direction, blade pitch control, breaking, *etc*.

- Production data giving the power generated as a function of time.

- Machine condition data take from internal sensors, *e.g*. rotor rotation rate, loads, strains, system temperature, pressure, gearbox data, *etc*.

Fundamentally all of these data will carry timestamps, operational data will typically be given for time windows, while all other data will be stored at the sampling frequency of relevant sensors.

The machine data will have common site information to the field data and will have overlapping time periods common to both data classes. Time fragmentation can be applied to the time series data to synchronise the 5-minute average parameters extracted from the data. A flagging method will need to be applied to map the windowed data onto the 5-minute fragments.

3.1.5 Source Specific Parameters

The operational status of a turbine will vary with time, and machines that are able to control their orientation (yaw) and to feather their blades (pitch) to reduce loading will have addition information that may be useful for sub-setting data based on conditions. There will be a record of power produced as well as the potential load from the grid. An operational tidal turbine has a large array of sensors for system monitoring and fatigue analysis. Only a sub-set of these will be considered as there is very little open-access tidal turbine data currently available. The turbine data collected during the ReDAPT project currently has limitations on what can be released publicly, currently only 5-minute averaged data for a small set of sensors can be released.

Table 3-2: Machine source specific parameters for 5-minute time fragments.

Data Source	Parameter Description	Parameter(s)

Machine Status	Orientation
	FF = Forward to Flood
	RF = Reverse to Flood
	FE = Forward to Ebb
	RE = Reverse to Ebb
	OA = Off-Axis
		θ_{YAW}
		$Orient\ \{FF, RF, FE, RE, OA\}$

Power Train	Generating	$\bar{P}$

Regional Model Data

This data class is specific to regional scale hydrodynamic flow modelling and wave modelling. These models are developed for a site where field data have been collect, with the aim of adding information that cannot be capture easily through instrument deployments. The 2-D and 3-D nature of these models mean that data sets are large, so in the first instance only sub-sets of these data will be collected. These sub-set will be chosen so as to complement the field data.

A model will be designed to provide detailed information for a particular site and for a specified time frame. To link these data to the in situ data, a common deployment identifier could be used. This will allow a similar entity structures to be developed for both model and in situ data. For the model, however, information about the model domain construction, open-boundary forcing, boundary friction data, numerical solver used, and model specifics such as turbulence closure scheme, time step, *etc.*, will need to be included.

There will be both a spatial and temporal commonality between the numerical model data and the site measurements. Therefore a similar time fragmentation needs to be applied to synchronise the information extracted from the model data and the field data. Given the volumes of data generated by regional models, the simplest approach is to only save the model output at 5 minute intervals. This will not be exactly the same as the 5-minute averaged in situ data, as the model values will be instantaneous values. To further limit the data content data from point locations corresponding to the key field data locations will be extracted as the core input data sets.

3.1.6 Model Specific Data

The data available based on model type are as follows:

1. Hydrodynamic Flow Model – these typically generate both 2-D and 3-D data fields at the model domain node locations. These data will be interpolated onto the required site locations for inclusion in the database. From the model field data a variety of other

parameters can be derived that depend on spatial variability. Spatial maps of integrated and statistical parameters can be generated for use in site characterization.

2. Wave Model – these generate wave spectra at the model node locations. From these wave spectra a range of integrated wave parameters and wave statistics can be derived. Maps of wave parameters and wave statistics can be generated for use in site characterization.

3.1.7 Model Specific Parameters

The following table gives a non-exhaustive lists of the parameters extracted from the 5-minute model outputs for two types of regional model considered. These provide the basis for both entity definition and metadata collation.

Table 3-3: Regional model specific parameters for 5-minute output data.

Model Type	Parameter Description	Parameter(s)
Hydrodynamic Flow Model	3-D velocity in the Earth coordinate reference frame	$\bar{u}_{geo}(x,y,z,t)$ $\bar{v}_{geo}(x,y,z,t)$ $\bar{w}_{geo}(x,y,z,t)$
	3-D velocity in the dominant stream-wise coordinate reference frame	$\bar{u}_{strm}(x,y,z,t)$ $\bar{v}_{strm}(x,y,z,t)$ $\bar{w}_{strm}(x,y,z,t)$
	3-D velocity fluctuations in the Earth coordinate reference frame	${u'}_{geo}(x,y,z,t)$ ${v'}_{geo}(x,y,z,t)$ ${w'}_{geo}(x,y,z,t)$
	3-D velocity fluctuations in the dominant stream-wise coordinate frame	${u'}_{strm}(x,y,z,t)$ ${v'}_{strm}(x,y,z,t)$ ${w'}_{strm}(x,y,z,t)$
	2-D spatial fields	$MSL(x,y)$ $h(x,y,t)$ $U(x,y,t)$ $V(x,y,t)$ $\frac{\partial h}{\partial x}(x,y,t)$ $\frac{\partial h}{\partial y}(x,y,t)$ $\frac{\partial U}{\partial x}(x,y,t)$ $\frac{\partial U}{\partial y}(x,y,t)$ $\frac{\partial V}{\partial x}(x,y,t)$ $\frac{\partial V}{\partial y}(x,y,t)$ $vort(x,y,t)$ $circ(x,y,t)$

Wave Model	Wave Spectra	$S(x,y,f,\theta)$ $S(x,y,f)$ $S(x,y,\theta)$ $S_{swell}(x,y,f)$ $S_{wind}(x,y,f)$
	Wave Parameters	$H_{m0}(x,y,t)$ $T_p(x,y,t)$ $T_{m01}(x,y,t)$

Test Tank Data

Test tank experiments target machine-fluid interaction through the use of scaled physical models of a machine deployed in a fluid tank that allows the control of the flow speed and direction and can simulate scaled waves that represent real sea states. The standard method for measure the tank fluid state is with acoustic Doppler velocimeters (ADVs) which collect point measurements of the 3-D velocity vector with time, and capacitive wave gauges which measure the surface elevation at point locations with time. Some tank facilities access to laser Doppler velocimeters (LDV's) which provide a vertical profile of the along-stream flow. Larger tanks may also deploy acoustic Doppler profilers to retrieve vertical profiles of the 3-D velocity vector with time. The machine being tested will have its own integrated array of sensors. Initially only data related to scale models of tidal turbines will be included, but the methodology can be extended to other types of machines.

There will be a set of parameters that define the operating state of the tank. In general an experiment will request a target state that the tank needs to meet. This will be a flow speed and direction and/or a predefined wave state, e.g. wave height, period and direction or wave spectral parameters for random sea states. Similarly the model TEC will have target control values that the experiment aims to meet. It is these target parameters that provide the conditional search metrics for this class of data. There will also be derived parameters from the instrumentation that will link a given experiment back to the relevant data records. Data from legacy projects, where the data can be shared publicly, has been reviewed and collated. These include testing of 1:15 scale tidal turbines [15] in single and array layouts under varying input conditions (combinations of waves and currents) and varying machine operating points [16–25]. Data from completed tests during RealTide where the data is not commercially sensitive will be shared [26].

3.1.8 Test Tank Parameters

The following table gives a non-exhaustive list of the test tank parameters against their source. These provide the basis for that data class entity definitions and metadata collation.

Table 3-4: Test tank specific parameter set

Data Source	Parameter Description	Parameter(s)
Tank Target State	Mean Flow (speed and direction)	$\bar{u}$ $\bar{\theta}$
	Regular Wave State	H T θ
	Irregular Wave State	H_{m0} T_p γ θ

Machine Target State	Rotation	TSR RPM
ADV	Flow Parameters	$\bar{u}(x,y,z)$ $\bar{\theta}(x,y,z)$ $TI(x,y,z)$
Wave Gauge	Wave Parameters	$MSL(x,y)$ $H_{m0}(x,y)$ $T_p(x,y)$
Machine Sensor	Strain Gauges	$RBM_i(x,y)$
	Controller	$\bar{P}$

CFD Simulation Data

The method for integration of CFD data is still under development, with on-going discussions with modellers to determine how best to describe the models and machine implementations in some form of standardised manner. There has been limited feedback on what end-user might require from the BEMT/CFD simulations, so further engagement is required with potential end-users of this data class. At the most basic level these, an archive of key numerical simulations (e.g., identified benchmarks) can be constructed and the appropriate searchable parameter tables can be added later once the requirements are defined. Further information on related simulation work carried out under RealTide can be found in [27].

4 DATABASE SPECIFICATION

The overall aim of the database is to simplify access to the information content in complex data sets, i.e. this goes beyond a simple data catalogue where a library of files can be searched. To achieve this the raw data need to be post-processed and key information extracted in the form of parameters and metrics that can be used to do refined searches based on user-define environmental conditions and/or machine status. The database must still provide the capability to search and extract the underlying library of data files. The database structure will require a cascade of related tables that collects metadata describing the files and their content, while allowing new data records and data sources to be added without the requirement to rebuild the database.

There is an underlying commonality in data structures for the field, machine and regional model data, a commonality in data structure for test tank and CFD simulation data. The key difference between these two groups is that the field, machine and regional model data correspond to specific real-world locations and times, whereas the test tank and CFD simulations are artificial constructs the represent fixed system states for process modelling. The test tank and CFD simulation data are typically short time series generated for stationary state conditions (i.e. constant in-flow speed, constant mean profile structure, fixed wave conditions, *etc*.). A picture of system responses is developed through a series of experiments where some or all of these stationary state conditions are varied. The commonalities between the data types within these two groups and the fundamental system state differences between the two groups, suggests that between these two data groups there will fundamental differences in the way the database entities are designed and linked.

Despite the system differences between the real-world data and the artificial constructs, there is a need to be able to link these data together in a searchable way. In principle the artificial constructs are designed to represent conditions that may be encountered in the real-world. When designing and developing turbines, physical scale models can be constructed and the performance and response tested under a range of conditions that are representative of those a tidal energy extraction site. This is also the purpose of BEMT and CFD numerical simulations. Both the physical scale model and numerical simulations have to be validated against real-world instances to provide confidence in the results of the simulated systems. To support this validation process, the real-world data and artificial construct data need to be searchable on common state variables, *e.g.* TEC in-flow speed, wave conditions (H_{m0}, T_p, ϑ_p, wave direction relative to in-flow, *etc.*), and, if possible, for the machine being represented by the artificial constructs. This needs to be kept in mind when designing the database entities.

Database Structure

The database is built around a set of core entities that allow records be added to the database without breaking relational linkages between entities, while minimizing the amount of information stored in the database and eliminating, or at least limiting, replication of information across entities. The relational links between these core entities defines the database structure. Definition of the entity table fields allows information needed to support data extraction to entered and the relational links defined explicitly. This process will be applied to each identified data class with a view to supporting the interrelationship of different data classes. The hierarchical data structures need to be kept as simple as possible to help simplify the SQL query statements.

4.1.1 Real-World Data

4.1.1.1 Field Data

Field data correspond to a real-world location, hence the site being studied is a core entity. Associated with a site there will be one or more data collection campaigns for one or more projects. Within a data collection campaign there will be one or more deployments of one or more instruments. Each instrument will be configured in a specific way for a given deployment, and associated with each instrument deployment there will be a set of data files containing the measurements collected. Post-processing of these measurements generates the 5-minute parameter data sets that will be used to search the data archive for instances that correspond to specific environmental conditions. This breakdown effectively identifies the core entities required to build the database structures for the field data. This is shown diagrammatically in **Error! Reference source not found.**.

The arrows in **Error! Reference source not found.** show the linkage to parent entity tables, i.e. a campaign is a child of a site, a deployment is a child of a campaign, an instrument and a configuration, data records are a child of a deployment, and parameter records are a child of the data records. For a given deployment we can search for the data records, the instrument and its configuration and campaign. Through the campaign we can get the site, and parameter records can be linked to a deployment through the parent data records.

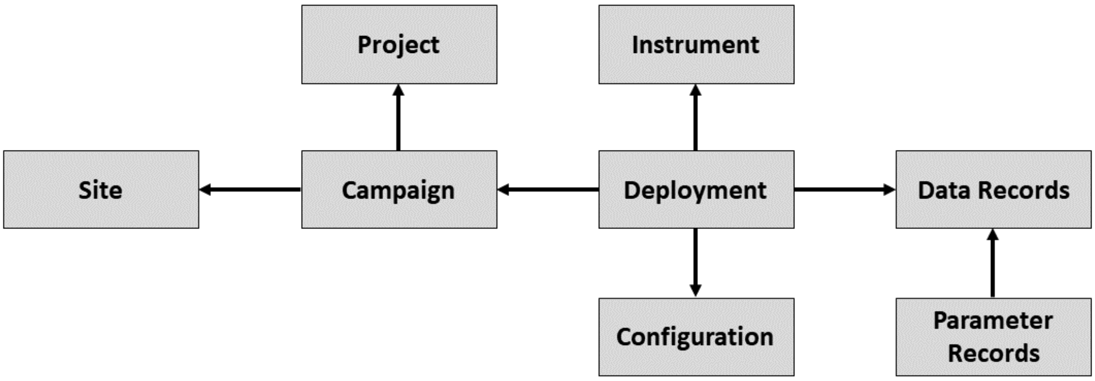


Figure 4-1: Base entities for field data database structure.

4.1.1.2 Machine Data

Machine data are related to the Tidal Energy Converter (TEC), these include machine behaviour, power production, and auxiliary sensor that measure the environment. The site entity is common to both the machine data and the field data. However, the machine data do not have an associated campaign, and the deployment is replaced by Installation/Recovery operations. While installed there may be maintenance periods where the turbine is non-operational, and failures may also lead to operations stopping. Therefore, a TEC will be directly linked to an installation, and an installation to a site. There will be data that relate directly to the TEC (e.g. operational status, orientation, power production, etc.), and there will be data collected from sensor within the TEC (e.g. rotation rate, loads, temperature, pressure, etc.). There are likely to be separate files for the various sensor records, but these will all be linked directly to the particular turbine with appropriate time stamping. Various state parameters can be derived from these data records.

For the machine data, an installation is linked to a specific site and a specific TEC. The TEC will be linked to a set of sensor from which data are collected, and there will be a status record linked to each turbine. Associated with an installation there will be a set of data records, from which relevant parameter records are derived. Figure shows the entity relationships for the machine data.

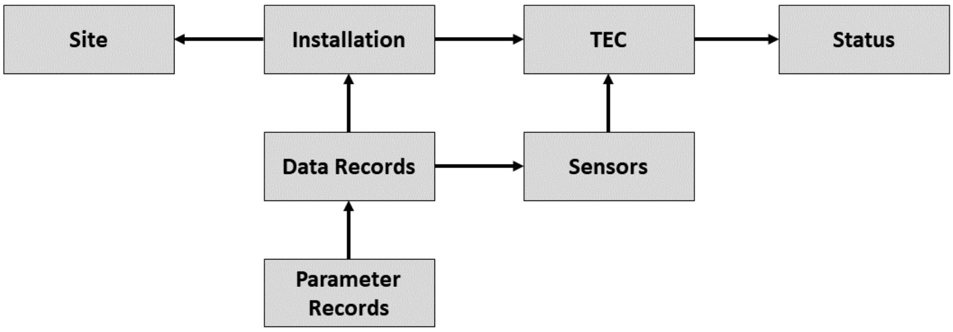


Figure 4-2: Base entities for machine data database structure.

4.1.1.3 Regional Model Data

Regional model data are more complex due to their 4-D nature (i.e. 3 space dimensions and time). It is out with the scope of this work to fully integrate large model data sets into the database construct. Instead time series of data extracted from the model at locations relevant to the end-use of the database will be the initial consideration. In essence this reduces the data to only that for specific locations and time periods. The site entity remains common, the deployment is replaced by a model run, and the model run is linked to a particular solver and model configuration. For a given model run there will be a set of output data, and from these data parameter records can be derived.

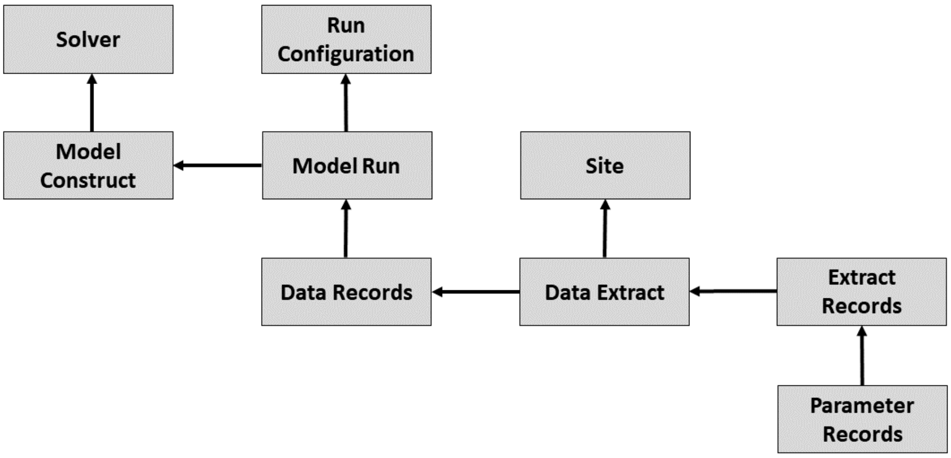


Figure 4-3: Base entities for regional model data database structure.

4.1.2 Artificial Construct Data

4.1.2.1 Test Tank Data

Test tank experiments fall into two categories: (1) environment state only, and (2) environment plus physical model. The first category is used to characterise the tank behaviour and to get measurements of the undisturbed flow prior to installing a machine model. The second category is used to collect data on machine response to environmental conditions. The environment state for a given test tank experiment is defined in terms of target values, e.g. a target flow speed and a target wave. There inherent limitations on how accurately the flow speed and wave conditions can be recreated in any given experiment. A given tank will have an inherent turbulence level, which can be modified by placing structures up-stream of the test area. The instrument measurement data are used to define the actual environment state achieved. For the purposes of an end-user request, the target conditions for the basis of the search. Therefore, the entity breakdown starts with the test tank facility in which experiments related to a particular project are carried out. A given experiment will required a definition of the target environmental system state, where a range of wave specifications can be selected. An experiment will always deploy measurement instrument, but may or may not include a machine, therefore separate tables are required to define the instruments and machines used in a given experiment. Data records are collected from both the instruments and the machine model sensors and a set of searchable parameters can be derived from these data. The entity structure and relationships are shown in Figure .

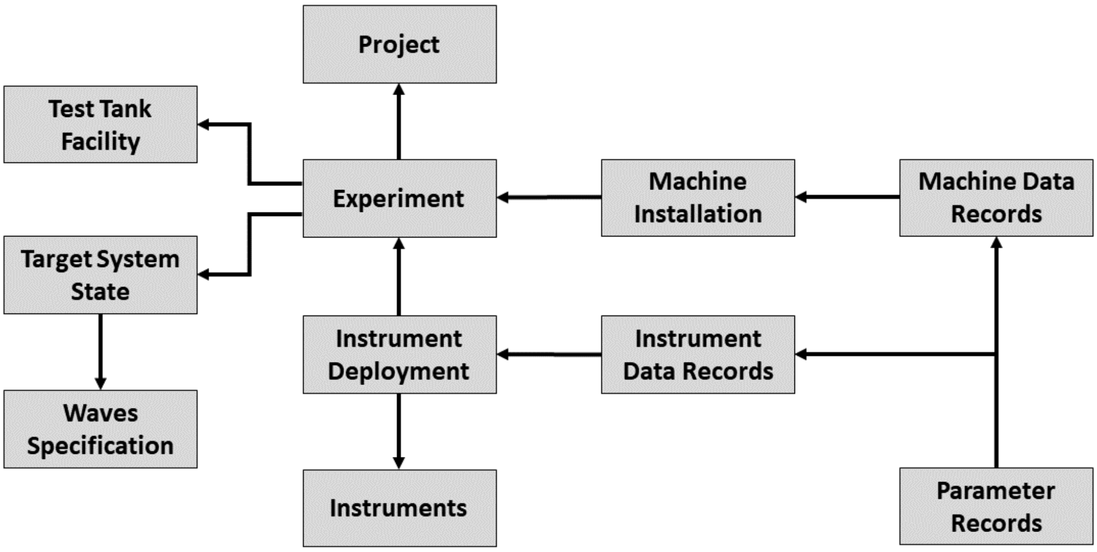


Figure 4-4: Base entities for test tank data database structure.

4.1.2.2 BEMT & CFD Simulation Data

The purpose of BEMT and CFD simulations are to model fluid machine interactions, so these simulations inherently required a method to define the fluid state and some method for representing the machine or machine component that is being simulated. These simulations can be for full-scale replication, of recreations of the physical scaled models used in test tank experiments. The inflow state can be samples in the case of CFD simulations or parameterised in the case of BEMT models. The machine of component models will include measure of their response to the fluid forces. These response parameters typically capture the same parameters that are measured for full scale systems or scaled physical models. They may also include other parameters that are not easily measured in real systems. Fundamentally a type of solver will be chosen for the numerical simulation work, this represents the core parent entity. A model construct is defined for a given simulation, and this construct may be based on a real full-scale machine or a scaled physical model used in test tank experiments. Associated with a model construct there will be one or more runs with a specific configuration used for each run. Each model run will generate a set of data records, and from these data records the searchable parameters are derived. Figure show the core entities require to capture the numerical simulation data.

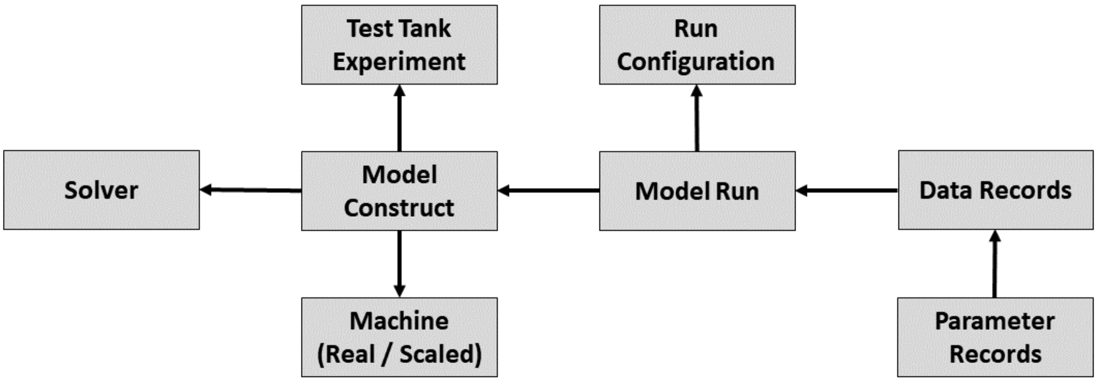


Figure 4-5: Base entities for BEMT and CFD simulation data database structure.

Database Entity Tables

The base entities are used to build the database tables. Each base entity will have a unique identifier for every record entered. The arrows in the database structure diagrams show where foreign key relational links need to be created to allow the joining of the related tables. Within each table there will be a set of fields that required to define the key information content for end-use searches and data description. For the purpose of the demonstrator database, a minimum set of tables and associated fields will be generated that show how a suitable construct can be formed that provides the flexibility required to manage, extend, and search complex data sets. Initially only the field, machine and model data classes have been implemented. These extend the original construct based purely on the ReDAPT data, and are sufficient to show how the database can be used and new data added.

4.1.3 Field Data Class

The minimum set of entity fields required to define the field data class are summarised in the entity relationship diagram shown in **Error! Reference source not found.**. In section 3.1.3 the range of possible parameters that could be derived from the various instruments covered a range of different physical processes. Some instruments record measurements from multiple sensors, so parameters from a range of processes could be derived from the data from a single instrument, but not all instruments of the same type will have the same set of sensors, e.g. ADPC's fundamentally measure water flow velocity profiles, many of these instruments will include a pressure sensor which can be used to extract both tidal elevation data and, if sensitive enough, integrated wave parameters. If an ADCP is deployed without a pressure sensor, or if the sensor is faulty then associated data records cannot be used to extract elevation and wave data. For this reason separate parameter tables associated with specific physical processes have been constructed. This allows data related to a specific process to be gathered from a variety of different instruments into a single table to simplify the data querying process.

In the demonstrator database construct, tables have been created for flow, wave and turbulence process parameters. A sub-set of the full range of parameters for each process have been implemented for testing and demonstration.

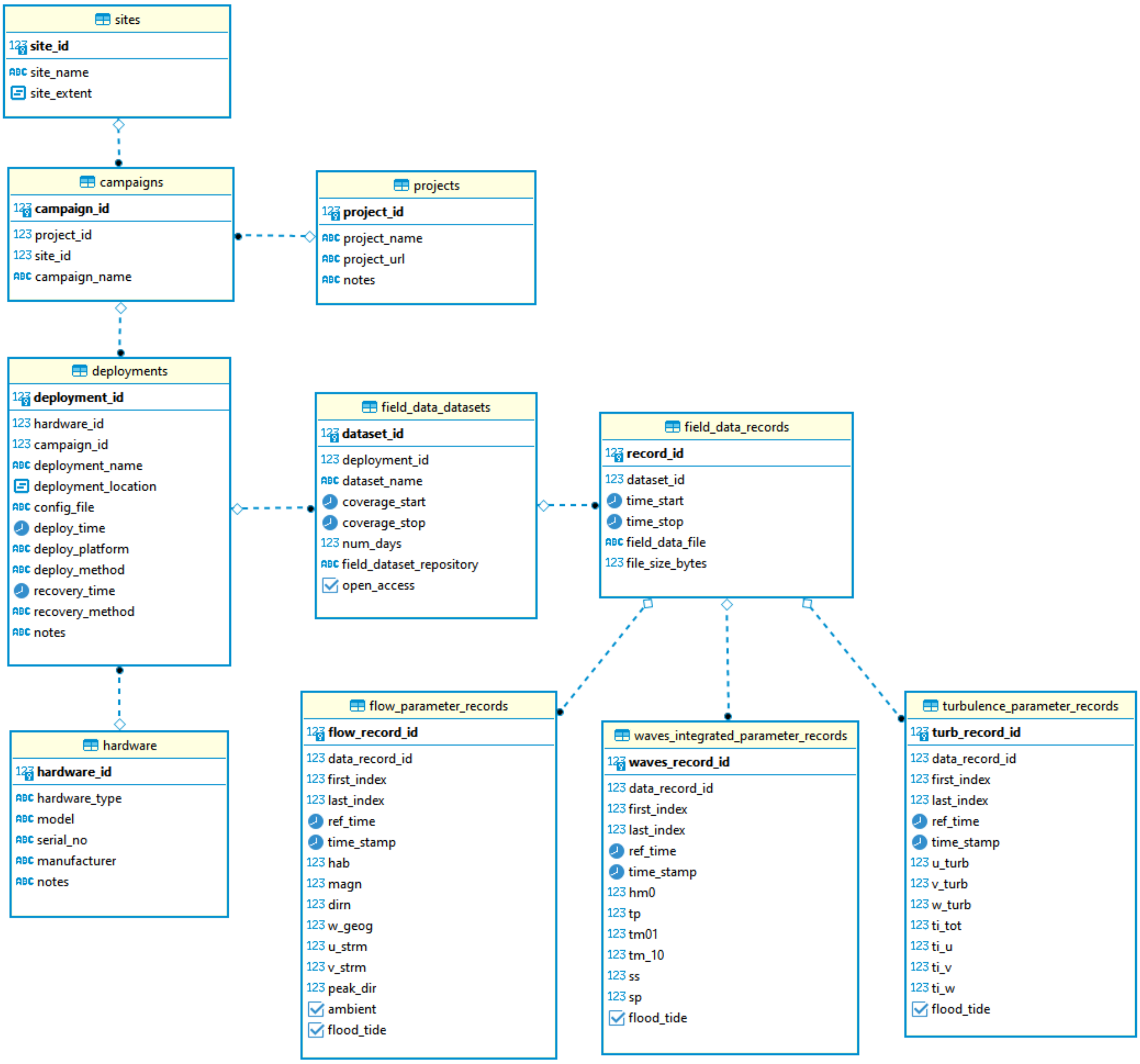


Figure 4-6: Field Data class entity relationship diagram

4.1.4 Machine Data Class

The minimum set of entity fields required to define the field data class are summarised in the entity relationship diagram shown in Figure . For the demonstrator construct, a single machine parameter table has included as we currently have limited open access TEC data available. The machine parameters could be subset based on assemblies and TEC sub-systems, this would allow integration with the reliability database created in RealTide WP1. The site entity is the same entity used for the field data class tables. This provides another method of linking the different data classes together, i.e. on location rather than time.

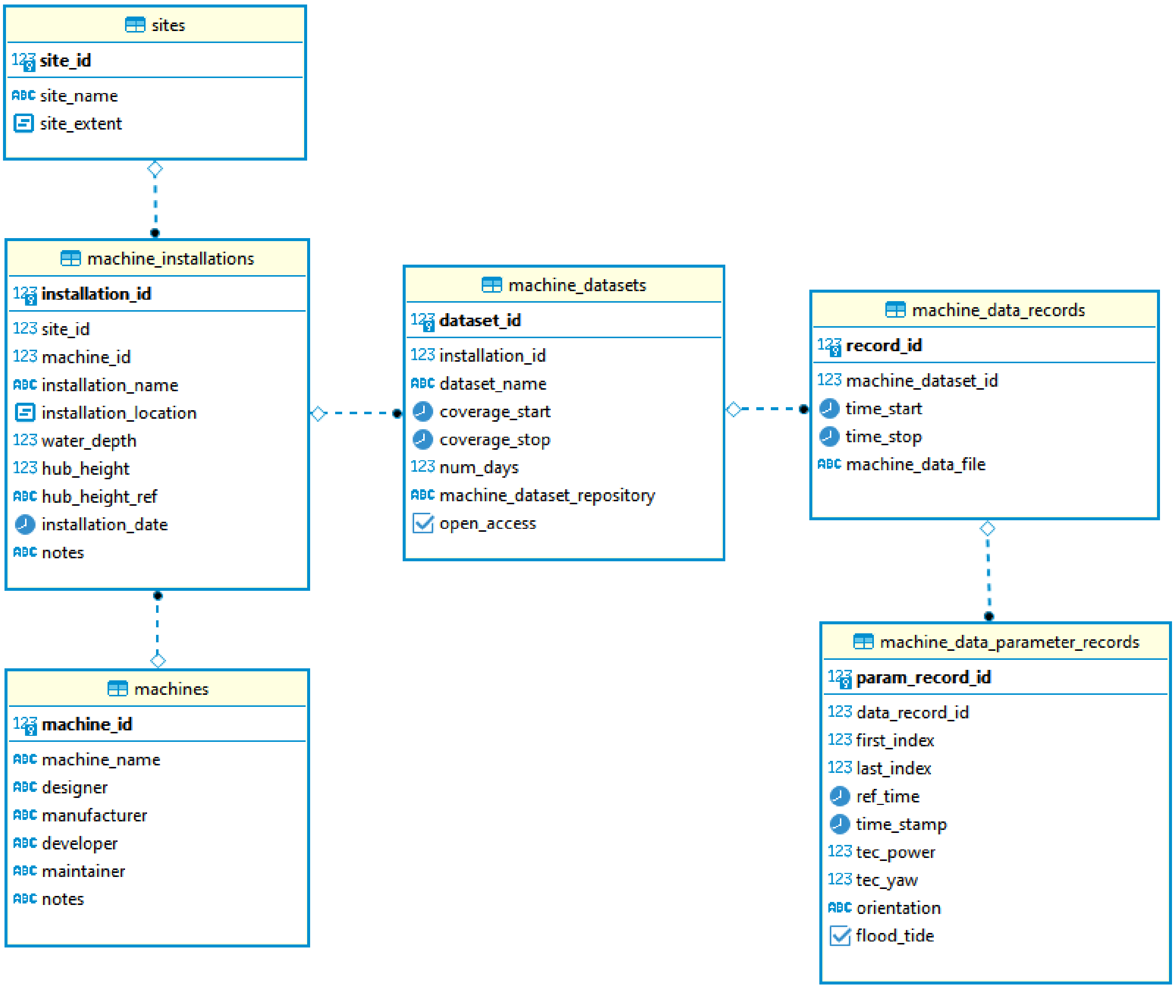


Figure 4-7: Machine Data class entity relationship diagram

4.1.5 Model Data Class

The minimum set of entity fields required to define the field data class are summarised in the entity relationship diagram shown in Figure .

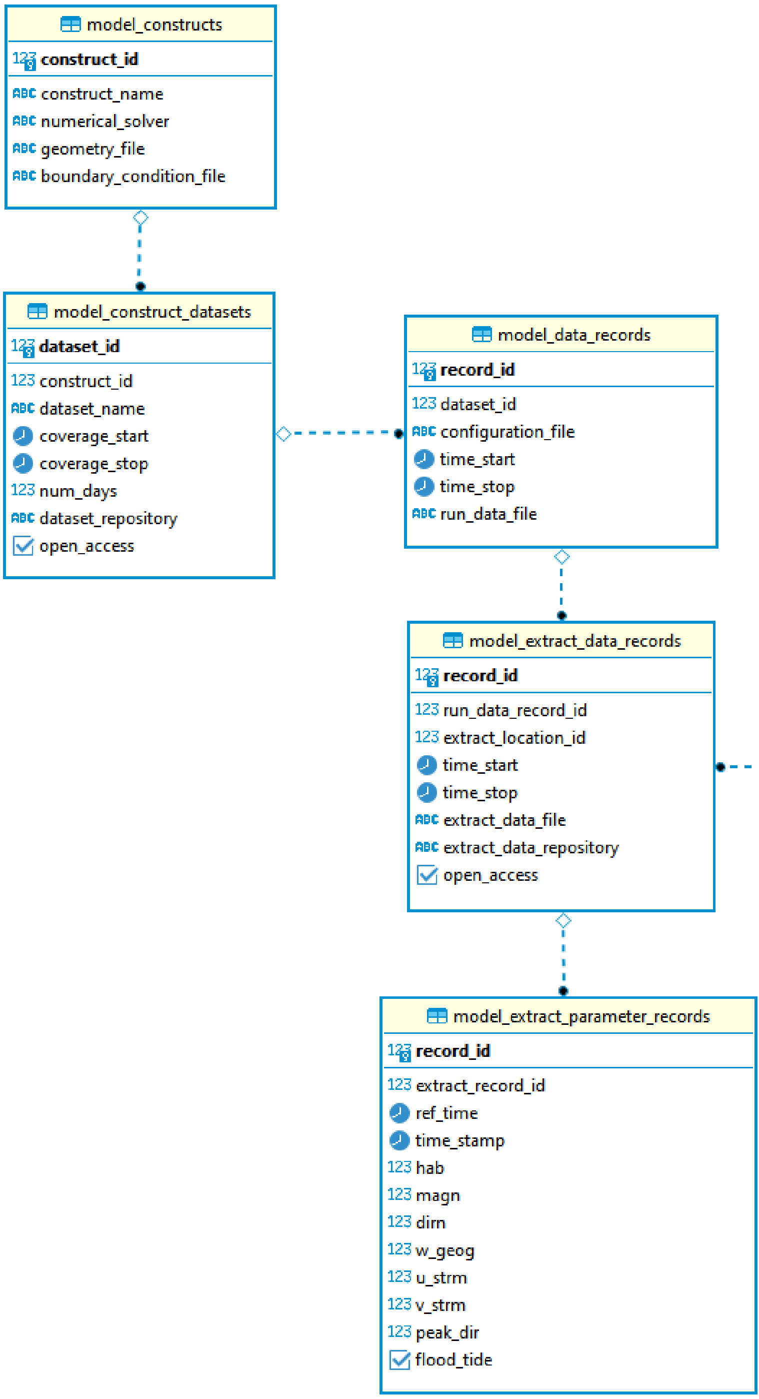
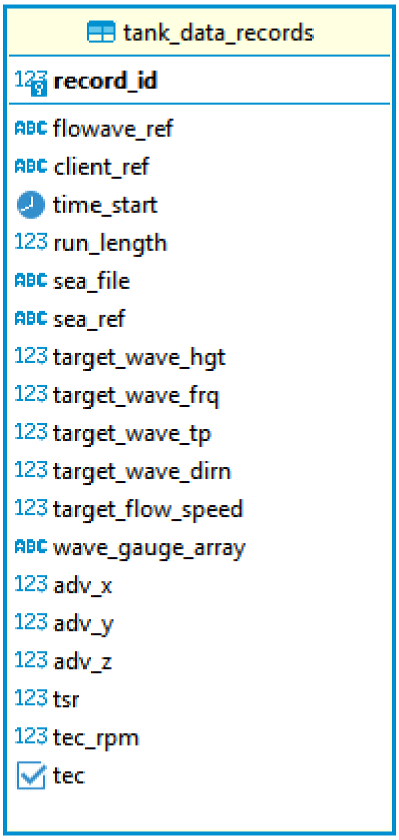


Figure 4-8: Model Data class entity relationship diagram

4.1.6 Test Tank Data Class

The full relational database has not yet been constructed for the test tank data. These data were captured in the interim flat-table (see Figure) database that was used to select test cases for the WP3 BEMT and CFD modelling work. This interim construct was built in MySQL and is served internally from a virtual machine using the University of Edinburgh Eleanor cloud service. Physical modelling (tank-testing) typically generates data relative to multiple coordinate reference systems e.g., tank centre, moveable gantries, instrument rigs, in-situ models etc. In the case of the datasets collated to date there are separate coordinate systems used to define the flow and wave setup to that used to define flow direction and instrument location relative to a positioned TEC or TECS. A single coordinate system will be implemented and the data referencing standardized.



tank_data_records

123 **record_id**

ABC flowave_ref
ABC client_ref
🕘 time_start
123 run_length
ABC sea_file
ABC sea_ref
123 target_wave_hgt
123 target_wave_frq
123 target_wave_tp
123 target_wave_dirn
123 target_flow_speed
ABC wave_gauge_array
123 adv_x
123 adv_y
123 adv_z
123 tsr
123 tec_rpm
☑ tec

Figure 4-9: Test tank data simple flat table used in interim database

4.1.7 Simulation Data Class

The database entity structure for this data class has not been built. The securing, collation and standardization of BEMT and CFD data is on-going. To date, the minimum requirements for these datasets are not clearly defined. These structures will be defined and implemented for the future roll-out.

5 DEMONSTRATOR DATABASE IMPLEMENTATION

The database was constructed and initially tested on a stand-alone PC. This process was necessary to ensure the entities and their associated relationships were sensible, and to determine how to construct SQL queries that reproduce the types of queries considered in the precursor matlab-based data cube construct. SQL constructor and data population scripts were developed and tested as each set of data class entities were built and populated with example data. Resulting in a set of scripts and associated data extracts that can be used to rapidly rebuild the demonstrator database on alternate platforms.

For the purpose of demonstrating the database system philosophy and its potential capabilities, a subset of data from the available data archive have been post-processed and entered into the demonstrator database. A set of example queries are provided that show how to take advantage of the relational structures to generate complex integrated search queries across all data classes.

Database Service Software

The development of the database has been restricted to the used of open-source software tools. The two most popular open-source database service tools are MySQL and PostgreSQL. Both systems have been trialled during the development of the demonstrator database. Both tools have pros and cons, but in general either will suffice. Geo-spatial referencing is a particular feature of the database design that aims to future-proof the system and allow the direct connection to the database from GIS tools. Previously MySQL did not support geospatial referencing, whereas PostgreSQL had a dedicated PostGIS package for this purpose. The most recent versions of MySQL now support geo-spatial referencing, and the methods implemented follow the same standards as those used by the PostGIS package. PostgreSQL has been used to build the final demonstrator based purely of the existing level of expertise with this software.

Demonstrator Service

Through the database development work and on-going discussions with the University's various IS and data services, it has been shown that no sustainable data service solution was possible. The need to improve data archiving and mining has been identified as an infrastructure gap where there is a rapidly growing need. Consequently, a new data facility is to be built by the University of Edinburgh – the Edinburgh International Data Facility (EIDF). The work from this project has been identified as a development case that will be used to scope out requirements and the types of interface capabilities that need to be supported. The development of the EIDF infra-structure was delayed by the COVID-19 outbreak. Services will become available in 2022.

For the purposes of internal testing of the database construct and access controls, the University of Edinburgh cloud service, Eleanor, has been used to host development versions of database. These have been used internally by graduate research students, who have helped with the testing of the systems. The Eleanor cloud services sit within the University firewalls, this limits access to people with a University IS account.

Two implementations of the public access data platform are discussed in Section 6: a medium-term prototype web app, and a long-term professionally curated service. Regarding the prototype, two versions of this service are mentioned here: a development version, and production version. The development version of the prototype web app was built using a remote connection to the main scientific database described in Section 4. This is accessible via VPN, where the service can be run as a localhost server by the RealTide WP2 team members - i.e. University of Edinburgh staff who have been provided a user account on the internally managed Eleanor server instance. The source code for the web application is then run using the internal server module within the latest version of the Django web framework software (the use of Django is discussed in Section 6). The resulting web pages for the development instance are only visible to the local VPN user.). The resulting web pages for the development instance are only visible to the local VPN user.

The production version of the prototype web application runs on a UK-based commercial server using best practice principles (SSH via limited non-root user, Ed25519 key authentication, *etc*.), served using the latest version of Apache. The web application is configured to run its internal database using PostgreSQL, for storing user account data, and headline project metadata for the purposes of front end page context. This service will be launched at www.tidalenergydata.org. This website is publicly available at the time of reporting and will initially feature public access to pre-packaged datasets across in-situ sensing, tank-testing and numerical modelling. Due to the nature of the full service prototype, maintenance and testing will cause interruptions to database service during the first 8 to 12 weeks. During this period users can access multiple pre-packed datasets.

Example Queries

The database construct needs to support a range of different levels of query, for example:

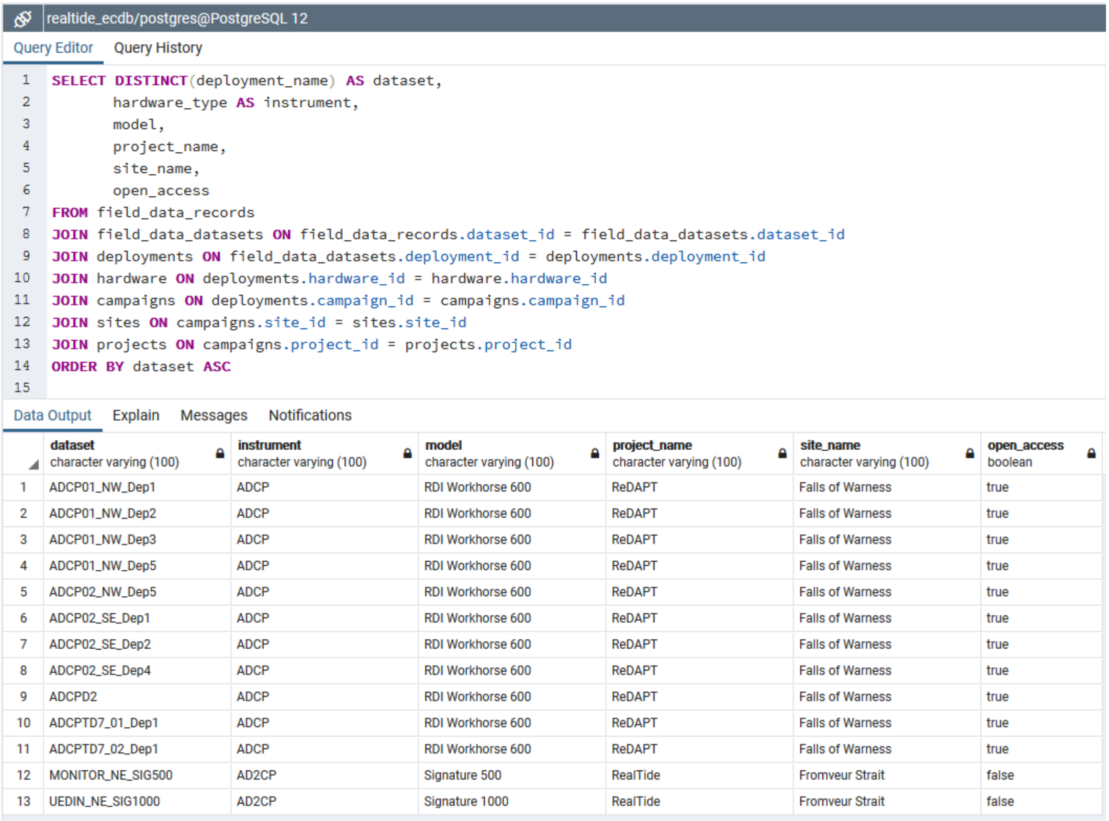
1. Simple catalogue enquiries – e.g. list all available datasets, or all files in a dataset, etc.
2. Select all parameter values for a given dataset or collection of datasets
3. Select data from different classes based on conditions
4. Construct tables of parameters from different sets within a data class based on conditions

There are many more possible combinations, these can built using the methods presented in the following set of example queries.

5.1.1 List Available Field Data Sets

The most basic cataloguing exercise is to list all available dataset in the archive.

```sql
SELECT DISTINCT(deployment_name) AS dataset,
   hardware_type AS instrument,
   model,
   project_name,
   site_name,
   open_access
FROM field_data_records
JOIN field_data_datasets ON field_data_records.dataset_id = field_data_datasets.dataset_id
JOIN deployments ON field_data_datasets.deployment_id = deployments.deployment_id
JOIN hardware ON deployments.hardware_id = hardware.hardware_id
JOIN campaigns ON deployments.campaign_id = campaigns.campaign_id
JOIN sites ON campaigns.site_id = sites.site_id
JOIN projects ON campaigns.project_id = projects.project_id
ORDER BY dataset ASC;
```



```sql
SELECT DISTINCT(deployment_name) AS dataset,
       hardware_type AS instrument,
       model,
       project_name,
       site_name,
       open_access
FROM field_data_records
JOIN field_data_datasets ON field_data_records.dataset_id = field_data_datasets.dataset_id
JOIN deployments ON field_data_datasets.deployment_id = deployments.deployment_id
JOIN hardware ON deployments.hardware_id = hardware.hardware_id
JOIN campaigns ON deployments.campaign_id = campaigns.campaign_id
JOIN sites ON campaigns.site_id = sites.site_id
JOIN projects ON campaigns.project_id = projects.project_id
ORDER BY dataset ASC
```

	dataset character varying (100)	instrument character varying (100)	model character varying (100)	project_name character varying (100)	site_name character varying (100)	open_access boolean
1	ADCP01_NW_Dep1	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
2	ADCP01_NW_Dep2	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
3	ADCP01_NW_Dep3	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
4	ADCP01_NW_Dep5	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
5	ADCP02_NW_Dep5	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
6	ADCP02_SE_Dep1	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
7	ADCP02_SE_Dep2	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
8	ADCP02_SE_Dep4	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
9	ADCPD2	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
10	ADCPTD7_01_Dep1	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
11	ADCPTD7_02_Dep1	ADCP	RDI Workhorse 600	ReDAPT	Falls of Warness	true
12	MONITOR_NE_SIG500	AD2CP	Signature 500	RealTide	Fromveur Strait	false
13	UEDIN_NE_SIG1000	AD2CP	Signature 1000	RealTide	Fromveur Strait	false

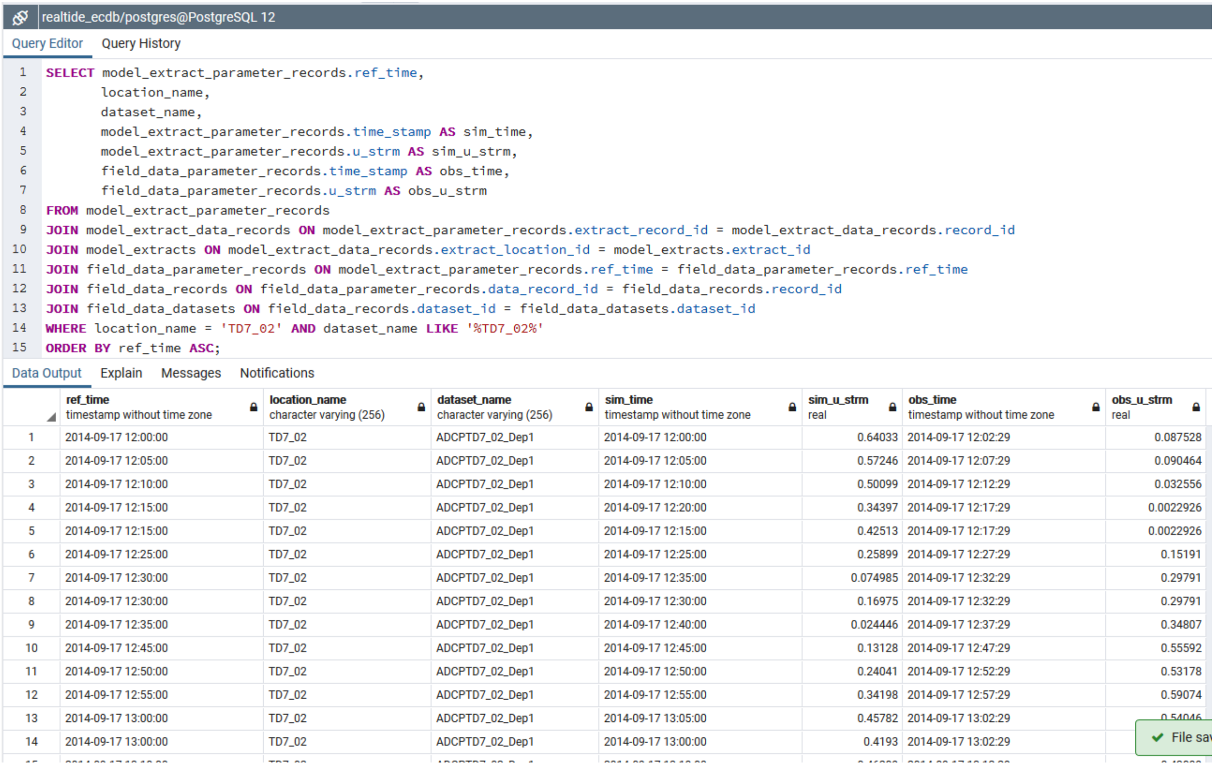
Figure 5-1: A sample from the response to a basic query, showing a tabulated list of all datasets.

This query links all of the parent tables to the field_data_records table using the foreign relational links built into the entity tables. This JOIN process effectively turns the field_data_records table into a large table that includes all of the linked parent entity tables. Figure shows a sample from this query.

5.1.2 Select Field Data and Model Data based on conditions

If the user needs to collect in situ and model data that are co-located, then a search based on common deployment names can be used. In the following example the field and model data that correspond to the ReDAPT ADCPTD7_02 deployment are selected based on the location name and dataset name. A more robust query could be built using the geo-spatial location data for these data (see Figure).

```
SELECT model_extract_parameter_records.ref_time,
   location_name,
   dataset_name,
   model_extract_parameter_records.time_stamp AS sim_time,
   model_extract_parameter_records.u_strm AS sim_u_strm,
   field_data_parameter_records.time_stamp AS obs_time,
   field_data_parameter_records.u_strm AS obs_u_strm
FROM model_extract_parameter_records
JOIN model_extract_data_records ON model_extract_parameter_records.extract_record_id =
     model_extract_data_records.record_id
JOIN model_extracts ON model_extract_data_records.extract_location_id =
     model_extracts.extract_id
JOIN field_data_parameter_records ON model_extract_parameter_records.ref_time =
     field_data_parameter_records.ref_time
JOIN field_data_records ON field_data_parameter_records.data_record_id =
     field_data_records.record_id
JOIN field_data_datasets ON field_data_records.dataset_id = field_data_datasets.dataset_id
WHERE location_name = 'TD7_02' AND dataset_name LIKE '%TD7_02%'
ORDER BY ref_time ASC;
```



```
SELECT model_extract_parameter_records.ref_time,
        location_name,
        dataset_name,
        model_extract_parameter_records.time_stamp AS sim_time,
        model_extract_parameter_records.u_strm AS sim_u_strm,
        field_data_parameter_records.time_stamp AS obs_time,
        field_data_parameter_records.u_strm AS obs_u_strm
FROM model_extract_parameter_records
JOIN model_extract_data_records ON model_extract_parameter_records.extract_record_id = model_extract_data_records.record_id
JOIN model_extracts ON model_extract_data_records.extract_location_id = model_extracts.extract_id
JOIN field_data_parameter_records ON model_extract_parameter_records.ref_time = field_data_parameter_records.ref_time
JOIN field_data_records ON field_data_parameter_records.data_record_id = field_data_records.record_id
JOIN field_data_datasets ON field_data_records.dataset_id = field_data_datasets.dataset_id
WHERE location_name = 'TD7_02' AND dataset_name LIKE '%TD7_02%'
ORDER BY ref_time ASC;
```

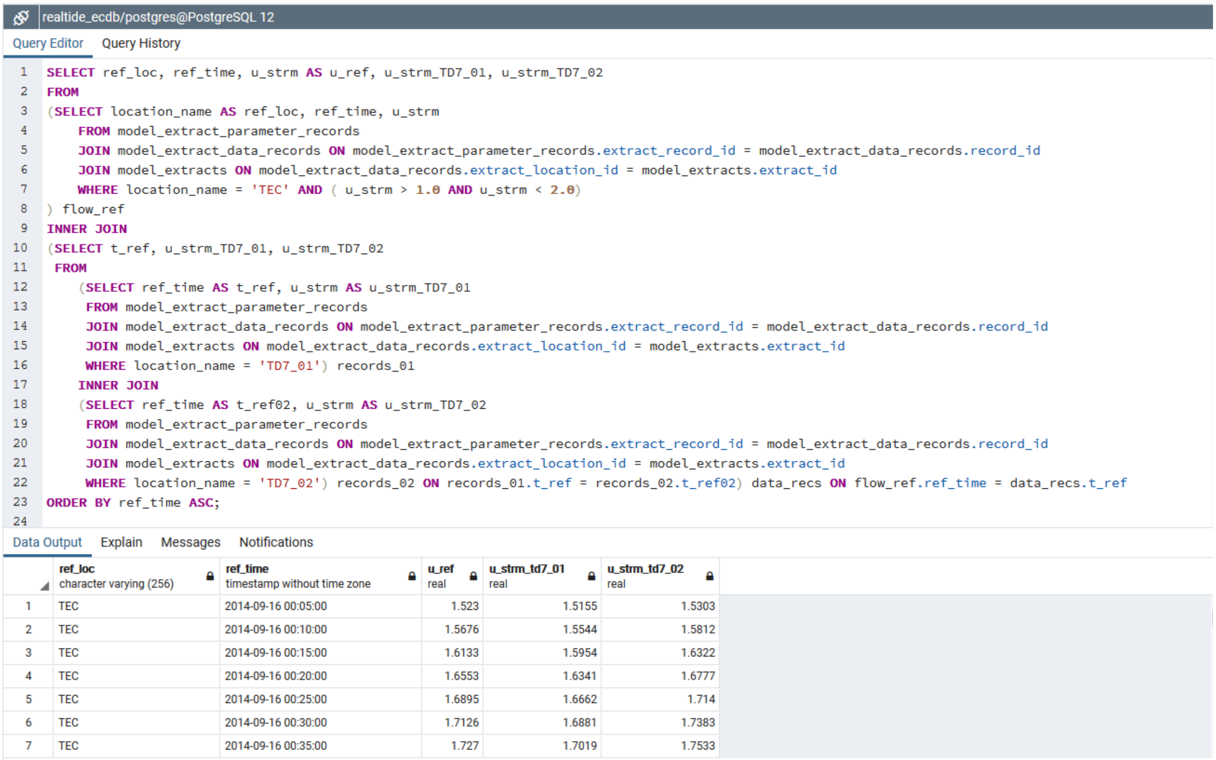
	ref_time timestamp without time zone	location_name character varying (256)	dataset_name character varying (256)	sim_time timestamp without time zone	sim_u_strm real	obs_time timestamp without time zone	obs_u_strm real
1	2014-09-17 12:00:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:00:00	0.64033	2014-09-17 12:02:29	0.087528
2	2014-09-17 12:05:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:05:00	0.57246	2014-09-17 12:07:29	0.090464
3	2014-09-17 12:10:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:10:00	0.50099	2014-09-17 12:12:29	0.032556
4	2014-09-17 12:15:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:20:00	0.34397	2014-09-17 12:17:29	0.0022926
5	2014-09-17 12:15:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:15:00	0.42513	2014-09-17 12:17:29	0.0022926
6	2014-09-17 12:25:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:25:00	0.25899	2014-09-17 12:27:29	0.15191
7	2014-09-17 12:30:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:35:00	0.074985	2014-09-17 12:32:29	0.29791
8	2014-09-17 12:30:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:30:00	0.16975	2014-09-17 12:32:29	0.29791
9	2014-09-17 12:35:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:40:00	0.024446	2014-09-17 12:37:29	0.34807
10	2014-09-17 12:45:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:45:00	0.13128	2014-09-17 12:47:29	0.55592
11	2014-09-17 12:50:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:50:00	0.24041	2014-09-17 12:52:29	0.53178
12	2014-09-17 12:55:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 12:55:00	0.34198	2014-09-17 12:57:29	0.59074
13	2014-09-17 13:00:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 13:05:00	0.45782	2014-09-17 13:02:29	0.54046
14	2014-09-17 13:00:00	TD7_02	ADCPTD7_02_Dep1	2014-09-17 13:00:00	0.4193	2014-09-17 13:02:29	[illegible]

Figure 5-2: Co-located field and model data, corresponding to a RealTide-analysed ADCP dataset .

5.1.3 Create parameter table based conditions

To split out dataset parameters into columns of search parameters based on search conditions, the common reference time field is used to select the synchronous data, and a cascade of INNER JOIN calls is used to build the table. This recreates a sub-set of the flat table originally used to interrogate the ReDAPT data in the original demonstrator built under the ReDAPT project.

```
SELECT ref_loc, ref_time, u_strm AS u_ref, u_strm_TD7_01, u_strm_TD7_02
FROM
(SELECT location_name AS ref_loc, ref_time, u_strm
   FROM model_extract_parameter_records
   JOIN model_extract_data_records ON model_extract_parameter_records.extract_record_id =
        model_extract_data_records.record_id
   JOIN model_extracts ON model_extract_data_records.extract_location_id =
        model_extracts.extract_id
   WHERE location_name = 'TEC' AND ( u_strm > 1.0 AND u_strm < 2.0)
) flow_ref
INNER JOIN
(SELECT t_ref, u_strm_TD7_01, u_strm_TD7_02
 FROM
   (SELECT ref_time AS t_ref, u_strm AS u_strm_TD7_01
    FROM model_extract_parameter_records
    JOIN model_extract_data_records ON model_extract_parameter_records.extract_record_id =
         model_extract_data_records.record_id
    JOIN model_extracts ON model_extract_data_records.extract_location_id =
         model_extracts.extract_id
    WHERE location_name = 'TD7_01') records_01
   INNER JOIN
   (SELECT ref_time AS t_ref02, u_strm AS u_strm_TD7_02
    FROM model_extract_parameter_records
    JOIN model_extract_data_records ON model_extract_parameter_records.extract_record_id =
         model_extract_data_records.record_id
    JOIN model_extracts ON model_extract_data_records.extract_location_id =
         model_extracts.extract_id
    WHERE location_name = 'TD7_02') records_02 ON records_01.t_ref = records_02.t_ref02
   ) data_recs ON flow_ref.ref_time = data_recs.t_ref
ORDER BY ref_time ASC;
```



```
SELECT ref_loc, ref_time, u_strm AS u_ref, u_strm_TD7_01, u_strm_TD7_02
FROM
(SELECT location_name AS ref_loc, ref_time, u_strm
    FROM model_extract_parameter_records
    JOIN model_extract_data_records ON model_extract_parameter_records.extract_record_id = model_extract_data_records.record_id
    JOIN model_extracts ON model_extract_data_records.extract_location_id = model_extracts.extract_id
    WHERE location_name = 'TEC' AND ( u_strm > 1.0 AND u_strm < 2.0)
) flow_ref
INNER JOIN
(SELECT t_ref, u_strm_TD7_01, u_strm_TD7_02
 FROM
    (SELECT ref_time AS t_ref, u_strm AS u_strm_TD7_01
     FROM model_extract_parameter_records
     JOIN model_extract_data_records ON model_extract_parameter_records.extract_record_id = model_extract_data_records.record_id
     JOIN model_extracts ON model_extract_data_records.extract_location_id = model_extracts.extract_id
     WHERE location_name = 'TD7_01') records_01
    INNER JOIN
    (SELECT ref_time AS t_ref02, u_strm AS u_strm_TD7_02
     FROM model_extract_parameter_records
     JOIN model_extract_data_records ON model_extract_parameter_records.extract_record_id = model_extract_data_records.record_id
     JOIN model_extracts ON model_extract_data_records.extract_location_id = model_extracts.extract_id
     WHERE location_name = 'TD7_02') records_02 ON records_01.t_ref = records_02.t_ref02) data_recs ON flow_ref.ref_time = data_recs.t_ref
ORDER BY ref_time ASC;
```

	ref_loc character varying (256)	ref_time timestamp without time zone	u_ref real	u_strm_td7_01 real	u_strm_td7_02 real
1	TEC	2014-09-16 00:05:00	1.523	1.5155	1.5303
2	TEC	2014-09-16 00:10:00	1.5676	1.5544	1.5812
3	TEC	2014-09-16 00:15:00	1.6133	1.5954	1.6322
4	TEC	2014-09-16 00:20:00	1.6553	1.6341	1.6777
5	TEC	2014-09-16 00:25:00	1.6895	1.6662	1.714
6	TEC	2014-09-16 00:30:00	1.7126	1.6881	1.7383
7	TEC	2014-09-16 00:35:00	1.727	1.7019	1.7533

Figure 5-3: Splitting of dataset parameters into searchable conditions, referenced against a common time field

6 DATA ACCESS

Web Platform: Scope and Purpose

The following outlines the specification of the data access platform:

1. A web-based portal is required to allow open access to tidal energy data, including data from field measurement campaigns (metocean and sensor records relating to assets), numerical modelling outputs (including TELEMAC, BEMT and CFD simulations), and tank testing data.
2. The data access platform will contain reanalysis datasets from completed projects, new data generated during this project, and will expand to accommodate future data as this becomes available.
3. The open-access portal will be moderated; new Users will be verified by the Data Controller (DC) and will log in to access the data browser.
4. The tool allows verified Users to browse and query data, to obtain truncated datasets for desired conditions, specific test parameters, durations, providing relevant columns as required.
5. The back-end engine will provide an indirect query mechanism to the main scientific database, to ensure limited and controlled access to read data, and ensure write privileges are withheld. The back end will also avoid imposing unnecessary structure on the main scientific database, by using its own internal database for User and front-end data.
6. The platform should be extensible to allow ongoing maintenance of existing datasets and allow introduction of new data as matter of course. The platform should enhance data provenance with embedded metadata protocols and documented Quality Control (QC). Annotation and commentary should be supported in the future to assist users, pointing directly to data associated with specific projects and publications.
7. The platform should provide documentation and guidance on the datasets and include pre-composed queries for demonstration.
8. The platform must be inherently secure and maintainable. Data handling must obey GDPR rules and regulations regarding personal data and advise Users on aspects of portal use (privacy statements and use of cookies).

Implementation

6.1.1 Long-term hosting

Ongoing discussions are being held with the Edinburgh International Data Facility (EIDF) regarding long-term hosting of the public access data platform. This entails ongoing development, maintenance, and expansion of both the main scientific database, and access-controlled web service, for public viewing and requests for data. As a specialist data organisation, EIDF can provide professional curation and maintenance of very large databases, along with full-stack development and operation of complex web services. As part of their ongoing work, EIDF would manage User access, data requests, and new data deposits from academia and private developers.

As a newly established facility, current activities at EIDF are dictated by the fundamental design and deployment of architectures, core systems and services at this time. This has imposed a requirement to deliver an interim solution for providing public access to the data.

6.1.2 Medium-term hosting

The prototype system serves a range of purposes beyond accelerating the route to public visibility of the data in the absence of immediate support from a specialist data service.

Concurrent, in-house design of both the main scientific database structure and web platform has supported decision-making while defining the main scientific database schema and back-end web application processes. As the main tidal energy database is maintained as a distinct, stand-alone database, no aspects of the web application's design have been imposed on the main database schema. Tidal energy data content, as used directly by the web application, has been partitioned from the main safeguarded scientific database, as this includes commercially sensitive data. This has been achieved by creating an aggregated and parameterised set of summary data to execute public queries, configured as a flat table within the database.

In developing this configuration of public and private databases, tractability and cost of hosting is also a major consideration, with the web service database being three orders of magnitude smaller than the main scientific database. To facilitate this configuration, different solutions for data transactions have been considered and tested during prototyping. This has permitted testing of remote database connections and the subsequent processing of query data through to front-end.

The web platform implemented for the medium-term uses the Python-based Django web framework, which works on the principal of Model-View-Controller. Django has a well-developed set of interfacing libraries which handle modifications and queries for widely used relational databases. This is achieved through 'Models' (representations of database tables), streamlining modifications to the database and extraction of data for web page context. Python acts as the controller language, which conveniently allows for the introduction of any additional Python modules for processing numeric data. Dynamic web content (Views) are then rendered via HTML, incorporating loops and conditionals based on the context delivered by the Python Controller. Django incorporates a range of security features, hashing user passwords for example, with default handling of user logins via token-based POST requests (a discrete way of passing user form data to the server, avoiding visible URL-based client-server communication). User access works on a three-tiered basis, for general users (hereafter referred to as 'Users'), admin, and super-users. As designed, the present web app is configured to have all public pages accessible only via verified general-user login accounts.

The specific web app developed for this project uses a PostgreSQL database, and has been designed for extensibility. Inclusions and adaptations to data sources and queries can be handled by Django's admin user interface pages, which have reserved access rights for staff.

Control of Access: the Data Controller

In the long-term, the nominated role of Data Controller (DC) is yet to be determined, subject to discussions with EIDF. The DC will be responsible for managing personal data, ensuring personal data is gathered, stored and deleted, in line with GDPR. The DC will also manage access to specific datasets according to specific licensing agreements, and User affiliations and responsibilities. The DC will also maintain and expand the main scientific database, supporting data depositors with implementing licenses. For the prototype public access data platform, the DC role will be shared by academic staff based at the University of Edinburgh. At the time of reporting nominated staff are the authors of this report.

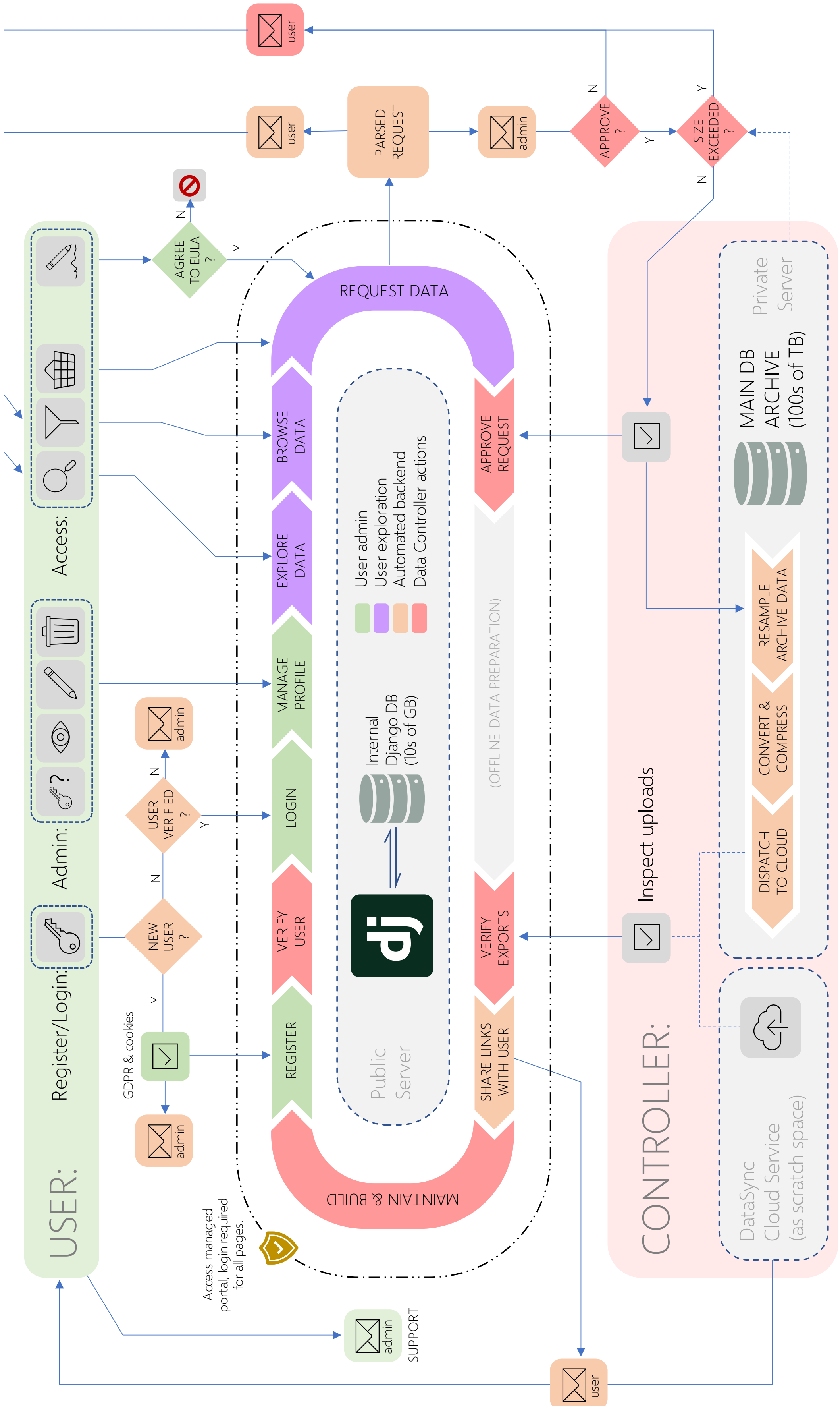


Figure 6-1: User Experience diagram for the web-application

App Design

The User experience is described diagrammatically in Figure . Selected segments from the cascade of web processes, shown in the clock-wise direction in the centre the diagram, are elaborated in the following subsections. User and DC actions are indicated above and below this cycle of processes, including key decision pathways. Actions and data transactions by the User and DC are via one of three modes:

1. Secure form input on the front-end pages;
2. email prompts, to and from the User and DC, with some of these emails generated and sent by the Django back-end;
3. data download via URL links to the DataSync cloud service [28].

Three distinct computational services are identified in Figure :

1. The Django web app: hosted on a commercial public web server, including the internal web app database. The surrounding processes are all handled by Django and are controlled by the DC or User through various front-end pages, generally via POST requests.
2. The private server: accommodating the safeguarded full-resolution marine energy database and data files. Modules from the open source RealTide toolbox are deployed on this service, for extracting, truncating, joining, resampling, converting and compressing data files. Public access to this server is not permitted.
3. UEDIN's DataSync cloud service: this is used to temporarily store compressed data exports for the User to download over the public network.

6.1.3 Register, Verify, and Login

During registration, the User is asked to review and accept the terms and conditions of the websites' use, which clarifies the purpose and methods of processing relating to personal data in the context of GDPR. Once registered, an automated email is sent to the DC to prompt for validation of the new account. If the User reattempts to login after some time, another prompt will be sent to the DC if validation is still pending.

6.1.4 Manage Profile

The User can carry out standard administrative processes on their personal data, in-line with GDPR. The User can inspect, modify and delete any personal data saved on the system at any time. Deleting certain data would entail closure of their account. In the prototype service, aspects of this functionality are yet to be implemented.

6.1.5 Explore Data

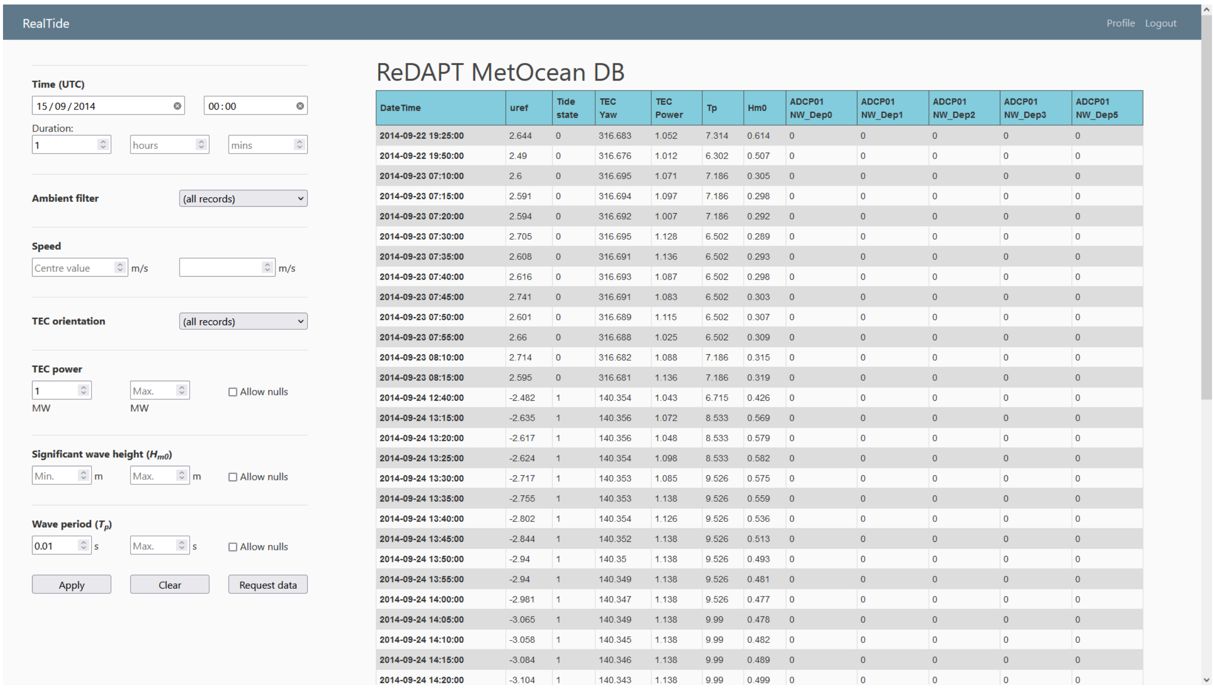
As an entry-point for Users, all projects associated with the main tidal energy database are displayed as tiles, including headline information regarding each project. Each tile links to a detailed page with additional descriptive content, references, DOIs, details of access restrictions, and the relevant End User License Agreement (EULA).

To help introduce the User to a specific test campaign or simulation set, pre-made queries can be included on these data exploration pages.

6.1.6 Browse Data

The User can query the data using the filter form, as shown in Figure . This internally generates an SQL query for executing on the parameterised summary database table, saved on the internal Django database. This SQL query itself is not relevant anywhere else outside this context. The data can be filtered on date and time, ambient meteocean conditions, flow speed, TEC orientation, TEC power and wave conditions (significant wave height and period). When applied, the filter performs dynamic updates on the aggregated 5 minutely table and displays this in the user interface.

The User can build up their desired queries, save these and tailor their data requirements for their purposes. For web-request protocol, the GET method is used to allow ease of query handling by the User, to share and record their queries for future use.



Time (UTC)
15 / 09 / 2014 00 : 00
Duration:
1 hours mins

Ambient filter (all records)

Speed
Centre value m/s m/s

TEC orientation (all records)

TEC power
1 MW
Max. MW
□ Allow nulls

Significant wave height (H_{m0})
Min. m
Max. m
□ Allow nulls

Wave period (T_p)
0.01 s
Max. s
□ Allow nulls

ReDAPT MetOcean DB

DateTime	uref	Tide state	TEC Yaw	TEC Power	Tp	Hm0	ADCP01 NW_Dep0	ADCP01 NW_Dep1	ADCP01 NW_Dep2	ADCP01 NW_Dep3	ADCP01 NW_Dep5
2014-09-22 19:25:00	2.644	0	316.683	1.052	7.314	0.614	0	0	0	0	0
2014-09-22 19:50:00	2.49	0	316.676	1.012	6.302	0.507	0	0	0	0	0
2014-09-23 07:10:00	2.6	0	316.695	1.071	7.186	0.305	0	0	0	0	0
2014-09-23 07:15:00	2.591	0	316.694	1.097	7.186	0.298	0	0	0	0	0
2014-09-23 07:20:00	2.594	0	316.692	1.007	7.186	0.292	0	0	0	0	0
2014-09-23 07:30:00	2.705	0	316.695	1.128	6.502	0.289	0	0	0	0	0
2014-09-23 07:35:00	2.608	0	316.691	1.136	6.502	0.293	0	0	0	0	0
2014-09-23 07:40:00	2.616	0	316.693	1.087	6.502	0.298	0	0	0	0	0
2014-09-23 07:45:00	2.741	0	316.691	1.083	6.502	0.303	0	0	0	0	0
2014-09-23 07:50:00	2.601	0	316.689	1.115	6.502	0.307	0	0	0	0	0
2014-09-23 07:55:00	2.66	0	316.688	1.025	6.502	0.309	0	0	0	0	0
2014-09-23 08:10:00	2.714	0	316.682	1.088	7.186	0.315	0	0	0	0	0
2014-09-23 08:15:00	2.595	0	316.681	1.136	7.186	0.319	0	0	0	0	0
2014-09-24 12:40:00	-2.482	1	140.354	1.043	6.715	0.426	0	0	0	0	0
2014-09-24 13:15:00	-2.635	1	140.356	1.072	8.533	0.569	0	0	0	0	0
2014-09-24 13:20:00	-2.617	1	140.356	1.048	8.533	0.579	0	0	0	0	0
2014-09-24 13:25:00	-2.624	1	140.354	1.098	8.533	0.582	0	0	0	0	0
2014-09-24 13:30:00	-2.717	1	140.353	1.085	9.526	0.575	0	0	0	0	0
2014-09-24 13:35:00	-2.755	1	140.353	1.138	9.526	0.559	0	0	0	0	0
2014-09-24 13:40:00	-2.802	1	140.354	1.126	9.526	0.536	0	0	0	0	0
2014-09-24 13:45:00	-2.844	1	140.352	1.138	9.526	0.513	0	0	0	0	0
2014-09-24 13:50:00	-2.94	1	140.35	1.138	9.526	0.493	0	0	0	0	0
2014-09-24 13:55:00	-2.94	1	140.349	1.138	9.526	0.481	0	0	0	0	0
2014-09-24 14:00:00	-2.981	1	140.347	1.138	9.526	0.477	0	0	0	0	0
2014-09-24 14:05:00	-3.065	1	140.349	1.138	9.99	0.478	0	0	0	0	0
2014-09-24 14:10:00	-3.058	1	140.345	1.138	9.99	0.482	0	0	0	0	0
2014-09-24 14:15:00	-3.084	1	140.346	1.138	9.99	0.489	0	0	0	0	0
2014-09-24 14:20:00	-3.104	1	140.343	1.138	9.99	0.499	0	0	0	0	0

Figure 6-2: The main browser page for public queries to the parameterised data

6.1.7 Request Data

To tailor the User's request, a series of additional actions will be necessary to make a formal data request. Having generated queries to inspect the aggregate data, the form parameters that were used can be saved by the user if they wish to proceed to checkout one or more queries. Once the User has collected their queries of interest, they can proceed to a pop-up window, where they can select specific columns for their output files and submit this request to the DC.

The User will then be provided with a copy of the End-User License Agreement (EULA). They must acknowledge that they observe all the conditions of the license before proceeding. Once confirmed, this action will generate a parsed request message, including the following information:

1. a unique ID for the request message,
2. the time of the request and current version of the database and relevant handling scripts,
3. the User's name and email address,
4. the query parameters that were applied by the User for their request(s),
5. a copy of the EULA, confirming that the User has accepted this agreement.

The Django email client composes a new email with this parsed message directly in the email body. This is sent to both the User and DC. This is the first notification the DC receives regarding this request for data.

6.1.8 Approve Data

The DC is required to approve the request by verifying that the User is permitted to have access to the specific data. As part of this approval process, the DC must confirm that the estimated size of data in the request is acceptable. If rejected, the DC will email the User directly, and log this in the Django web app to close the request. If approved, the DC will accept the request on the web app, and begin processing the data for the User.

6.1.9 Offline Data Preparation

A number of processes are necessary on the safeguarded private server to prepare the data for the User. For the prototype service, these offline data preparation processes are initiated directly by the DC. A series of scripted tasks have been developed and trialled for the individual processes, to select the appropriate data, compose metadata, compress, and send to the cloud service.

As data requests can be of arbitrary length with regard to timestamp ranges, operations on the data can involve accessing multiple raw data files when gathering contiguous data. Similarly, the User-defined query parameters for date and time, duration, and numeric parameters results in subsequent querying of the data for the requested periods and fields. The data is saved as a (set of) NetCDF file(s), and is stored in scratch space on the private server.

During the above process, secondary checks are made on the request relating to the estimated size of the export. If the User has made too large a request, the DC will respond to the User via email to adjust the parameters used for the request, or reduce the number of fields to be included in the export files. If the User makes a legitimate request to gather very large amounts of data, the DC may allow a discussion to make special arrangements for that request. This process, however, would be subject to the discretion of the long-term service provider.

As part of the scripted offline preparation process, the metadata for the data export is then composed. This includes the core metadata from the main scientific database, and supplements this with the following:

- the unique data request ID from the web app,
- the name and email address of the User,
- the date and time of extract (this can differ from the date and time of the original request),
- the back-end database and script versions,
- summarised descriptions of the specific data fields included in the data files, along with details of the data origin to clarify data provenance,
- a copy of the EULA, confirming that the User has accepted this agreement.

The data and metadata are then compressed. The compressed data exports are then uploaded to DataSync, with these files configured with a retention period of one week. Links to these data exports are then generated. If any step in this process fails, the DC is notified.

6.1.10 Verify Export Links and Share with User

The DC is notified by email when the export data is ready for the User. The DC will then be prompted to verify that the uploads to DataSync have been successfully, by testing the links that have been prepared for the User. The DC will then log these links on the Django server, at which point Django will send an automated email to the User. Unless any issue is flagged by the User, the request will close automatically after one week. If the User failed to collect the data, they will have to submit a new request via the web service.

As part of this export verification process, legacy export files on the private server's scratch space are also removed following closure of the request, this may include large uncompressed files.

Maintenance and Integration of New Datasets

Maintenance and expansion of the open-access tidal data resource are subject to operational processes used by the long-term service provider. Whilst many aspects of maintenance and expansion are beyond the scope of this work, the prototype web app and database schema have been designed to demonstrate extensibility.

To register a new project on the web platform, new entries in the internal Django database are created via the staff GUI pages. Adding a new project creates a new tile on the User home page, and a new project specific page providing expanded details on that project. The detailed content is also entered by staff via the back-end GUI.

Other entities may be recorded in the Django internal database which can be added to over time. These include linked publication references and data annotations. These can both have their own tables, which are separate to the project description table. Annotations would be linked via foreign key to a specific project or timeseries (using a 'many-to-one' database relationship). In general publications may not be rigidly associated with a single project and can be mapped to any project via a through table (via 'many-to-many' relationships).

End User License Agreements

As licensing conditions are dictated on a project-by-project basis, license conditions are in themselves the overriding factors in what defines a 'project' within the internal web app database (the main safeguarded database can, however, describe projects on different terms). To maintain robust procedures for data access rights, the approvals process managed by the DC should be unambiguous regarding the commercial sensitivity of the data, and any other aspect of the predefined license agreement for that dataset. When a User makes a request for data, they will be asked to agree to the associated terms, as identified by the Django back-end. The correct license must therefore be configured by the DC when the project is initially registered on the web portal. As the DC's subsequent approval is used as the formal validation of access rights, there must again be a clear link between data and corresponding license agreement.

Where projects have a mixture of access agreements, relating to commercial and open-access data for example, two separate projects should be registered on the service, with the corresponding EULA and data subsets appropriately mapped to each 'sub-project'.

All projects will be licensed, as a minimum, using the open MIT License.

7 DATASETS AVAILABLE

RealTide deliverable D1.5 [4] identified a number of potential data sources for inclusion in the database construct. Combing secured open-access legacy datasets with datasets generated in RealTide WP2 and WP3 -including previously un-processed legacy datasets - has produced a suitable aggregated dataset for this demonstrator database construct. The largest available open-access data sets are the EMEC site data and DeepGen IV data generated during the ReDAPT project, and the extensive set of FloWave Test Tank data generate through projects funded by UK SuperGen Marine projects, covering FloWave tank characterization and generic three-bladed HATT scale model turbine tests. These data are supplemented with new open-access components of the RealTide FloWave testing campaigns, regional model data generated for RealTide WP2, and CFD analysis carried out for RealTide WP3. Datasets that have been pre-identified for inclusion in the database are shown in Table 7-1. It is anticipated that as ongoing and future internal and external research projects generate outputs, the RealTide database can host and help to accelerate dissemination and uptake. Five datasets, one from each of the classes Tank and Model, and three from Field are being integrated with the highest priority (see Table 7-1).

Table 7-1: Status of data for inclusion in demonstrator database

Data Source	Deployment / Experiment / Simulation	Data Class	Secured	Catalogued	QC Applied	Processed	Standardized	Implemente	Open Access
ReDAPT	2011 - Multiple ADCPs	Field	🟩	🟩	🟨	🟨	🟥	🟨	Y
	2013 - Multiple ADCPs	Field	🟩	🟩	🟨	🟨	🟥	🟨	Y
Priority->	2014 ADCPTD7a not previously processed/ released.	Field	🟩	🟩	🟩	🟩	🟥	🟨	Y
Priority->	2014 ADCPTD7b not previously processed/ released.	Field	🟩	🟩	🟩	🟩	🟥	🟨	Y
	DeepGen IV Basic averaged turbine status	Mach.	🟩	🟨	🟨	🟨	🟥	🟨	Y
	DeepGen IV High Resolution extensive dataset	Mach.	🟨	🟨	🟨	🟨	🟥	🟥	N
	MIKE-21 EMEC Model (proprietary)	Model	▨	▨	▨	▨	▨	▨	N
SuperGen Marine	Generic TEC in combined wave-currents	Tank	🟩	🟨	🟨	🟨	🟨	🟥	Y
	Generic TEC array in combined wave-currents	Tank	🟩	🟨	🟨	🟨	🟨	🟥	Y
FloWTurb	Ambient-conditions rotor plane mapping	Tank	🟩	🟨	🟨	🟨	🟨	🟥	Y
MARINET	Round Robin 2 (TBD)	Tank	🟥	🟥	🟥	🟥	🟥	🟥	Y
SABELLA	2016 ROWE Deploy 01 data recovery/re-process	Field	🟩	🟩	🟨	🟨	▨	▨	N
	2016 ROWE Deploy 02 data recovery/re-process	Field	🟩	🟩	🟨	🟨	▨	▨	N
RealTide	2019/21 Long-duration ADCP Fromveur (SIG500)	Field	🟩	🟩	🟨	🟨	▨	▨	N
	CADP-Sub-System Test (USA) [29,30]	Field	🟩	🟩	🟨	🟨	🟥	🟥	Y
	CADP Sub-System Test & Flowave Ambient [31,32]	Tank	🟩	🟩	🟩	🟨	🟥	🟥	Y
	2019/21 Long-duration ADCP Fromveur (SIG1000)	Field	🟩	🟩	🟨	🟨	▨	▨	N
	2021 EMEC C-ADP Advanced SBD	Field	🟩	🟨	🟥	🟥	🟥	🟥	Y
Priority->	2021 EMEC C-ADP SIG500	Field	🟩	🟩	🟨	🟥	🟥	🟥	Y
Priority->	ORK_BASE	Model	🟩	🟩	🟩	🟨	🟥	🟨	Y
	FOW_HiRes	Model	🟩	🟩	🟩	🟨	🟥	🟥	Y
	IS_MODEL_{A,B,C,D} (Fromveur)	Model	🟩	🟩	🟩	🟨	▨	▨	YN
	Ambient-conditions rotor plane mapping gap-filling	Tank	🟩	🟩	🟨	🟨	🟥	🟥	Y
Priority->	Generic TEC Benchmarking Set (at Flowave)	Tank	🟩	🟩	🟨	🟥	🟥	🟥	Y
	BEMT – WP3	Sim.	🟩	🟩	🟨	🟩	🟥	🟥	YN
	SOWFA – WP3	Sim.	🟩	🟩	🟨	🟩	🟩	🟥	YN
	STAR CCM+ CFD –WP3	Sim.	🟩	🟩	🟨	🟩	🟥	🟥	YN

YES 🟩 PARTIAL 🟨 Planned 🟥 Not Planned ▨

In-Situ (EMEC) Dataset: Example

The RealTide project deployed in August 2021[1] an advanced flow-measurement platform comprising multiple sensors at the north west region of the tidal energy test site at the European Marine Energy Centre (EMEC). Unlike some other project-developed datasets, which are restricted due to commercial confidentiality, all successfully acquired datasets from the August-2021 multi-instrument campaign will be made publicly available. At the time of reporting of Deliverable D2.3, a dataset comprising a 5-beam ADCP measuring sea-bed to sea-surface for approximately 48 days has passed preliminary analysis and will therefore be incorporated in the D2.3 database. For further information please see RealTide Technical Report D2.2 on relevant measurement campaigns. A data visualisation of 40 seconds of the 48-day dataset is shown in Figure 7-1 where the impact of wind-driven surface gravity waves is evident on the mean tidal current (shown as a grey transparent surface bisecting the wave field). Two further high-priority sets have been identified (see Table 7-1) and Deliverable D2.2 for further information.

Table 7-2: RealTide final measurement campaign: summary specification of component dataset

Instrument	ADCP
Manufacturer	Nortek
Frequency	500 [kHz]
Number of beams	5
Sample Rate	4 [Hz]
Bin Size	1 [m]

Measurement Start	02/08/2021 12:00
Measurement End	18/09/2021 10:35
Duty Cycle	12 mins. on, 8 mins. off
Profiling Range	~1 [m] to ~30 [m] (surface)
UTM Easting: UTM Northing: UTM Zone:	509968.4 6556364.2 30V

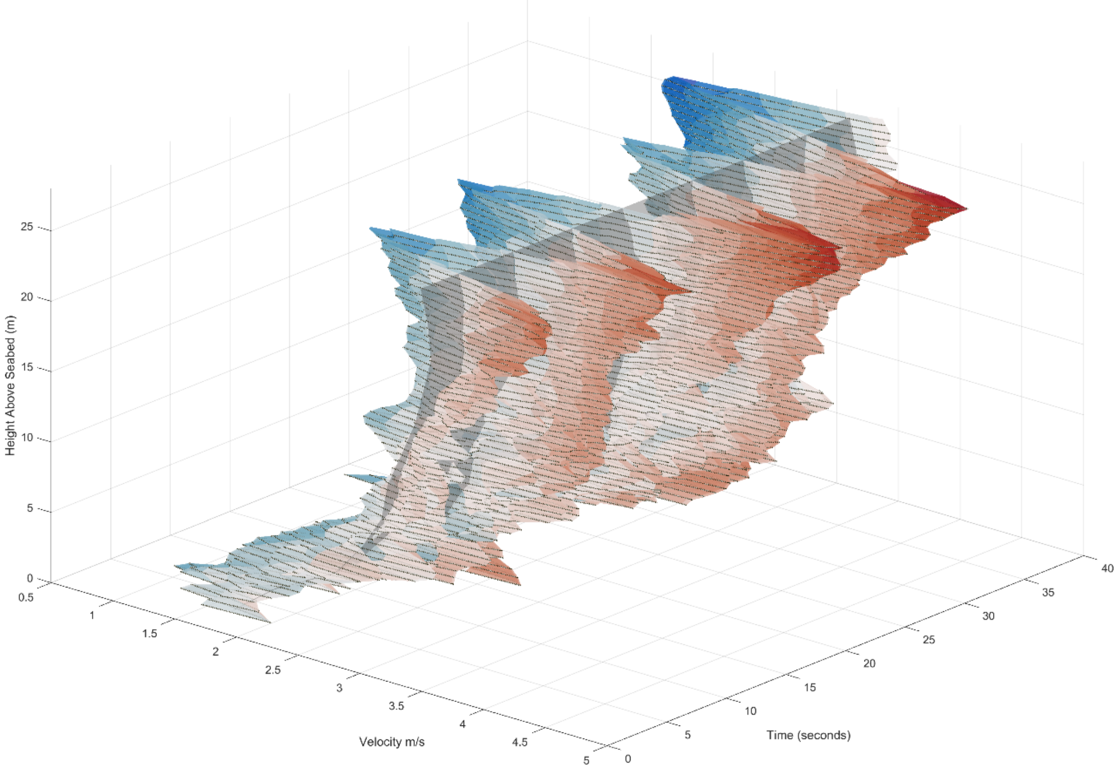


Figure 7-1: RealTide final measurement campaign: high resolution and energetic open data set showing streamwise velocity from a 5-beam Signature 500 in ~30m water depth during period of wave activity.

[1] The final measurement campaign of RealTide comprising an advanced flow measurement platform was delayed significantly through 2021 due to multiple logistical challenges caused by Covid19.

Test-Tank (FloWave) Dataset: Example

A dataset was targeted and acquired at the Flowave facility, University of Edinburgh, in the summer of 2021 with the purpose of serving (via the RealTide database) as a readily accessible high quality and well-documented benchmark set of tests. It is anticipated that this will be useful by a wide range of stakeholders for the development of simulation tools for tidal energy applications. The FloWave facility is a circular basin, 25m in diameter and a working water depth of 2m. It features 168 active absorption wave makers, around the 360 degree perimeter of the tank and is equipped with 28 flow drives in a circular arrangement. The inner part of the tank floor, with a diameter of 12m, is raisable, allowing dry installation of equipment and models. The wave and current systems are independently controlled, allowing for the recreation of realistic and directionally complex wave-current sea sates.

The test layout is shown in Figure Figure 7-2: Schematic showing scale model TEC and instrument layout during RealTide testing at Flowave. where measurement locations in the X-Z plane along the tank-centreline are indicated with circles. Data was also collected laterally across the tank at varying spatial resolution. The data has been mapped to a common coordinate system. The origin of the right handed co-ordinate system is in the centre of the tank when it is submerged, and the z-axis is positive upwards. Consequently, the still water surface has a positive z value of 2m. The x-axis is pointing in the positive flow direction. Instrumentation, such as wave gauges, are typically attached along the gantry, which spans parallel to the y-axis of the global tank co-ordinate system, and can move laterally in the x-axis. The turbine was installed on a six degree of freedom (DoF) force plate.

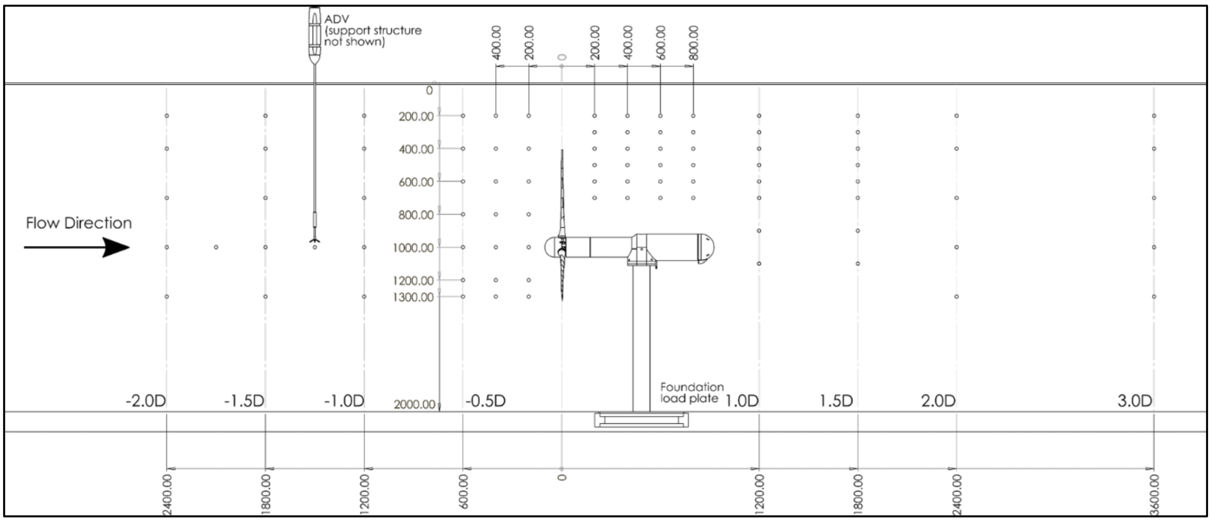


Figure 7-2: Schematic showing scale model TEC and instrument layout during RealTide testing at Flowave.

7.1.1 Test conditions

The turbine was tested under a series of combined wave-current test conditions, as well as a current only flow (see Table 7-3). The wave height (0.1m) and frequency (0.4Hz) refer to the generated wave height and frequency (calibrated to be achieved at the model location in 'open tank' i.e., with no model turbine present. The current speed is the inflow streamwise (*u*) velocity. Waves were generated in 6 different directions: directly following and opposing the current, and ±20° following and opposing the current.

7.1.2 Generic 3-Bladed Horizontal Axis Tidal Turbine

The three-bladed horizontal axis turbine (HATT) model was developed by the University of Edinburgh and is described in [33]. The turbine is 1.2m in diameter (D) and was operated for this subset dataset in angular velocity control mode (via installed 14-Bit incremental quadrature encoder) at a fixed Tip

Speed Ratio (TSR) of 5.5. The turbine was instrumented with a torque-thrust transducer in the turbine nacelle and root bending moment (RBM) transducers on each blade measuring flapwise loads. The turbine was installed on a 6-DOF loadcell.

Table 7-3. Test-tank operating conditions for an example RealTide dataset: the open benchmarking dataset

Wave Height [m]	Wave Frequency [Hz]	Current Speed [ms^{-1}]	Wave Direction (relative to current) [°])
0.1	0.4	0.8	0
0.1	0.4	0.8	180
0.1	0.4	0.8	-20
0.1	0.4	0.8	20
0.1	0.4	0.8	160
0.1	0.4	0.8	200
--	--	0.8	current only

7.1.3 Instrumentation

Three wave gauges were located concurrently with the ADVs on the rear of the moveable gantry, situated 0.30 m (0.25xRotor Diameter) either side of the ADVs. A further 4 wave gauges were mounted on the front of the gantry (1.6 m downstream of the first three wave gauges). All wave gauges moved longitudinally (x-axis) with the gantry, and the first 3 wave gauges also moved laterally (y-axis) with the velocity instruments.

Table 7-4: Instrumentation summary information

Instrument	3-Bladed Generic TEC
Manufacturer	UEDIN
Number of units	1
Position	Near-tank centre
Sample Rate	1024 [Hz]
Parameters Measured	

Instrument	Acoustic Doppler Velocimeter
Manufacturer	Nortek
Number of units	2
Position	600 [mm] apart (y-direction) on traversing rig (x,y and z)
Sample Rate	96 [Hz]
Parameters Measured	$\bar{u}(x,y,z)$ $\bar{\theta}(x,y,z)$ $TI(x,y,z)$

Instrument	Wire resistive wave gauge array
Manufacturer	FloWave / ED
Number of units	7
Sample Rate	128 [Hz]
Position	4 along centre-line
Parameters Measured	$MSL(x,y)$ $H_{m0}(x,y)$ $T_p(x,y)$

Instrument	6DOF Load Cell
Manufacturer	AMTI OR-6-7
Number of units	1
Sample Rate	1024 [Hz]
Position	Under TEC monopile
Parameters Measured	$F(x), F(y), F(z),$ $M(x), M(y), M(z)$

Model (Orkney) – Example

Multiple 3D models of the Fromveur channel, France have been designed and operated within RealTide WP2. Due to the commercially-sensitive nature of the data, the methodology derived and lessons-learned from these model builds was transferred to cover an open-access site to allow data-sharing beyond the RealTide project. The site chosen was Orkney waters, targeting at high resolution the Fall of Warness tidal energy test site operated by the European Marine Energy Centre (EMEC). Multiple in-situ datasets are already held for this region which can be used for model calibration and validation.

A subset of these EMEC models has been prioritised for implementation in the database to allow cross-comparison in space and time of model outputs at the locations of co-deployed ADCPs. The dataset is summarised in Tables 7-5 and 7-6 and Figures 7-3, which also illustrates bathymetry and mesh density. Further information on this subset, and related datasets can be found in RealTide Deliverable D2.2.

Table 7-5: Model subset data example: EMEC base model for month of November 2014

Model	Time Step (s)	N Days	N Layers	Average Run Time per Day	2-D File Size (GB)	3-D File Size (GB)	Subset Data Volume (GB)
ORK_BASE	10	SUBSET 30 Days covering November 2014	15	01:15	1.0	10.3	~250 data reported at coincident ADCP locations

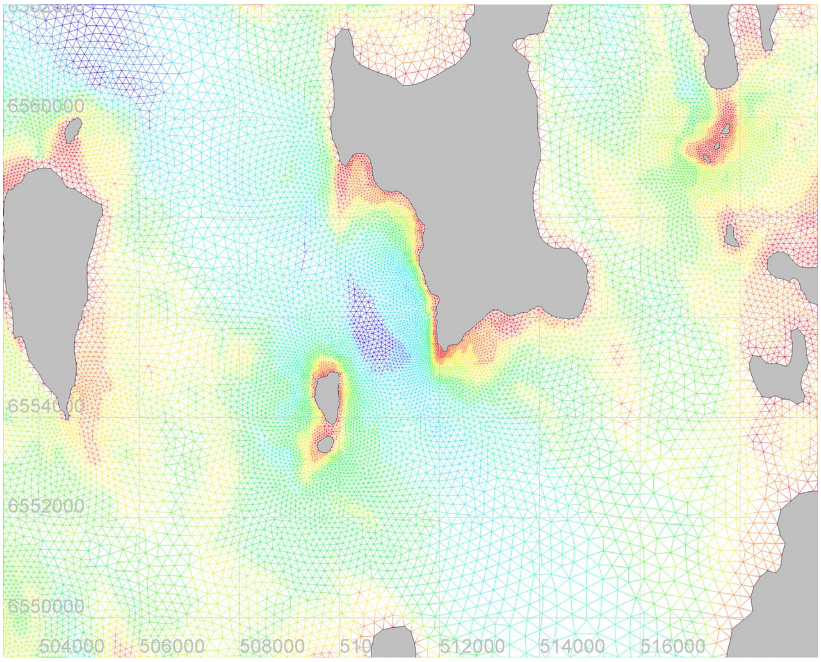
Orkney Model – ORK_BASE	
Mesh Statistics Number of Nodes: 157699 Number of Elements: 296828 Number of Layers: 15 Min. Edge Length: 6m Max. Edge Length: 1530m Falls of Warness Edge Length: ~30m Mainland Edge Length: ~200m Open Boundary Edge Length: ~1000m Mid-Channel Edge Length: ~80m	 6560000 6554000 6552000 6550000 504000 506000 508000 510 512000 514000 516000

Figure 7-3: Orkney base model domain outline description

Table 7-6: Model fields stored for each run

Field Type	Model Variable	Description
2-D	U(T,N)	Depth-averaged eastward component of velocity
	V(T,N)	Depth-averaged northward component of velocity
	H(T,N)	Total water depth (bathymetry + surface elevation)
	S(T,N)	Surface elevation
	B(T,N)	Bathymetry (or mean water depth)
	W(T,N)	Bottom friction
3-D	Z(T,L,N)	Sigma layer positions
	U(T,L,N)	Eastward component of velocity
	V(T,L,N)	Northward component of velocity
	W(T,L,N)	Upward component of velocity.

8 END USER FEEDBACK

Feedback to date on the form of the database and the content has been on an informal basis with end-users within the University and the RealTide consortia. Presentation of the concept and methods have been presented at international meetings and during side events, where there was general feedback from participants. The overall comment has been that this type of service would be very welcome and very useful, the most common complaint from the community has been the difficulty regularly encountered when trying to access data suitable for MRE research, design and development work. The process of linking together field data, model data, tank data and simulation data was identified as being both a useful and time-saving feature of the database. The ability to target specific environmental system states was seen as useful by the research community.

Features to Add

Based on feedback from internal users the following improvements/modifications have been suggested:

- Include a list of the variables available within specific datasets for cataloguing purposes – it is helpful for users to know what variables are available when making a data request. This can be used to reduce the data volume returned.
- Embed instrument configuration information for data cataloguing purposes – currently the database points to a file that contains the configuration information. User need sufficient information to determine whether the data stored in the database is suitable for their intended end-use; this can be provided by the core configuration parameters.
- Identify when field data are affected by the presence of either a deployed turbine or any infrastructure that may generate an up-stream wake that will alter the measured flow away from “undisturbed” site data - this is important for data used to validate hydrodynamic models.
- Include a meteorological data class – local weather information can be used to better understand and attribute non-tidal signals and processes.

The intention is to open the interim database service for beta testing and collect feedback before the completion of the RealTide final report.

9 CONCLUSIONS

A database methodology for integrating a wide variety of MRE data with the functionality to support targeted data extraction based on user-defined system state criteria has been successfully designed, built, populated with a demonstrator dataset, and tested.

The components for an interim service have been constructed, this service will be implemented in the last quarter of 2021. The collation and processing of the data to be added to this service is on-going, and software tools to automate some of these processes are near completion. SQL scripts are available to build the demonstrator database from scratch on a PostgreSQL server. There is a corresponding set of input files that populate the database with the demonstration data.

Discussions are on-going with the newly established Edinburgh International Data Facility (EIDF) to build a long-term service solution. The development of the EIDF services have been significantly delayed by the COVID-19 pandemic. Currently their focus is on finalising infrastructure and systems. They aim to be ready for initial data cataloguing and archiving in early 2022. The output from this project has been identified by EIDF as a development case that can be used to scope out requirements and the types of interface capabilities that need to be supported.

There is strong support both within the UEDIN research community and the wider MRE community for the continued development of this construct into an operational service. FASTWATER, a recently funded (UK SuperGen ORE Hub) research project, aims to integrate data products into this service. There is also interest in incorporating data from other facilities, such as UEDIN FASTBLADE. As more users are identified, the tools for managing data integration and post-processing will be extended