NIWA
Taihoro Nukurangi

Interoperable Models for Land and Water

Framework selection and preliminary design.

Prepared for Our Land and Water Science Challenge

Prepared by:
Sandy Elliott (NIWA)
Serina Callachan (Ministry for the Environment)
Nic Conland
Chris Daughney (GNS)
Hans Eikaas (DairyNZ)
David Eyers (University of Otago)
Alex Herzig (Landcare Research)
Bethanna Jackson (Victoria University)
Juan Monge (Scion)
Paul Johnstone (Plant and Food Research)
Asaad Shamseldin (University of Auckland)
Jo Sharp (Plant and Food Research)
Tarek Soliman (Landcare Research)
Gabriella Turek (NIWA)
Iris Vogler (AgResearch)
Steve Wakelin (Scion)

Further contributors to reviews:
Wayne Schou (Scion)
Zara Rawlinson (GNS Science)
Mike Toews (GNS Science)
Catherine Moore (GNS Science)
Lawrence Kees (Environment Southland)
Rachael Millar (Environment Southland)
Tim Ellis (Environment Southland)
Brent King (Greater Wellington Regional Council)
Natasha Tomic (Greater Wellington Regional Council)

For any information regarding this report please contact:

Sandy Elliott
Principal Scientist – Catchment Processes

+64-7-859 1839
sandy.elliott@niwa.co.nz

National Institute of Water & Atmospheric Research Ltd
PO Box 11115

Hamilton 3251

Phone +64 7 856 7026

NIWA CLIENT REPORT No: 2017239HN
Report date: July 2017
NIWA Project: AGR17204

Quality Assurance Statement		
	Reviewed by:	Annette Semadeni-Davies, PhD. Urban Aquatic Scientist, NIWA Auckland
	Formatting checked by:	A.Wadhwa
	Approved for release by:	Neale Hudson Group Manager – Catchment Processes NIWA Hamilton.

Contents

Tables

Figures

Executive summary

Report purpose and background

This report presents a proposal (with accompanying rationale), for the development of an interoperable modelling system suitable for integrated spatial assessment the economic, production and environmental implications of land use and land use change, operating at farm to catchment scales. The anticipated uses of the system include assessment and accounting of production potential and water quality contaminant dynamics. This is the first stage of a three-stage programme for the Our Land and Water Science Challenge (OLW). The programme is based on the idea that the availability of better, more trusted, and targeted modelling tools within an interoperable modelling framework will result in more effective use of integrated modelling for improved production and environmental management. Stage 1 of the programme, which is the subject of this report, establishes a plan for work to be undertaken in Stages 2 and 3 of the programme. Additionally, this stage of the programme identifies modelling objectives, reviews previous interoperability efforts, defines an overall framework for New Zealand applications, and identifies an agreed initial set of models, data sources, and visualisation components, along with a software framework for linking and accessing the components. Stage 2 will take place from August 2017 to June 2019, and will focus on fully implementing and demonstrating the initial set of models and data within the framework, and will also prepare a plan for Stage 3. Stage 3 (for tranche 2 of OLW) proposes to enrich the range of models in the framework and demonstrate and evaluate the use of the framework in multiple contexts, including linking to social and cultural attributes.

Methods

The programme entails the establishment of a Governance Group working in tandem with a Technical Group, to ensure focus, relevance, course-correction and evaluation of the programme. The Governance Group includes representatives of partners or other key organisations and is responsible for steering the work, resolving IP and resourcing issues, and providing links to other stakeholders. The Technical Group incorporates specialists from a range of science providers, covering the required areas of technical expertise- including catchment hydrology, production systems, water quality, agro-economics and computer science. They were responsible for preparing this report.

Future stages of the programme will rely heavily on co-funding from programme partners, the Challenge itself providing only providing part-funding. Stage 1 of the programme builds the case for such funding by evaluating modelling approaches and developing an implementation plan targeted to the objectives provided by the Governance Group, with accompanying rationale. This report assists funders to see where their models fit in with a broader landscape of models and interoperable systems, contributing to the value proposition for this programme.

The objectives of the modelling system were provided by the Governance Group, identified from two meetings and a survey, and within overarching OLW requirements.

The Technical Group reviewed and evaluated seven interoperability frameworks and seven integrated models, with emphasis on systems used previously in New Zealand. Each system was described according to a set of common attributes and experiences with the system were documented, to lay the foundation for framework selection. The frameworks were evaluated by means of a workshop and subsequent video-conferences, and a framework was selected for use in Stage 2 on the basis of this evaluation. New and emerging technologies related to interoperability were also reviewed, to ensure that the programme can harness these if they are relevant.

A high-level structure for organising data sources, models, and model outputs was developed, and populated with list of specific items. From this list, a subset of models were selected for initial implementation in order to meet a range of key objectives. These were moulded into a specific staged work plan, with indicative resourcing. Key risks and mitigations were also identified.

Key findings and recommendations

Objectives.

Key objectives from the OLW programme include providing a nationally applicable system of models, in an open source interoperability framework, which draws on national datasets where possible and uses interoperability standards. The system should be suitable for integrated and spatial assessment of the economic, production and environmental implications of land use and land use change at farm to catchment scales. The Governance Group reinforced these objectives, with particular emphasis on linking from farm to catchment scale, biophysical modelling including time-stepping models, and including both water demand and quantity, with less emphasis on economic aspects in the initial set of models. The survey of the Governance Group fleshed out and expanded on these objectives by identifying benefits of interoperable modelling overall, and to their organisation; listing particular models and datasets of interest; and identifying gaps. It was also stressed that the set of models needs to have a rapidly-running tier, even if that entails reduced temporal and spatial resolution and complexity. Also, there was encouragement to be realistic about expectations and to make progress on getting a system up and running.

Framework evaluation and selection.

The CSIP/OMS3 (Cloud Services Infrastructure Platform/ Object Modelling System) framework was selected for use in Stage 2 of the programme. This is an open-source framework developed by Colorado State University and used primarily by the USDA (United States Department of Agriculture) for agro-hydrology and soil conservation purposes. CSIP provides an infrastructure for running models through web services, which provides considerable flexibility in the way that models can be accessed via the world-wide web, and can harness web-based visualisation and data provision. OMS3 was trialled successfully in the Interoperable Freshwater Modelling project conducted in New Zealand. A downside associated with flexibility is the need to provide more developer support than might be required for alternative systems that have a richer set of pre-programmed functions. Specialist models from other frameworks could still be linked in to CSIP through loose coupling.

Models and datasets: high-level design

A diagram organising key datasets, model types, and outputs was developed to provide a long-term structure for framework development (Figure 5-1). The overall design is to have core modules addressing key biophysical and economic system components (for example, rainfall-runoff and farm production), with sets of higher-level models interacting with the core modules for tasks such as economic optimisation or parameter calibration. These models will be fed by key environmental and economic datasets. Within this overall design, there is flexibility to substitute and group model components and data sources; the design serves more as a general guide rather than a rigid structure.

Model selection and work plan

A set of models was selected to be implemented in the first two years, based on a number of factors such as previous use in New Zealand and addressing key modelling objectives. This list does not preclude individual organisations from adding further models to the framework or developing further integrated models using the framework; indeed, it is hoped that such broader adoption will occur (in Stage 3 of the programme, for example).

The particular models were grouped into three areas:

1. Steady state water quality (N) and production models, from farm to catchment scale.
    - Lookup tables, such as for the Canterbury matrix of good management
    - OVERSEER
    - CLUES / SPARROW stream routing
    - Simplified groundwater model from the Smart Aquifer Models programme
    - FarmMax
    - Linear spatial optimisation from LUMASS.
2. Dynamic water quality models
    - Irrigation demand component of LUCI
    - APSIM
    - Dynamic stream routing from Source
3. Water resources models
    - Irrigation demand component of LUCI (daily soil moisture balance and crops)
    - Abstractions and reservoir operation and stream routing from Source or similar

These models will be applied to a specific catchment where previous modelling efforts have occurred (e.g., Hurunui, Waituna, Ruamāhanga, or Selwyn).

A work plan has been devised which will incrementally implement the identified component models, with periodic re-evaluation of the next steps and the overall process. This will enable delivery of some demonstration integrated models, while managing risks and providing flexibility and agility.

Supporting datasets, standards, and visualisation tools have also been identified. Not all datasets are available in a standardised form, so some manual data preparation will also be required. Standards-based web services will be used where these are available.

The work programme will entail significant input from modellers and software developers. The overall estimated budget is $2,050,000 for Stage 2 of the project. It is expected that the Challenge will provide funding of $700,000, with the remainder to be provided by co-funding. Arranging this co-funding is a key task for the Governance Group.

The Governance Group will also need to address significant IP constraints related both to models and data provision. These constraints have been itemised in Section 8.4.

The proposal does not cover the generation of datasets or converting these to standards-based formats and provision through web services. Specific gaps in this area were identified. This is a critical area for the Challenge or other parties to address, and the Governance Group should seek reassurance that this is given attention.

A number of other risks were identified, with various mitigation measures. The staged approach and strong governance will help mitigate these risks, but there are some inevitable risks associated with adopting cutting-edge and rapidly-developing technologies in a complex multi-provider setting.

Overall, a reasoned, specific, and tractable programme has been proposed for Stage 2, and this should be approved with minimal delay to ensure that Challenge timelines are met.

1 Introduction

1.1 Programme background

This report has been prepared as part of the “Interoperable Modelling Systems for Integrated Land and Water Management” programme in the “Innovative and Resilient Land and Water Use” theme of the Our Land and Water National Science Challenge (OLW). The programme addresses Theme 2 of the Challenge by providing modelling tools to support “Innovative and resilient land and water use”, and Theme 3 by building collaborative capacity within the modelling and model-user communities.

The programme aims to develop an interoperable modelling system that is suitable for national use in integrated spatial assessment of environmental, production and economic implications of land use and land use change. The uses of the system will include assessment and accounting of productivity potential and water quality contaminant dynamics at farm and catchment scales. It is proposed that the availability of better, more trusted, and targeted modelling tools within an interoperable modelling framework will result in more effective use of integrated modelling for improved production and environmental management.

A staged approach has been developed to achieve these aims. Stage 1 of the programme, which is the subject of this report, establishes a proposal for work to be undertaken in Stages 2 and 3 of the programme. Stage 1 reviews previous interoperability efforts, defines an overall framework for New Zealand applications, and identifies an agreed initial set of models, data sources, and visualisation components, along with a software framework for linking and accessing the components. Stage 2, which will run from July 2017 to June 2019, will focus on fully implementing and demonstrating the initial set of models and data within the framework. Step 3 (for tranche 2 of OLW) proposes to enrich the range of models in the framework and demonstrate and evaluate the use of the framework in multiple contexts, including linking to social and cultural attributes.

The programme entails the establishment of a Governance Group working in tandem with a Technical Group, to ensure focus, relevance, course-correction and evaluation of the programme. The Governance Group includes representatives of partners or other key organisations and is responsible for steering the work, resolving IP and resourcing issues, and providing links to other stakeholders. The Technical Group incorporates specialists from a range of science providers, covering the required areas of technical expertise- including catchment hydrology, production systems, water quality, agro-economics and computer science. They were responsible for preparing this report. The members of both groups and the organisations they represent are listed in Appendix A.

Future stages of the programme will rely heavily on co-funding from programme partners, the Challenge itself providing only part-funding. Stage 1 of the programme builds the case for such funding, by evaluating modelling approaches and developing an implementation plan targeted to the objectives provided by the Governance Group, with an accompanying rationale. This report assists the programme partners to see where their models fit within a broader landscape of models and interoperable systems, to help build towards development of a business case for adopting a particular interoperability system and set of models.

1.2 Introduction to interoperability concepts

This section provides a brief introduction to interoperability, both to assist readers new to the topic and to identify key concepts and terminology. We also note that a previous New Zealand project, Framework for Interoperable Freshwater Models (FIFM), summarised framework technologies in a wiki and report[1].

Modelling of systems such as agro-environmental systems entails integrated modelling of several processes or elements, such as plant and animal production, soil water movement, leaching, runoff, water and contaminant transport and ecological effects, farm system economics and regional economics.

One approach to such modelling is to develop a tightly-coupled integrated model, where the model components (i.e., calculations for a part of the system), data files, and the user interface are knitted into closely interdependent code, perhaps as a single monolithic program. While such approaches can be targeted well to a particular problem, it can be difficult to apply the resulting models to other problems or reuse them as part of larger, more integrated applications. They can be inflexible in that it is difficult to modify or extend the model by adding new components, either for conceptual or technical (programming) reasons. A further restriction is that it may be difficult to take out a useful component for implementation within a different integrated model, because parts of the model code are so interwoven, including inter-dependencies with the user interface (for example, parameter input) and data sources.

At the other extreme, individual stand-alone models may be set up and run in a loosely-coupled manner. Models may be run on different computer types (platforms) by different specialist modelling groups or modellers, with input data gathered separately. The results of the different models may be collated once all the runs are complete, or, if there are dependencies, then models may be run in cascading fashion with transfer of output files from one model to the next. Examples of applications of this approach in New Zealand are considered later in this report. In some cases, this approach is suitable and effective, such as running an estuary receiving water body model after a catchment model has been run to obtain the estuarine inputs. However, the approach is difficult to manage, may increase costs, can be time-consuming and is difficult to apply when there are feedbacks between model components, and often needs to be set up in a customised or bespoke way for each model application. One way to address some of these problems is to use software for running the data collation, translation, model and output stages in an automated workflow; such an approach is attractive when a fixed set of models and data are required to be run repeatedly for a given study area, such as in operational flood forecasting.

In software engineering, it is common to avoid difficulties with monolithic or bespoke coupling by breaking the software into components (building blocks) which can be assembled and re-used in flexible ways. This involves separating out aspects such as the user interface from data, model engine, visualisation, and control or orchestration. Further, the models can be broken into model components. Such components could do small computational jobs (for example, scale a rainfall time series) or larger tasks (transfer water through a soil profile). There is no ideal level of componentisation. In principle, the task of a component would be clearly defined, such that it could

[1] https://teamwork.niwa.co.nz/display/IFM/Framework+for+Interoperable+Freshwater+Models

be substituted by a different algorithm – for example, changing the component that converts rainfall to runoff. Similarly, a model component could be modified from a base model, or new model components could be added to the chain.

Some kind of software is required to pull the components together, ensuring that the parts interface correctly and model components are run in a structured way, passing information appropriately. We refer to this infrastructure as the interoperability framework. Conceptually, different model components, datasets, user interface components can be linked into this framework, and set up to operate together in an orchestrated fashion.

Data and model coupling standards facilitate organised interoperability. Data standards can be used to specify how data describes itself (aspects such as the location, provider, variables, units) as well as specific formats and data structure (schema). Data standards are in common use in the geospatial, climate, and marine communities, and increasingly in hydrology. Standards for coupling models are relatively less mature, although they are evolving rapidly. For the world-wide web, there are some key standards for requesting and passing data and interacting with remote computers, although such web APIs (Application Programming Interface) may put limits on the size of data that can be transferred. However, in hydrology, agronomy, and economics, the standards are less developed.

The literature on interoperability and integration distinguishes between technical and conceptual aspects of integration. That is, it is possible to join models in ways that produce meaningless results. For example, it may be technical possible to join a runoff model producing hourly flow rates to a stream transport model which expects daily contaminant flux, but the results would be meaningless. More subtly, it can be difficult in practice to decide how best to combine models of different types – for example, groundwater models on a grid delineated for the aquifer to hydrology models based on sub-catchments and covering a larger area; or how groundwater rising to the ground surface affects a soil water model.

The literature on interoperability also addresses how data can be given meaning. In the simplest case, this might involve using a data dictionary for model variables, so that at the interface between models there is a common understanding of what a variable of a given name represents in reality. There are more sophisticated ways to describe the relationships between variables (for example hierarchical relationships) and some interoperability systems attempt to introduce such relationships in a formal way. A further enhancement is for data sources and models to describe themselves in standardised and accessible way, so that resources to solve a particular problem can be located, possibly in an automated and intelligent way.

Finally, we stress that an interoperable modelling system requires a supporting data infrastructure, with methods to obtain and pass data. Visualising data (showing data in maps, graphs, tables, summary reports, and animations), and interacting with users are also important.

1.3 Scope and purpose of this report

This report builds a proposal for Stage 2 of the programme, beginning with the modelling objectives and culminating in a specific proposal, mirroring the work activities in the programme to date.

Following the presentation of the methodology (Section 2), information on modelling objectives, as identified by the Governance Group, is presented in Section 3. The outcomes of a survey of members of the Governance Group are collated and summarised in Section 3 and the full survey results are given in Appendix B.

Reviews of selected interoperability frameworks and integrated modelling in the land-water space are presented in the appendices, with brief summaries in Section 4. The full reviews are given in Appendices C to P) These reviews emphasised New Zealand experiences to: inform selection of integration technologies; build justification for future integration efforts; and ensure that salient lessons for future modelling have been identified. We included integrated models, not just interoperability frameworks, to recognise past integration efforts which could provide useful information for further work.

Reviews of new and emerging technologies (Appendix Q) are summarised in Section 4.2.8, to identify whether there are new trends that could be of use for our interoperability programme.

Based on Technical Group workshops and subsequent discussions, Section 5 identifies, at a generic level, data and model types that would be included in a modelling framework.

Informed by the objectives, reviews, and generic model components, Section 6 assesses frameworks against a set of selection criteria, leading to selection of a framework for further development and accompanying rationale. Risks associated with the framework selection are also presented.

Section 7 identifies specific models and data sources that could be used to populate the framework, leading to selection of an initial set of models for implementation in a two-year timeframe, and associate datasets.

The plan for Stage 2 is presented in Section 8.

The report does not address development of data systems to support interoperability, as it was not in the programme scope. However, this development will be critical for success of the overall interoperability initiative. It does not address IP constraints or how the work will be funded, which is a matter for the Governance Group.

The programme's scope excludes the following: biocontrol, biodiversity and biosecurity (and by inference, the conservation estate); greenhouse gasses; urban environments; development of sensors and sensor networks; and generating data sources. In later discussions, it was agreed that urban environments will need to be considered to some extent in this programme when modelling mixed rural-urban catchments.

2 Methods

2.1 Governance Group and Technical Group

The programme entails the establishment of a Governance Group (members listed in Appendix A), to ensure focus, relevance, course-correction and evaluation of the work. The Governance Group will also address other critical matters such as resolution of intellectual property constraints, and is responsible for setting the objectives for the modelling work. The Governance Group works in tandem with the Technical Group, which involves specialists from a range of science providers, encompassing the required areas of technical expertise such as catchment hydrology, production systems, water quality, agro-economics and computer science. The Technical Group will make technical recommendations to the Governance Group and is responsible for planning and delivering the modelling framework and associated models.

2.2 Objective identification

The Governance Group was tasked with identifying objectives for the programme, within the broad boundaries established by the Challenge Directorate, to provide guidance to the Technical Group. This was achieved through two open and collaborative meetings, where participants were expected to not only represent their own organisations, but to consider the 'public good' component of the challenge's mandate. It was accepted that not all participants' needs would be met immediately, this is a complex programme that is expected to run over a number of years.

At the second Governance Group meeting, it was agreed to obtain more information on objectives from members individually. A written survey of needs was therefore conducted (Appendix B).

The objectives are summarised in Section 3.

2.3 Evaluation of past interoperable and integrated modelling efforts

The Technical Group reviewed a number of past attempts at building interoperable or integrated model frameworks (see Section 4 and Appendices C to Q for details). In addition to this, a review of emerging approaches and technologies was undertaken by the Technical Group. These reviews offered insight into which framework this project should use, the risks associated with this project, and how they could be mitigated.

A Technical Group workshop was held to identify the frameworks that should be reviewed, and the common criteria that they all should be assessed against. The criteria were derived from the modelling objectives set by the Governance Group. The frameworks were then allocated out to Technical Group members based on experience with the framework. In most cases, one person who was experienced with the framework or model was paired with someone who was less familiar with it to write the review. The pairings along with the reviewed frameworks and models are list in Section 4. All reviews were then peer reviewed by Paul Johnstone (Plant and Food), Iris Vogeler (AgResearch) and/or Sandy Elliott (NIWA) for consistency and to ensure that the assessment criteria were covered.

2.4 Framework software selection

Following the completion of the reviews, the Technical Group met at a workshop to evaluate which of the frameworks would be most suitable for use in this programme. This evaluation was based upon the objectives of the Governance Group, past experiences with interoperable and integrated modelling, and knowledge of emerging technologies. Each framework was summarised and presented to the Technical Group, and those with major limitations were excluded from contention.

The frameworks considered feasible were then evaluated against additional criteria by the Technical Group (shown in Section 6), leading to a shortlist of two top contenders. A sub-committee was formed to contact the developers (including a video conference with one developer) to address concerns or questions that had been identified at the workshop, and provide feedback to the Technical Group. A leading candidate was then agreed upon by the Technical Group.

2.5 Determining model components and data needs

At the second Technical Group workshop, the model component and data needs were determined based upon the Governance Groups objectives. These were drafted into a diagram (upon which Diagram 5.1 is based), and the model capabilities identified were used to start a discussion regarding which specific models should be included in the programme.

The initial shortlist (see Section 7) was identified as being too ambitious for Stage 2 of the programme, and was further narrowed to ensure feasibility whilst still demonstrating a strong value proposition to stakeholders at the end of the two-year cycle. This was achieved by a sub-committee and an initial proposed set of models was canvassed with the Technical Group.

An additional subcommittee was formed to assess approaches for incorporating economic linkages and optimisation- their feedback has been incorporated into this report. Further information on ways to use of OVERSEER and APSIM in interoperable models was provided.

Finally, a specific proposal for Stage 2 was developed, built around the needs identified, framework selected, and the data and model requirements. An indicative timeline and estimation of resourcing requirements was also established.

3 Modelling objectives

A key component of designing an interoperable model system is establishing the high-level objectives. In this section, programme requirements from the initial programme proposal, an early Governance Group meeting, and a survey of Governance Group members is presented. These modelling objectives were used to guide framework evaluation, designing the set of models and data at generic level, and selecting an initial set of models to implement. This section presents the objectives, while later sections use the objectives for evaluating frameworks, selecting model components at generic level, and selecting an initial set of models for implementation in the framework.

3.1 Programme requirements

The overall mission of the OLW Challenge is "*To enhance primary sector production and productivity while maintaining and improving our land and water quality for future generations*". So, the interoperable modelling programme should address production, productivity and water quality implications of land use. The productivity aspect implies economic or resource utilisation efficiency assessment at some level.

While this programme is related to others in the Challenge, the purpose is to serve the broader goals of the Challenge, rather than to service other programmes. For example, the purpose is not to provide models to support the Sources and Flows programme in Theme 2 of the Challenge, although there may be mutually beneficial model development in both programmes.

The programme proposal established some high-level objectives for the interoperable modelling system, as summarised below. Some of these are aspirational or for the ultimate system, rather than expectations to be achieved immediately.

Fundamentally, the system should:

- Draw on national datasets.
- Have national applicability.
- Be suitable for integrated and spatial assessment of economic, production and environmental implications of land use and land use change.
- Be suitable for assessment and accounting of productivity potential and water quality contaminant dynamics at farm and catchment scales.
- Make use of data repositories accessed via standards-based (e.g., Open Geospatial Consortium (OGC) compliant) interfaces.
- Use an open source software system for model interoperability.
- Adopt or build upon international frameworks, standards and component libraries.
- Allow extension to include optimisation and trade-off analysis that takes into account scenarios, the worth of data and its predicted uncertainty.
- Have a strong alignment with modelling efforts in projects aligned to and within OLW, for example, integrated surface-groundwater modelling.

- Have applicability to a range of production enterprise types, spatial and temporal scales.
- Make use of existing models or model components and visualisation components, but also allows for incorporation of new models as they emerge.
- Incorporate state-of-the-art information visualisation capabilities to enable end-user-targeted and easily understandable summaries, visualisations, interactive info-graphics, and animations of modelling results.
- Be suitable for both complex spatio-temporal simulation models and simplified models (developed as approximate representation of more complex models, or from a separate basis).

Ideally, the system would eventually transition to a web-based infrastructure - initially for data supply but ultimately extending to visualisation - running models on distributed computers and multiple platforms, and model orchestration.

The programme will exclude: biocontrol, biodiversity and biosecurity (and by inference the conservation estate); greenhouse gasses; urban environments; generating data sources, and development of sensors and sensor networks. The exception to this would be where these must be modelled to adequately represent catchment processes (for example, some representation of urban areas in a mixed urban-rural catchment).

3.2 Initial Governance Group requirement

The first Governance Group meeting addressed objectives for the programme. The summary points recorded in the minutes were:

- *"We want a system linking land and water at a catchment scale to allow farmers to predict effects of their decisions at a catchment scale. Linking to farm scale would enable catchment-scale benefits of farm-scale actions to be evaluated. Time-steps would ideally be daily, although monthly/seasonal may be more practical".*
- *"Biophysical modelling needs encompasses more than just nutrient flow; water demand and quantity is also a major issue*".

During the second Governance Group meeting, it was decided that economics aspects (for example, farm profit and regional economics) were of secondary importance compared with biophysical and production aspects, at least for the initial phase of the project.

3.3 Governance Group survey

At the second Governance Group meeting, it was realised that deeper consideration should be given to the objectives, and that members should be polled on an individual basis. Consequently, Governance Group members were asked to respond to an online written survey which asked for high-level objectives and also drilled down into some of the specifics of models, supporting datasets, and the survey also asked for gaps and general feedback.

The raw results from the survey, which also incorporates earlier feedback about objectives, are presented in Appendix B. Not all members of the Governance Group managed to respond to the survey. In some cases, earlier advice from some of the parties were incorporated into the survey format, if the parties had not responded directly to the survey. The responses were collated and are summarised below (with some paraphrasing).

What questions do you see the interoperable model system addressing?

- The programme should provide a recognised national system to effectively link existing land-environment models. Common base datasets should be utilised in a streamlined and effective manner, for the purposes of:
    - Assessing environmental risks such as water quality from land use and management scenarios.
    - Limit setting under the NPS-FM.
    - Assessing the implications of policy, resource constraints, technology, and environmental mitigation, on production, environmental quality, and wider values such as ecosystem services.
    - Identifying options for moving to higher value use of land within environmental limits.
    - Flow and contaminant accounting and allocation.

- The system should be suitable for use by stakeholders involved in limit setting, and for enterprises in commercial environments.
- Provision of a seamless linkage from farm system models to catchment models.
- Financial data should be included.

What benefits would a successful interoperable model system bring to your organisation?

- Ability to co-ordinate modelling investment with others.
- Establish plausible and recognized basis for resource accounting.
- Provide agility and speed for regional council scenario assessment with communities.
- Efficiency in bringing models and datasets together to model particular issues.
- More efficient use of stakeholder resources and provision of robust evidence for decision making (e.g., limit setting processes).
- Confidence that models that impact farming businesses are drawing on all information and linking this information together to get robust results.
- Enable more complex modelling to address complex policy at multiple temporal and spatial scales to address social, economic and environmental issues.
- Allow contention in limit setting to focus on values rather than disputing science.
- Efficient allocation of water quantity.

What are key things that the system should be able to predict, and at what spatial and temporal scales?

- Impact of predominant land uses on water quality, including cost and effect of mitigations.
- Flows in response to water allocation.
- Farm and catchment economics, including employment.
- Range of spatial scales, linked from local to catchment to national scale, and a seamless linkage from farm to catchment scale.
- Two tiers of timescale: annual (or seasonal) and daily.
- Both loads and concentrations of nutrients.

Are there any particular models you would like to see catered for by the framework?

Note: many of these models are discussed further in Section 4 and Section 7.

- NZFarm
- SedNetNZ
- FARMAX[2]
- Source
- SPASMO
- FEFLOW
- CLUES
- LUCI (link with other frameworks using standards)
- LUMASS
- TopNet
- APSIM
- OVERSEER
- MODFLOW
- MyLand
- Urban catchment models

Key supporting datasets the framework should utilise, especially in your area of interest

Note: data sources are discussed further in Section 7.2.

- S-map soil properties
- Land use and title including fertilizer use, stocking rate
- Agricultural production survey data
- Daily climate data from the virtual climate station network (VCSN)
- Drought index data
- LCDB
- Other LRIS spatial data (now available from Koordinates)
- Agricultural Census data
- Climate data
- Climate change scenarios

[2] Note that FARMAX is developed and distributed by Farmax. In this document

- River network
- Root flux meter observations
- Council SOE observations
- Topography
- Sediment discharge data
- Flow monitoring observations

Are there any key gaps in models, data, or interoperability that need to be addressed?

- Getting hydrology models to work across scales.
- Make models and data compliant with interoperability standards.
- Groundwater lag time models.
- Farm systems.
- Coupling to forest models.
- Knowledge of timescale of environmental response for farmers.
- Harmonisation of spatial units (e.g., 'blocks') across models.
- Bacterial transport models.
- Need agreements for sharing private and public data.
- Range of models for appropriate complexity, detail, and uncertainty.
- Ki uta ki tai (mountains to sea) integration.
- Process for testing data quality.
- Availability of a single source of data.
- Difficulty getting information on farm performance.
- Contaminant attenuation models, and implications of this for farmers.
- In-stream processing models.
- Tactical farm management.
- Spatial data at fine scales.
- Testing regimes for models.
- Integration of consents and compliance data.
- Linked urban and rural models for peri-urban areas.
- Biophysical models that can be run quickly.
- Land use and practice/treatment data.
- Actual data from cropping systems.
- Methods for assessing uncertainty propagation through models and system uncertainty.

Do you have any further comments or feedback you would like to pass on?

- Resolution of IP and licensing will be critical.
- Concerned about exclusion of urban land from the project scope. Leaves a gap in terms of model use for councils in mixed and urban-dominated catchments.
- Need to see real progress in interoperability- i.e., getting acceptable numbers and graphics out.
- Need to be realistic about what can be achieved in terms of spatial and temporal scale and resolution.

- Concerned about continuing the system beyond the timeframe of the project. Such frameworks require long-term funding to keep them current and useful.

3.4 Summary of objectives from the programme and Governance Group

Objectives provided by the Governance Group confirmed the original intentions of the programme, with some changes in emphasis.

It was considered important to link between the farm and catchment scales to maximise uptake of management measures by enterprises.

Water quantity, and the response to water allocation measures, was highlighted specifically as part of the environmental impacts.

Generally, there was interest in using the interoperability system for contaminant and water accounting, not just limit setting or optimisation.

Economics was given lower weight than originally anticipated, although maximising the value of land was still important.

It was also stressed that models need to have a rapidly-running tier (i.e., level of operation, or option), even if at the expense of lower temporal and spatial resolution and complexity, while still allowing for detailed complex models.

A number of specific models and data sources were identified, which will help direct the system development.

Several gaps and challenges were identified, with encouragement to be realistic about expectations and to make progress on getting a system up and running.

4 Reviews of interoperability frameworks and integrated models

4.1 Interoperability frameworks

Reviews of the interoperability frameworks, listed in Table 4-1, that were undertaken by the Technical Group are contained in the Appendices C to I. In this section, we provide a summary of key points.

Table 4-1: List of interoperability frameworks reviewed.

Framework	Primary Author/s	Other contributor or primary reviewer[1]
Bespoke loose coupling	Chris Daughney (GNS Science) Zara Rawlinson (GNS Science) Mike Toews (GNS Science) Catherine Moore (GNS Science)	Lawrence Kees (Environment Southland) Rachael Millar (Environment Southland) Tim Ellis (Environment Southland) Brent King (Greater Wellington Regional Council) Natasha Tomic (Greater Wellington Regional Council)
Delta Shell	Hans Eikaas (DairyNZ)	Alex Herzig (Landcare Research)
eWater Source	Nic Conland (HortNZ)	Sandy Elliott (NIWA)
GeoJModelBuilder	Alex Herzig (Landcare Research)	Sandy Elliott (NIWA)
Land Use Management Support System (LUMASS)	Alex Herzig (Landcare Research)	Gabriella Turek
OMS3 (Object Modelling System) and CSIP (Cloud Services Innovation Programme) (CSIP/OMS3)	Gabriella Turek (NIWA)	Sandy Elliott (NIWA)
OpenMI and Pipistrelle	Gabriella Turek (NIWA)	Sandy Elliott (NIWA)

Note[1]: Final reviews were provided by Iris Vogeler (AgResearch) and Paul Johnstone (Plant and Food Research).

4.1.1 Bespoke loose coupling

This review presented an approach whereby stand-alone models are linked in a sequential loosely-coupled fashion on a case-by-case basis, along with experiences from some recent New Zealand applications. Such an approach may involve running of models by separate parties, and development of custom code for data translation. More information on bespoke lose coupling can be found in Appendix C.

An advantage of this approach is that it enables creative and flexible linking, it leaves specialist models intact, and it leaves experts from different institutions to focus on their own models.

Example applications of this approach (in integrated surface-groundwater modelling Southland surface-groundwater modelling and as part of limit-setting in the Ruamāhanga catchment) showed that linking complex environmental models involves conceptual complications (e.g., translating between model element types and areal coverage) and is time-consuming with considerable up-front planning required. In some cases, glue code needs to be written, for example to translate output from one model to be compatible with input to another model. There were useful lessons about the general process of coupling, including conceptual development. At a purely technical level, software tools are improving that help developers create usable application programming interfaces, which may themselves assist in developing loosely coupled systems. Software containers are also a useful tool for building loosely-coupled systems, as discussed below in the emerging technologies section.

The approach was considered more as a reference approach, to compare with other interoperability approaches, rather than promoting the approach for this programme, and also to highlight lessons that may be helpful for future work.

4.1.2 Delta Shell

Delta Shell (Appendix D) is a model coupling framework developed by Deltares, a research and consultancy organisation located in the Netherlands. The framework is based on Microsoft .Net C# programming language. It includes a graphical user interface with mapping, project management, and time series and table displays. It includes a scripting functionality to enable workflows (e.g., data import, manipulation, model runs and display) to be automated. Some geospatial libraries are already linked into the Delta Shell. Calibration libraries can be applied to models set up in Delta Shell. The user can develop plug-ins to add models or other components (for example a different rainfall-runoff model).

The software is open source under GNU Lesser General Public Licence (LGPL) libraries, and most of the plug-ins (which can also be referred to as add-ins or extensions) are available under more liberal GNU General Public Licences (GPLs). However, currently access to the code needs to be arranged with Deltares, as their open source repository system for external users is not mature.

Key models that have been implemented in Delta Shell focus on dynamic hydrodynamic and water quality simulations, reflecting the core interests of the host organisation.

Delta Shell is being used by DairyNZ (plug-in development) and NIWA (hydrodynamic and water quality model users, trialling development of plug-ins for periphyton growth).

Overall, Delta Shell is a rich and powerful framework. But it uses custom libraries for interoperability, and it is likely that plug-ins would have to be developed to meet the needs of the OLW programme.

4.1.3 eWater Source

This framework (Appendix E) was developed by eWater in Australia to serve as a National Hydrological Modelling Platform. It is based on an underlying interoperability system TIME (The Invisible Modelling Environment) developed by CSIRO, which uses custom interoperability methods and libraries built in in Microsoft .Net. languages.

Source is based on a node-link conceptualisation of catchments, with a number of 'functional units' representing area with similar behaviour within each sub-catchment. Simulations are usually performed with a daily time-step.

A modular approach with plug-ins is used. For example, different rainfall-runoff models can be inserted. Users can also provide their own plug-ins to complement the default set of models.

Source models can be run through the rich GUI, or from a command line, or through small programmes in .Net or Python, including possibilities for web access for running models and visualising results.

Several applications for Source have been undertaken in New Zealand, including adding a plug-in for soil moisture balance under irrigation and groundwater routing. However, those plug-ins are not open-source or available freely, and were provided through a small number of consultants.

Access to the source code and advanced modelling features requires being a member of the Hydrological Modelling Platform initiative, which is funded through member contributions.

Overall, Source has some powerful capabilities, but its primary emphasis on hydrology, the node-link spatial basis, and limited openness may be restrictive for the OLW programme.

4.1.4 GeoJModelBuilder

This framework (Appendix F) represents an example of recent web services approach for environmental models. In such approaches, the web is used as the basis for accessing and orchestrating models and data, with model components and data provision able to be provided on remote services.

GeoJModelBuilder takes a workflow approach using open interoperability standards- such as OGC Web Processing Service (WPS), Sensor Web, and OpenMI (1.4) standards. It offers some interesting possibilities for triggering models in response to sensor events, although this is not an objective of the OLW programme. The software itself is open source and based on the Java programming language.

An example application has implemented geoprocessing of spatial data using the open-source Geographic Resources Analysis Support System (GRASS) GIS libraries, and a rainfall-runoff model. There has been little other use, as this software has only recently been developed (published in 2015).

The model is supported by Wuhan University. The user manual is current only available in Chinese, pointing to potential language difficulties if it were to be adopted.

Overall, this system meets several needs of the project, and the flexible open-source standards-based approach is attractive. However, the nature of the support and the immaturity introduce risks, limiting the applicability to the current project. Despite this, we do expect such approaches to develop further in the future, as web services approaches to model interoperability are an active area of advancement. Web services and geoprocessing as the basis for interoperability are discussed further in other sections of this report.

4.1.5 Land Use Management Support System (LUMASS)

LUMASS (Appendix G) is a spatial system dynamics framework including spatial optimisation, developed by Landcare Research. LUMASS is a mostly set up for raster-based (grid based) system dynamics models, with small model components (for example, one equation) and its associated data set up in a workflow. It uses as a custom API (interface), and can be set up on a high-performance computer, when additional computational performance is required. The source code for LUMASS is open-source, and LUMASS uses open-source spatial processing and optimisation libraries. Rasters and tables can be displayed, Graphical User Interface (GUI) based desktop applications can be built, and applications can be run from the operating system command line.

LUMASS has been used for modelling in New Zealand, including the erosion model SedNetNZ, a forest growth model, and image processing.

LUMASS provides powerful technical features for the OLW programme, such as spatial optimisation, powerful raster libraries, and open-source approaches. However, it would need to be adapted to use interoperability standards, and the user and developer base would need to be expanded beyond the current single developer.

4.1.6 OMS (Object Modelling System) and CSIP (Cloud Services Integration Platform)

OMS3 is an interoperability platform developed by Colorado State University, mainly for USDA agro-hydrological and soil conservation applications. CSIP is a system which enables OMS3 models, data provision, and other services to be run via the internet, thus delivering models to a wider range of users with a polished user interface. OMS3 and CSIP are reviewed in Appendix H.

OMS3 uses the Java language which can be deployed on most computing platforms, and a custom annotation-based approach whereby small pieces of code can be inserted into programmes to help build an interface. The GRASS geospatial library has been set up as OMS3 components, and visualisation components (e.g., maps) are also available. OMS3, CSIP, and a library of components are open source.

Several hydrological models have been set up within OMS, and several models utilising over 200 services have been set up with CSIP. Some of these models are complex – for example, a more spatially-explicit and componentised version of the catchment model SWAT (Soil and Water Assessment Tool).

OMS was tested for the interoperable freshwater models project[3]. It was found to be promising, allowing for a range of model types. However, that project did not attempt to set up complex sets of models. It was also found that programming expertise is required to adapt models into the framework and to set up coupling between models. There is little uptake of OMS3 outside the core US government agencies funding development, excepting the University of Trento in Italy.

The open nature of OMS3 along with its agro-hydrology genesis, strong government support, and linking to web delivery of models are attractive for the OLW programme. However, the core

[3] https://teamwork.niwa.co.nz/display/IFM/Framework+for+Interoperable+Freshwater+Models

interoperability methods are not standardised, and there is a limited community outside the core institutions which may provide some challenges for use in the OLW programme.

4.1.7 OpenMI and Pipistrelle

OpenMI (Appendix I) is a standard for model coupling, initially developed to allow EU research organisations to link dynamic water and agricultural models. The standard has been implemented in a software development kit and GUI-based environment (Pipistrelle), which is open source.

This standard has been approved by OGC (Open Geospatial Consortium) with some international uptake. However, there was a major revision (2010) which introduced additional complexity and was not backward compatible. As a result, the older standard is often used. The future of standard is uncertain, with limited support and seemingly little uptake within the original community. Even so, it is one of the few model coupling standards available.

4.2 Integrated models

Some models are already being used in New Zealand for integrated modelling. We review these in this section to identify whether they provide suitable guidance for future modelling. Also, some of the integrated models have been implemented in a broader framework, so they are relevant to framework evaluation and selection. Note that in Section 7, we consider using some of the component models from these integrated models for specific tasks, such as routing.

The reviews of selected integrated models that are currently used in New Zealand (Table 4-2) are included in Appendices J to P. These were included in this study because they highlight some key directions for integrated modelling and interoperability frameworks in New Zealand. Similarly, some of the integrated models below are built in a modular extensible way, and are in a sense frameworks. In addition to these models, new or emerging technologies are also reviewed (Appendix Q, Section 4.2.8).

4.2.1 CLUES

CLUES Appendix J is a GIS-based integrated model to predict annual average instream loading, nutrient concentrations, estuarine potential concentrations, sediment and microbial loads, and socio-economic indicators on the national stream network (River Environment Classification version 2, REC2). It utilises national datasets (e.g., soil drainage, rainfall, land use), and a simplified version of OVERSEER for nutrient generation from pasture. It is calibrated to measured loads from national observations dataset. An estuarine component has been added to CLUES. It is free to use but closed source, and relies on proprietary ArcGIS. There are a number NZ applications, including some national applications, and some modifications and extensions for local needs (for example, extensions for spatial refinement, lookup tables). It continues to be maintained by NIWA.

CLUES is not suitable as a framework for this programme due to the limited range of models and use of proprietary GIS software. But it does illustrate the use of OVERSEER at national scale, coupling with national datasets, predictions for multiple contaminants and environments, and coupling with economic indicators. Many of these features and components could be incorporated into the OLW framework.

Table 4-2: List of New Zealand integrated models reviewed.

Integrated model	Primary Author/s	Other contributor or primary reviewer[1]
Catchment Land Use for Environmental Sustainability (CLUES)	Sandy Elliott (NIWA)	Bethanna Jackson (Victoria University)
Land Application and Management (LAM)	Graeme Doole (University of Waikato)	Juan Monge (Scion)
Land Utilisation and Capability Indicator(LUCI)	Bethanna Jackson (Victoria University)	Sandy Elliott (NIWA)
MyLand	Steve Wakelin and Wayne Schou (Scion)	Juan Monge (Scion)
NZFarm	Tarek Soliman (Landcare Research)	Juan Monge (Scion)
OVERSEER Integration	Iris Vogeler (AgResearch)	
WISE (Waikato Integrated Scenario Explorer) application of Geonamica	Bethanna Jackson (Victoria University)	Sandy Elliott (NIWA)

[1]Overall reviews were provided by Iris Vogeler (AgResearch) and Paul Johnstone (Plant and Food Research).

4.2.2 LAM

LAM (Appendix K) is an approach for assessing land use activities for economic, production and environmental outcomes. It can be used for optimisation under environmental constraints. It incorporates information from models such as OVERSEER and FARMAX, and has been coupled with simple biophysical models (e.g., SedNetNZ and the SPARROW component of CLUES). LAM is generally implemented in the specialist economics software GAM.

There have been multiple applications of this approach in NZ and Australia, including for limit setting processes. Generally, an application is custom-made for the particular problem.

Concepts and techniques from LAM will be useful for this programme, but the reliance on GAM, simplified representation of environmental effects, limited developer base, and lack of use of standards or GUI means that the model will not be directly applicable in the OLW programme.

4.2.3 LUCI

LUCI (Appendix L) is a GIS-based approach for integrated assessment of land activities on environmental and other outcomes. It is largely raster-based (with arbitrary resolution) and mostly uses non-dynamic models for rapid run-time, although the hydrodynamic component can be dynamic.

LUCI can use a rule-based spatial overlay approach for identifying trade-offs and good locations for landscape protection and mitigations, but optimisation algorithms have also been applied for spatial allocation.

Some concepts and components from LUCI are likely to be useful for incorporation into the OLW framework, but the lack of standards and use of proprietary GIS mean that it is not suitable as a framework for OLW.

4.2.4 MyLand

MyLand (Appendix M) is a web-based strategic land use planning tool designed to assist land owners to improve the long-term profitability and sustainability of their land management. It is intended to simulate production, economic and environmental attributes of a range of land uses over the long term (i.e., one or more forestry rotations). It was developed by Scion, with emphasis on forestry but can encompass other land uses.

It utilises Microsoft Silverlight web technologies which are being phased out- so shelf-life is limited. Furthermore, it is closed source, so is not suitable as a framework for the current project. The underlying models are probably useful for this programme- especially for the forestry sector.

4.2.5 NZFARM

The Landcare model NZFARM (Appendix N) is a modelling approach that shows how changes in environmental policy could affect land use and its subsequent spill-over effects on a group of economic & environmental performance indicators. It maximises the net revenue from agricultural/forestry production subject to feasible land-use and land-management options. The economic returns are obtained by integrating several sources of farm budgets for relevant enterprises, while environmental impacts are obtained from existing biophysical models such as SedNetNZ, NZEEM, CLUES, OVERSEER, WATYIELD, NZ GHG Inventory algorithms, and CENW. NZFarm is implemented in the proprietary optimization software GAMS.

There are multiple applications of NZFarm at catchment and national scale.

Several concepts and models within NZFARM are likely to be useful for the OLW programme. However, general reliance on a comparative static approach and GAMS software and the need to incorporate simplified biophysical models into GAMS means that NZFARM is not suitable as a framework for the OLW model.

4.2.6 OVERSEER Integration

OVERSEER® Nutrient Budgets (which we refer to as OVERSEER, Appendix O) is a decision support system (DST) farm model that has been developed in New Zealand as an industry standard for recommending nutrient inputs and estimating long term average nutrient losses to water. We reviewed how OVERSEER has been integrated into other models, because this is likely to be an important component model in the OLW framework, and past integration efforts could provide guidance on how to incorporate OVERSEER into the OLW framework.

OVERSEER has an API which enable calls to the internal functions of another program. Input and output files are in XML format with an associated data schema.

Various institutions have built tools for running multiple OVERSEER files based on a base scenario and variation in certain parameters (pers. com. Mike Rollo, AgResearch; Hemda Levy, Dairy NZ), with compilation of the output data into a spreadsheet.

OVERSEER has been linked with various different models. Examples include:

- CLUES (see 4.2.1 and Appendix J). OVERSEER was provided as a DLL which internally uses default model settings and enables key parameters such as stocking rate, rainfall, and soil to be modified.
- Outputs from OVERSEER have been linked with GIS data within the MitAgator which distributes OVERSEER sources within a farm to identify critical source areas of contaminant generation and locations for mitigation.
- OVERSEER has been "linked" with FARMAX via manual transfer of data/information.

Further examples on the linkage of OVERSEER with other models include LUCI, MyLand and NZFARM, all of which are described in separate reviews within this report.

At present, coupling of OVERSEER into other software is subject to a licence fee. This may be a barrier for adoption into the interoperability framework.

OVERSEER was set up as a web service and coupled into OMS3 in an exploratory exercise in the Interoperable Freshwater Models project (Elliott et al., 2014), altering an example input (rainfall) through modification of the XML input file and reading the results.

Full coupling of OVERSEER within a system by which individual models exchange input and output files and define their parameters through a joint calibration/inversion procedure is probably not feasible, partly due to licence issues. Other constraints such as temporal scale of the models, with OVERSEER mainly providing annual outputs, might also hinder full coupling. However, these integration examples point to good opportunities and prior experience with linking OVERSEER into other models, either through direct coupling and manipulation of a limited set of model inputs, or through approaches such as lookup tables. Integration of OVERSEER in the current programme is considered further in Section 7.1.3.

4.2.7 WISE application of Geonamica

Geonamica is a dynamic land use evolution programme, largely based on cellular automata approaches (a grid-based simulation approach whereby grid cells evolve according to rules and interactions with neighbouring cells). It has a powerful set of simulation, coupling, and visualisation components.

Models must be adapted by the Geonamica developers (often rewritten) to work in the system, and the software is proprietary and closed source.

The WISE model (Waikato Integrated Scenario Explorer) was a key New Zealand application of Geonamica (Appendix P), bringing together land use evolution, population, economic, and environmental models.

Geonamica not suitable for use in this programme due to its closed and proprietary nature, and general cellular automata focus. However, some of the components of WISE may be suitable for incorporation.

4.2.8 New and emerging technologies

In this section, we describe a number of software engineering approaches that are gaining traction, either in terms of delivery of modelling services, or in terms of technologies that help develop systems that need to run models, such as particular software platforms that sets of models might run on. Further details are available in Appendix Q.

Key new and emerging technologies include:

- Software container technology—*e.g.*, Docker, *etc*. Containers are lightweight virtual machines that isolate computing and data with a lower performance overhead than previous virtualisation approaches. Containers are an ideal way to package software for development and production, as they can include software dependencies (*e.g.*, libraries, *etc.*) in a convenient way. Containers are increasingly used as a convenient unit within cloud computing, due to their assistance in componentising the application-specific (*e.g.*, models, in this case) and operating system software.
- Serverless computing—*e.g.*, Amazon Web Services' Lambda product. This paradigm aims to allow developers to ignore any aspects of managing underlying computing infrastructure (such as servers). Software code (*e.g.*, models in this case) can be deployed directly into infrastructure such as an appropriate cloud offering. The serverless computing framework will automatically optimise how data flows in and out of this code. Serverless computing platforms often operate as stream-processing systems, that react immediately to the arrival of new input data, and present their results as dynamic streams of output data.
- Unikernels. These can be viewed as an evolution of software containers that compile down the user code and operating system into a customised operating system that can only support that particular user code. This minimises the size of the software container, reduces its resource needs, and provides security benefits. Unikernels may become a key building block in serverless computing.
- Virtual Laboratories—*e.g.*, Australia's Nectar offerings. These virtual laboratories provide a convenient, web-based interface to underlying models and data, alongside visualisation of results. There is an increasing trend for High Performance Computing facilities to be coupled with virtual laboratory platforms (*e.g.*, the New Zealand eScience Infrastructure platform), to facilitate easier access to supercomputers, by reducing the need for specific technical training in how to operate such HPC platforms.

Beside the importance of the technical components of the interoperable model and data system, social and community aspects need to be considered as well to create "A dynamic web of models, integrated with databases and websites, to form a consultative infrastructure where researchers, manager, policy-makers, and the general public can go to gain insight into 'what if' questions" (Nativi et al. 2013). Nativi et al. understand the "Model Web" as a generic system, which facilitates access to and interaction between models. The desire to update the world-wide web to provide capabilities for orchestrating interactions of web-based software systems is not new: the semantic web[4] has been a goal for over a decade, and examples of various sorts are available today. However, in some fields it seems that the trend toward open access to "big data" is helping reduce fragmentation in standards and interacting software systems. In the context of the OLW interoperability programme, the Nativi

[4] https://www.w3.org/RDF/Metalog/docs/sw-easy

et al.'s vision of the Model Web provides useful guidance in terms of key characteristics that need to be considered to achieve the long-term objectives of this programme (i.e., the ten-year vision).

An emerging technology which should be considered for future proofing of the adopted interoperability framework is the Internet of Things (IoT, see Appendix R for more detail). The Oxford dictionary defines the IoT as "*The interconnection via the Internet of computing devices embedded in everyday objects, enabling them to send and receive data*". In the development of the interoperability framework, consideration should be given to its compatibility with the on-the-fly data management typical of distributed stream processing, and thus to be better positioned to interact with the IoT in future.

In the context of the OLW interoperability programme, the Nativi et al.'s vision of the Model Web provides useful guidance in terms of key characteristics that need to be considered to achieve the long-term objectives of this programme (i.e., the ten-year vision).

5 Generic model components and data

A diagram of generic model components, data and outputs is shown in Figure 5-1. This was based on information provided by the Technical Group in response to the modelling objectives, with some subsequent re-organisation, classification, and refinement. By generic components, we mean that specific data sources, models or output variables are not mentioned. Some further information on output variables is provided in Appendix S.

While it would be desirable in principle to show how models can be linked to different data sources, and to each other along with the exchange items, this would make an unwieldy diagram. Linkages for the initial subset of models (Stage 2 of the project) are identified later in the report.

The list of models involves some conceptual separation between model types, which is somewhat arbitrary, but reflects logical separation as viewed by the Technical Group. In reality, there may be cases where a single available model performs multiple aspects of this list, or where further subdivision is appropriate. For example, an existing crop growth model may also estimate leaching, and contaminant generation may involve separate sub-models for leaching and overland flow.

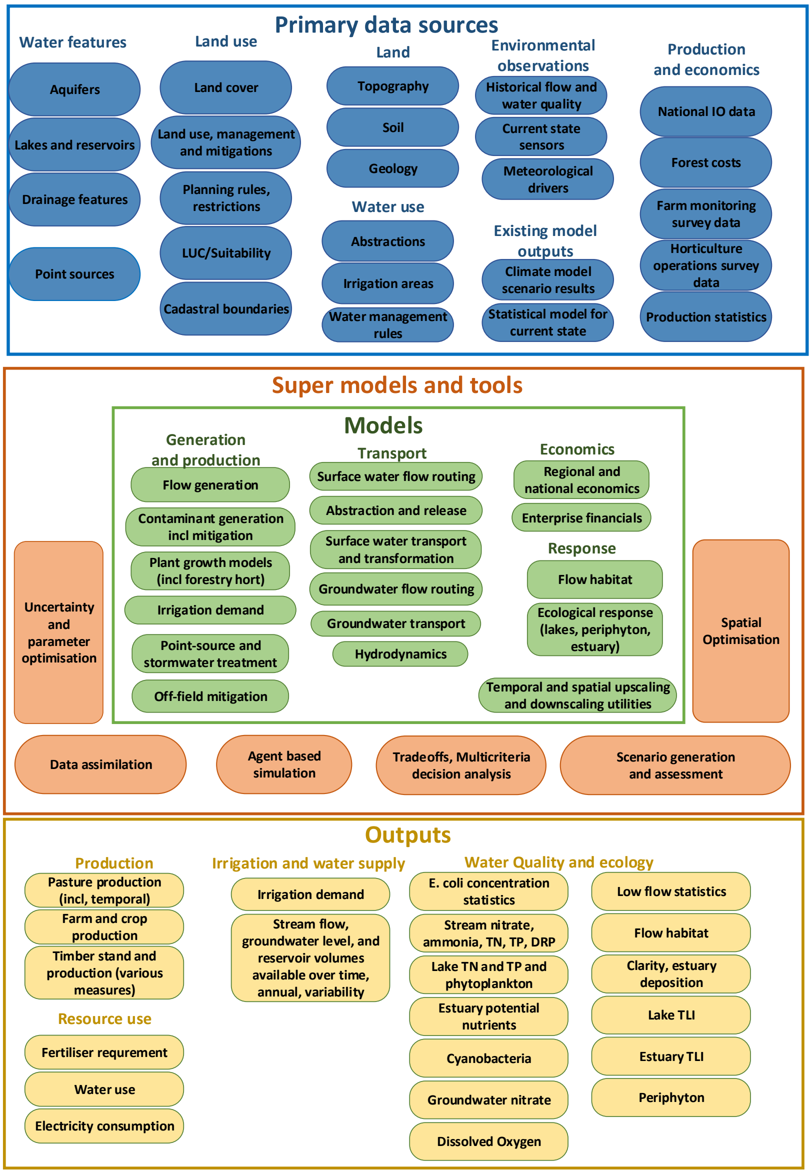

Figure 5-1: Generic data types, models and outputs.

6 Framework evaluation and selection

One of the primary objectives of this project is to recommend a model interoperability framework to be tested and further developed within Stage 2 of this programme, within the context of the OLW Challenge. This matter was explored by the members of the Technical Group in a two-day workshop held in Wellington from 6-7 June 2017.

6.1 Framework evaluation

The decision of which model interoperability framework(s) to test in Stage 2 must be transparent, scientifically sound and publicly defensible. To achieve this, the Technical Group members defined 18 criteria, and their rationale, to facilitate the selection of the model interoperability framework (Table 6-1). This list draws on modelling objectives as identified by the Governance Group, and added criteria where this was considered appropriate to address technical considerations. Most of the criteria pertain to general usability of the model interoperability framework, e.g., supporting ease of interpretation of outputs (Criteria 1-2), facilitating uptake by end-users and expert developers (Criteria 3-7), ensuring flexibility of use (Criteria 8-9), and enhancing robustness and reducing modelling run times (Criteria 10-15). The Technical Group developed consensus scores of 'yes', 'no', 'partial' or 'maybe' for Criteria 1-15.

The final three criteria are used to provide differentiation amongst otherwise similarly ranked interoperability frameworks. Criterion 16 reflects the Technical Group's assessment of the overall effort required to apply the interoperability framework. For example, some effort may be required to adapt a component model so that it can be interfaced with the interoperability framework. Criterion 16 therefore represents the 'barrier to entry' for use of a particular interoperability framework. Criterion 17 reflects the level of transferability inherent to the interoperability framework. For example, some frameworks may make it quite simple to re-use datasets and swap component models from one modelling exercise to another. This criterion therefore encodes 'barrier to re-use'. Criterion 18 reflects the overall functionality of the interoperability framework, for example in terms of the different types of component models that can be incorporated or the spatial or temporal scales on which they can operate. The Technical Group developed consensus scores of 'high', 'medium' or 'low' for Criteria 16-18. The ideal interoperability framework would have low effort to develop, high transferability and high functionality; however, the Technical Group considered the trade-offs between these three criteria (e.g., high effort to develop but high transferability vs. low effort to develop but low transferability).

Criteria 1-18 were applied to each of the six interoperability frameworks considered in this project (see Section 4 and Table 4-1). As noted previously, there is no intent to further explore the bespoke loose coupling approach in Stage 2 of this programme. Rather, the bespoke loose coupling approach is largely used here as a benchmark against which other interoperability frameworks are compared. Following application of the evaluation criteria, the Technical Group aimed to develop a consensus about which interoperability frameworks should be taken forward into Stage 2, though points of dissent were noted when they arose.

Table 6-1: Criteria for selection of interoperability framework(s) for testing in Stage 2, and their rationale.

	Criterion	Rationale
Ease of interpretation of outputs		
1	Supports geospatial and temporal data	Aids interpretation of outputs at a range of spatial and temporal scales
2	Supports data visualisation	Aids interpretation of outputs via maps, tables, graphs, tables, etc., including uncertainty
Facilitating uptake		
3	Runs on multiple computing platform	Assists with uptake
4	Compatible with a range of existing NZ models	Assists with uptake
5	Open source	Assists with uptake (no IP barriers to use); facilitates further development
6	Suitable for users with a range of technical expertise	Assists with uptake by non-experts while enabling further development by experts
7	Uses recognised data and interoperability standards	Assists with uptake; reduces model development time
Flexibility of use		
8	Support a range of spatial data types and formats	Enables modelling from farm to catchment scales
9	Good technical support and documentation	Ensures ease and continuity of use; future-proofing
Enhancing robustness and reducing runtimes		
10	Includes auditability, provenance, metadata	Ensures robustness, reliability, interpretability and reproducibility of outputs
11	Supports QA/QC of input data	Ensures robustness, reliability, interpretability and reproducibility of outputs
12	Scripting and command line control	Faster operation (e.g., by experts for uncertainty analysis)
13	Can run via distributed computing	Faster operation (e.g., faster run times by parallelisation)
14	Can run steady-state coarse models	Faster operation (facilitates use of models that can run quickly)
15	Support distributed data sources	Reduces model development time and ensures modelling is undertaken with up-to-date data
Barriers to use		
16	Effort to develop	Minimal development effort saves time and cost
17	Transferability	Aids re-use for other locations or modelling objectives
18	Functionality	Ensures flexibility and future-proofing

6.2 Framework Selection

Table 6-2 presents the Technical Group's scoring of the interoperability frameworks based on the criteria discussed in the previous section. Table 6-3 provides the Technical Group's consensus decision as to which of the six interoperability frameworks should be taken forward into Stage 2 of this Programme, which takes information from the scoring and also considers further information.

The Technical Group recommends that OMS3/CSIP as the preferred framework with Delta Shell ranked as the second most suitable framework.

The following points summarise the Technical Group's rationale for selecting OMS3/CSIP over Delta Shell for Stage 2:

- **Range of model applications.** Several agricultural and catchment models have already been configured to operate within OMS3/CSIP, given the framework's main applications from the U.S. Department of Agriculture. This range of model applications was considered by the Technical Group to provide an advantage to using OMS3/CSIP in Stage 2 of the Interoperable Modelling project. In comparison, the narrower range of models already operating within Delta Shell are mostly simulating hydrodynamics and water quality, given the interests of Deltares.
- **Flexibility.** Delta Shell has a more locked-in way of doing things via its API, whereas OMS3/CSIP is more flexible in this regard. The Technical Group concluded that the lower flexibility of Delta Shell may create more difficulty in gaining end-user and developer buy-in.
- **Web services capability.** CSIP has an established interface for delivering models through web services, whereas Delta Shell does not, as it is presently primarily intended for desktop use. The Technical Group recognised the growing importance of delivering models through web services, and concluded that OMS3/CSIP's current capabilities will provide an opportunity for testing web delivery of models within Stage 2 of this project.
- **User interface.** The Technical Group noted that substantial effort may be required to build a user interface for particular applications within OMS3/CSIP, whereas Delta Shell already has a well-developed GUI. However, it was further noted that the development of user interfaces within OMS3/CSIP will benefit from the wide range of tools and scripts already available in the Java environment. Thus, the potential difficulty in developing user interfaces within OMS3/CSIP was not considered a barrier for its use in Stage 2 of this project.
- **Evidence of third party use.** The Technical Group was not aware of any examples of third parties outside Deltares making use of the Delta Shell API. In contrast, there are known examples of third party users building interoperable models through OMS3/CSIP. This track record of successful third party use suggests that the tractability of OMS3/CSIP may result in improved take up by end users following Stage 2 of this project.
- **Evidence of use in New Zealand.** The Technical Group was not aware of any previous or current application of OMS3/CSIP in New Zealand. There has been some use of Delta Shell in New Zealand, for example by DairyNZ for modelling of the Waituna catchment (Southland). The Technical Group felt that the limited examples of previous use in New Zealand would not provide a substantial advantage to use of Delta Shell in Stage 2 of the project.

Table 6-2: Application of selection criteria to six interoperability framework(s) for potential testing in Stage 2.

	Criterion	Bespoke loose coupling	DeltaShell	eWater Source	GeoJModelBuilder	LUMASS	OMS3 and CSIP
1	Supports geospatial and temporal data	yes	yes	yes	yes	yes	yes
2	Supports data visualisation	maybe	yes	yes	yes	yes	yes
3	Runs on multiple computing platforms	no	no	maybe	yes	yes	yes
4	Compatible with a range of existing NZ models	yes	yes	maybe	yes	yes	yes
5	Open source	maybe	yes	maybe	yes	yes	yes
6	Suitable for users with a range of technical expertise	no	maybe	yes	yes	maybe	yes
7	Uses recognised data and interoperability standards	no	partial	partial	yes	no	yes
8	Support a range of spatial data types and formats	yes	yes	yes	yes	partial/yes	yes
9	Good technical support and documentation	no	yes/no	yes	yes/no	yes/partial	yes
10	Includes auditability, provenance, metadata	no	yes	no	?	yes	yes
11	Supports QA/QC of input data	maybe	yes	yes	maybe	no	no
12	Scripting and command line control	maybe	yes	yes	?	yes	yes
13	Can run via distributed computing	yes	yes	maybe	follow up	yes	yes
14	Can run steady-state coarse models	yes	yes	yes	yes	yes	yes
15	Support distributed data sources	maybe	yes	yes	yes	partial	high
16	Effort to develop	high	medium	medium	high	medium	medium
17	Transferability	low	high	high	high	high	high
18	Functionality	high	high	high	high	high	high

Table 6-3: Technical Group's assessment of which interoperability framework(s) should be taken forward for testing in Stage 2.

Interoperability Framework	Take forward to Stage 2?	Rationale
OMS3 and CSIP	Yes	OMS3/CSIP is the framework recommended by the Technical Group for use in Stage 2 of this project. A wide range of types of models has already been made operable within OMS3/CSIP, and the framework's Java basis, open source context and web services support provide important capability for the future. There are some examples of successful use of the framework by third parties, though not yet in New Zealand beyond testing in the FIFM project[5].
Delta Shell	No	Delta Shell has powerful capabilities and there is evidence of incorporation of various of Deltares' hydrodynamics and water quality models in a New Zealand context. Delta Shell has a well-developed user interface. However, Delta Shell appears to offer less flexibility than OMS3/CSIP in terms of range of models already available, open source approach, and flexibility of use via its API.
eWater Source	No	Source has some powerful capabilities and has already been applied in New Zealand. However, its primary emphasis in on hydrology with some water quality aspects, and as such it has limitations for the type of wider economic modelling sought by the OLW Challenge. Further, Source's node-link spatial basis and limited openness may be restrictive for the OLW programme. Recently, Source has moved to more of a proprietary approach (charging for previously-free model use).
GeoJModelBuilder	No	GeoJModelBuilder meets several needs of the project and its flexible open-source standards-based approach is attractive. However, it is still an immature system and its support and development is presently lacking, limiting its applicability to the current project. Despite this, we do expect such approaches to develop further in the future, and the web services approaches to model interoperability is an active area of advancement.
LUMASS	No	LUMASS provides powerful technical features for the OLW programme, such as spatial optimisation, powerful raster libraries, and open-source approaches. However, it is not yet adapted to use interoperability standards, it has a limited user base and it relies on just one developer. These factors limit its appropriateness as an interoperability framework in the OLW programme at present. However, it may serve as plug-in to the selected framework to provide otherwise missing functionality, such as spatial optimisation.
Bespoke loose coupling	No	Bespoke loose coupling is the status quo. Existing applications in New Zealand show that it is typically time-consuming to implement, with considerable up-front planning required. In some cases, glue code needs to be written, for example to translate output from one model to be compatible with input to another model. Thus, bespoke loose coupling is considered as a reference method to compare with other interoperability approaches that will be evaluated in Stage 2 of this programme.

[5] https://teamwork.niwa.co.nz/display/IFM/Framework+for+Interoperable+Freshwater+Models

- **Evidence of open source operation.** The Technical Group concluded that there is more evidence of open source models being provided through OMS3/CSIP compared to Delta Shell.

7 Model list and selection, and supporting data

This section presents a list of models identified at a Technical Group workshop and a list of data to support models, along with data gaps and availability constraints. We also proposed approaches for using OVERSEER and APSIM, and an initial approach for spatial economic optimisation. This is used as a guide for future reference, and also leads to the development of a specific proposal for an initial set of models and supporting data to implement in the selected framework (Section 8).

7.1 Models

7.1.1 Long-list of models

At a two-day workshop held in Wellington on 6–7 June 2017, the members of the Technical Group developed a preliminary list of potential models to be used in conjunction with the chosen interoperability framework, grouped by key model type (Table 7-1). Further details on the models are presented in Table 7-3. This is not an exhaustive list, but illustrates that in many cases there are a number of potential candidates to fill a particular modelling role. The modular and open nature of the interoperability framework should enable adoption of alternative models. An initial attempt at identifying some leading candidates was also made (see highlighted items). In some cases, integrated models from Section 4 are mentioned in the table; in such cases, we envisage extracting a component model from the integrated model.

Table 7-1: Candidate models identified during the technical workshop. Highlighted (bold) items were tentative leading candidates for initial implementation

Model role/class	Initial Candidates from Technical Group meeting[1]
Flow generation	**TopNet**, **SMWBM**, Woods-Henderson, APSIM, SPASMO, WFlow, Irricalc, LUCI.
Contaminant generation	**APSIM**, SPASMO, **LUCI**, **OVERSEER**, Lookup tables (e.g., MGM), NuBalm, SedNetNZ
Surface water flow routing	MIKE models, **Source** routing algorithms
Groundwater flow transport	FEFLOW, **MODFLOW**, GFlow, bucket model
Surface water contaminant transport	Source routing method, CLUES method
Groundwater contaminant transport	FEFLOW, **MODFLOW-MT3D**, GFlow (bucket model)
Production	**APSIM**, SPASMO, **Pasture Growth Forecaster**, CENW, **LUCI**
Water demand	Empirical models, **SMWBM**, Irricalc, SPASMO, **Lookup Table**
Enterprise financials	**NZFarm**, **FARMAX**, **FIF,** WISE, Inform
Ecological response	EcoLab, DELWAQ, CLUES Estuary, Delft3D, CAEDYM, IFIM models, Vollenweider, Periphyton model from TRIM.
Spatial optimization	**NZFarm**, **LUMASS**

[1]This is not an exhaustive list, and does not preclude addition of further models.

Following the workshop, the attributes of these models relevant to the needs of the programme have been identified, to assist with model selection (Table 7-2). A selection models from Table 7-1 deemed to be relevant to the programme are assessed according to these attributes in Table 7-3). Some key potential national datasets relevant to the model are also identified.

Table 7-2: Model attributes assessed

Attribute	Description/alternatives
Static/dynamic	Static Analysis Framework Dynamic Analysis Framework
Model class/use	Flow Generation Flow routing [ground water and surface water] Constituent transport [ground water and surface water] Flow accounting [abstractions, reservoirs] Constituent accounting [point sources, decay] Physical processes Water demand Land-water management RMA Regulation NPS-FM Limit setting Economics
Spatial Scale	Paddock, farm, river reach and catchment
Spatial types	Geometric mesh units Polygons Nodes Networks
Key model domain	Environmental, production

To meet the analysis objectives, the potential models are evaluated against a set of criteria for model inclusion in the potential model list. Some key potential national datasets relevant to the model are also identified.

Table 7-3: Attributes of models.

			Attribute		
Model	**Static/Dynamic**	**Model use**	**Spatial Scale**	**Spatial types**	**Model Domains**
TopNet	Dynamic Analysis	Flow generation Flow routing [sw] Physical processes	River Reach Catchment	Polygons Nodes Networks	Environmental
SMWBM	Dynamic Analysis	Flow generation Physical processes Water demand	Paddock Farm River Reach	Polygons Nodes	Environmental, Production
WFlow	Dynamic Analysis	Flow generation Flow routing [sw] Physical processes	River Reach	Nodes Networks	Environmental
APSIM	Dynamic Analysis	Flow generation Flow routing [groundwater] Constituent accounting [point source, decay] Physical processes Land-water management RMA regulation	Paddock Farm	Nodes	Environmental, Production
SPASMO	Dynamic Analysis	Flow generation Flow routing [groundwater] Constituent accounting [decay] Physical processes Land-water management RMA regulation	Paddock Farm	Nodes	Environmental, Production
IrriCalc	Dynamic Analysis	Physical processes Water Demand	Paddock Farm	Polygons, Nodes	Environmental, production and economic

			Attribute		
Model	**Static/Dynamic**	**Model use**	**Spatial Scale**	**Spatial types**	**Model Domains**
LUCI	Static Analysis	Flow routing [surface water] Constituent transport [surface water] Flow accounting [Abstractions, Reservoirs] Constituent accounting [Point Source, Decay] Physical processes Water demand Land-water management	Farm River Reach Catchment	Polygons Nodes Networks	Environmental, production and economic
OVERSEER	Static Analysis	Constituent accounting [Decay] Physical processes Water demand Land-water management RMA Regulation Economics	Paddock Farm	Nodes	Environmental, production and economical
Nu-balm	Static Analysis	Constituent accounting [Decay] Physical processes Land-water management	Farm Catchment	Polygons Nodes	Environmental, production and economical
SednetNZ	Static Analysis	Constituent transport [surface water] Constituent accounting [Decay] Physical processes Land-water management RMA Regulation NPS Limit setting	River Reach Catchment	Polygons Nodes Networks	Environmental

		Attribute			
Model	**Static/Dynamic**	**Model use**	**Spatial Scale**	**Spatial types**	**Model Domains**
Mike SHE	Dynamic Analysis	Flow routing [groundwater + surface water] Constituent transport [groundwater + surface water] Flow accounting [Abstractions, Reservoirs] Constituent accounting [Point Source, Decay] Physical processes Water demand Land-water management RMA Regulation NPS Limit setting	Paddock Farm River Reach Catchment	Geometric Mesh Units Polygons Nodes Networks	Environmental
e-water Source	Dynamic Analysis	Flow routing [groundwater + surface water] Constituent transport [groundwater + surface water] Flow accounting [Abstractions, Reservoirs] Constituent accounting [Point Source, Decay] Physical processes Water demand Land-water management RMA Regulation NPS Limit setting	Farm River Reach Catchment	Polygons Nodes Networks	Environmental

		Attribute			
Model	**Static/Dynamic**	**Model use**	**Spatial Scale**	**Spatial types**	**Model Domains**
FEFLOW	Dynamic Analysis	Flow routing [groundwater] Constituent transport [groundwater] Flow accounting [Abstractions] Constituent accounting [Decay] Physical processes Land-water management RMA Regulation NPS Limit setting	Paddock Farm River Reach Catchment	Geometric Mesh Units Polygons Nodes Networks	Environmental
MODFLOW	Dynamic Analysis	Flow routing [groundwater] Constituent transport [groundwater] Flow accounting [Abstractions] Constituent accounting [Decay] Physical processes Land-water management RMA Regulation NPS Limit setting	Paddock Farm River Reach Catchment	Geometric mesh units Polygons Nodes networks	Environmental
MT3D	Dynamic Analysis	See MODFLOW these are inseparable	See MODFLOW these are inseparable	See MODFLOW these are inseparable	See MODFLOW these are inseparable

		Attribute			
Model	**Static/Dynamic**	**Model use**	**Spatial Scale**	**Spatial types**	**Model Domains**
CLUES	Static Analysis	Flow routing [surface water] Constituent transport [surface water] Flow accounting [Abstractions, Reservoirs] Constituent accounting [Point Source, Decay] Physical processes Water demand Land-water management Economics	Farm River Reach Catchment	Polygons Nodes networks	Environmental, production and economic
Forecaster	Static Analysis	Physical processes Land-water management Economics	Farm Catchment	Polygons Nodes	Environmental, production and economic
CenW	Static Analysis	Physical processes Land-water management Economics	Farm Catchment	Polygons Nodes	Environmental, production and economic
Empirical Models (water demand)	Static Analysis	Flow routing [groundwater+surface water] Constituent transport [groundwater+surface water] Flow accounting [Abstractions + Reservoirs] Constituent accounting [Decay] Physical processes Land-water management Economics	Paddock Farm River Reach Catchment	Polygons Nodes networks	Environmental, production and economic

		Attribute			
Model	**Static/Dynamic**	**Model use**	**Spatial Scale**	**Spatial types**	**Model Domains**
NZFARM	Static Analysis	Physical processes Land-water management Economics	Farm Catchment	Polygons Nodes	Environmental, production and economic
FARMAX	Static Analysis	Physical processes Land-water management Economics	Farm Catchment	Polygons Nodes	Environmental, production and economic
FIF (Forest Investment Finder)	Static Analysis	Physical processes Land-water management Economics	Farm Catchment	Polygons Nodes	Environmental, production and economic
WISE	Dynamic Analysis	Physical processes Land-water management Economics	Farm Catchment	Polygons Nodes	Environmental, production and economical
Ecolab	Dynamic Analysis	Flow routing [surface water] Constituent transport [surface water] Flow accounting [Abstractions, Reservoirs] Constituent accounting [Point Source, Decay] Physical processes Water demand Land-water management RMA Regulation NPS Limit setting	Catchment	Geometric mesh units Polygons Nodes networks	Environmental
Delwaq	Dynamic Analysis	Physical processes Land-water management Economics	Farm River Reach Catchment	Polygons Nodes Networks	Environmental

			Attribute		
Model	**Static/Dynamic**	**Model use**	**Spatial Scale**	**Spatial types**	**Model Domains**
Clues Estuary	Static Analysis	Flow routing [surface water] Constituent transport [surface water] Constituent accounting [Decay] Physical processes Land-water management	Catchment	Polygons Nodes Networks	Environmental
Delft3D	Dynamic Analysis	Flow routing [groundwater + surface water] Constituent transport [groundwater + surface water] Flow accounting [Abstractions, Reservoirs] Constituent accounting [Point Source, Decay] Physical processes Water demand Land-water management RMA Regulation NPS Limit setting	Paddock Farm River Reach Catchment	Geometric mesh units Polygons Nodes networks	Environmental
FIFM	Static Analysis	Physical processes Land-water management Economics	Farm Catchment	Polygons Nodes	Environmental, production and economic

		Attribute			
Model	**Static/Dynamic**	**Model use**	**Spatial Scale**	**Spatial types**	**Model Domains**
CAEDYM	Dynamic Analysis	Flow routing [surface water] Constituent transport [surface water] Constituent accounting [Point Source, Decay] Physical processes Water demand RMA Regulation NPS Limit setting	Catchment	Geometric mesh units Polygons Nodes networks	Environmental
LUMASS	Static Analysis	Physical processes Land-water management Economics	Farm Catchment	Polygons Nodes Networks	Environmental, production and economic

7.1.2 Proposed set of models for implementation

Table 7-1 shows that there are many models relevant to the objectives of the programme (and that list itself is not exhaustive). It is not practical nor cost effective to include such a large number of models in Stage 2 of the project. Accordingly, a small set of models was selected. The following factors have been considered in selecting models for the initial implementation:

- Avoid duplication in the initial set of models (no more than one model serving the same purpose)
- Preference for models used previously in New Zealand
- Preference for models that are open source and free from IP issues
- Covers a range of key objectives of the programme:
    - Steady state and dynamic.
    - Water quality and quantity.
    - Production and farm financials.
    - Economic model linkage.
    - Demonstrates the ability of the interoperability framework to deliver integrated sets of models and link to available datasets and visualisation.
    - Considers the interests of a range of model providers and end-users.
    - Considers ease of implementation.

Accordingly, the following models have been selected and grouped into three key areas:

1. Steady state water quality (N) and production model.
    - Lookup tables, as for the Canterbury matrix of good management
    - OVERSEER
    - CLUES stream routing
    - Simplified groundwater model from the Smart Aquifer Models programme
    - FarmMax
    - Linear spatial optimisation from LUMASS.
2. Dynamic water quality model
    - Irrigation demand component of LUCI
    - APSIM
    - Dynamic stream routing from Source

3. Water resources model
    - Irrigation demand component of LUCI (daily soil moisture balance and crops)
    - Abstractions and reservoir operation and stream routing from Source or similar

This approach is further elaborated in Section 0. Use of OVERSEER, APSIM, Spatial optimisation and interoperation with other frameworks are described below.

7.1.3 Use of OVERSEER

OVERSEER has already been used in several ways in catchment and within-farm modelling (see Section 4.2.6 and Appendix O). Key methods include:

1. Setting up a library of synthetic OVERSEER models for key farm/system types in different regions, allowing for modification of key inputs (e.g., stocking rate, fertiliser input, irrigation). This would most likely be achieved modifying input XML files, and extracting key model outputs of interest. Care must be taken to ensure biological feasibility when changing such parameters. The OVERSEER model engine would be run each time that OVERSEER outputs are required. This approach has been used for CLUES and has been demonstrated using in the OMS3 framework in the FIFM project, including running OVERSEER as a web service. We envisage running OVERSEER for each property in a catchment. Conceptually, a farm could be broken into blocks, but linking this to spatially-explicit information about block location would be difficult, unless blocks are defined by topography, soils, and land cover only.

2. Providing OVERSEER files for each property and running the OVERSEER programme for each property. Clearly this would only be applicable where there is available data.

3. Pre-running OVERSEER for each property, and providing results to the framework. This has limitations for scenario testing.

4. Developing look-up tables of losses based on typologies of land use. The loss coefficients could be obtained from results of OVERSEER model runs from actual farms of each typology class. This approach could be extended further to include mitigation measures. Such an approach has been adopted in recent catchment scale studies (e.g., the Healthy Rivers Waikato study).

We propose that initially a look-up table method be implemented and run on a farm-by-farm basis. As a subsequent step, full OVERSEER can be run, based on a farm library approach with modifications for key parameters.

Critically, integration of OVERSEER into the proposed Interoperability framework will require resolution of IP and licence issues at an early stage. Where the provision of data comes at a cost (e.g., soil and climate data) this also needs to be negotiated prior to the use of the framework.

7.1.4 Use of APSIM

APSIM is a modelling framework established in Australia with strong New Zealand linkages, for simulating soil-production dynamics at point scale. APSIM models entail setting up detailed crop and soil parameters and process sub-models. As APSIM is a framework in its own right, coupling it into another framework in any intimate way would be complicated and difficult. We therefore propose a simpler approach, whereby APSIM models are set up for different representative farm types, allowing for modification of only a limited set of inputs such as daily climate, fertiliser, and soil characteristics. Outputs would include plant/pasture productivity and some measure of stocking intensity, as well as daily values of drainage, nitrogen leaching losses and N concentrations in the drainage. These APSIM models would be run in their native environment (.Net with MS Windows) for each property, presented as a combined module to the framework.

Initially, we propose to use a selection of APSIM farm systems for Canterbury to demonstrate the approach. We have some experience with this approach already. For pastoral systems, two sheep and beef farm types would be developed that would include the possibility of running a selection of fertiliser and irrigation management systems based on pre-set rule types. Similarly, for cropping systems a multi-paddock approach (similar as done in MGM) with pre-defined crop rotations could be developed that could also include a range of fertiliser and irrigation practices. If this approach is successful, we would proceed to setting up other farm types such as dairy.

A further opportunities to use a APSIM for temporal disaggregation of mean annual OVERSEER results, scaling the APSIM time series to match the OVERSEER annual average value. We proposed that this approach be trialled as a temporal downscaling module in Stage 2.

APSIM is available free of charge for non-commercial use, but commercial uses are subject to negotiation with the APSIM Steering Committee on a case-by-case basis[6]. This would need to be addressed early in Stage 2 of the programme.

7.1.5 Spatial economic optimisation

Current approaches to spatial economic optimisation (in the broad sense) are discussed in Appendix T. Spatial and economic optimisation are applied in many different contexts in New Zealand that are relevant for this programme. For example, to model the economic aspects of land-use change (e.g., Doole, 2015; Daigneault et al., 2016; Monge et al., 2016), to optimise forestry management (e.g., Garcia, 1990; Manley et al., 1991), to maximise ecosystem services (e.g., Herzig et al., 2013), to maximise farm profit (Rendel et al., 2016), and to perform multi-criteria optimisation (Chikumbo et al., 2014). In general, the optimisation applications used in New Zealand so far differ by the:

i) type of objective function, i.e., linear vs. non-linear,
ii) number of objectives, i.e., single vs. multi-objective,
iii) temporal context, i.e., static vs. dynamic; and
iv) algorithm used to solve the optimisation problem, e.g., mathematical programing vs. search heuristics.

According to the most recent literature on spatial economic optimisation, potentially the way to move forward would be to use heuristic approaches (e.g., genetic or evolutionary algorithms) based on their comprehensive inclusion of spatial correlation (i.e., agglomeration), uncertainty, utility-based economic criteria (including preferences as opposed to profit-based only), correlated variation (i.e., copulas), dynamic dimension, multi-criteria objectives, etc. (Rabotyagov et al., 2016; Chikumbo

[6] http://www.APSIM.info/Products/Licensing.aspx

et al., 2014; Lehmann et al., 2013; Musshoff et al., 2009). Such comprehensiveness has proven quite challenging for models/frameworks that rely on mathematical programming principles.

However, given the limited available resources over the next two years in this programme, a trade-off decision needs to be made, as to which "approach" is going to be integrated into (or adapted to) the selected interoperability framework. This is because it is impossible to select more than one model at this stage, or develop one based on more advanced protocols (e.g., heuristics). Therefore, consideration needs to be given to the specific model capabilities required by the stakeholders (i.e., Governance Group), the required effort/cost to adapt the particular approach to the interoperability framework, the modularity of such approach (i.e., independence from monolithic models) and its open source nature. The last two are quite crucial since most complex economic models to date have been written in the General Algebraic Modelling System (GAMS), which contains its own programming language and solver licenses as opposed to the more basic open-source programming languages used in the biophysical and ecological disciplines (i.e., Java, R, Python, C++, C#, etc.).

Based on the previous criteria established for the 2-year pilot project, it is proposed to use the mathematical programming approach, linear programming (LP) to be more specific.

There are now various open-source Linear Programming (a type of optimisation) solvers that could be used (or are being used) by various models or frameworks included in this project. We propose to adopt the LP approach from LUMASS or LUCI for this purpose (the exact decision will be an early work item in the programme).

In the longer term, we propose to adopt more complex economic models as well as innovative approaches (i.e., heuristics), to make the solutions more robust and contemporary.

7.1.6 Interoperation with models from other frameworks

The proposed framework offers opportunities to couple loosely to stand-alone models that are provided via other frameworks (Figure 7-1). We envisage that in some cases it would be appropriate to set up a specialist model in a framework that is well suited to a particular model class, such as a detailed hydrodynamic and water quality model for a particular estuary set up in Delta Shell, and then to link to this model through a command line or other elementary interface. This approach might also provide opportunities to re-use an existing model for a particular area. The model and its input files would be provided as a standalone module able to be run from the proposed framework, which would use selected output files from the model. The model would be set up in its native user interface, but then run through the command line or scripting interface. The user interface, visualisation, primary models, and orchestration would be done via the primary framework. This approach is not as neat or as powerful as incorporating models into the primary framework, due to the need to set up external models for each application. But, it does offer opportunities to leverage specialist models that may not be well suited for incorporation in the primary framework (due to incompatible spatial structures or complexity, for example), or legacy models where the model has already been set up for a particular area.

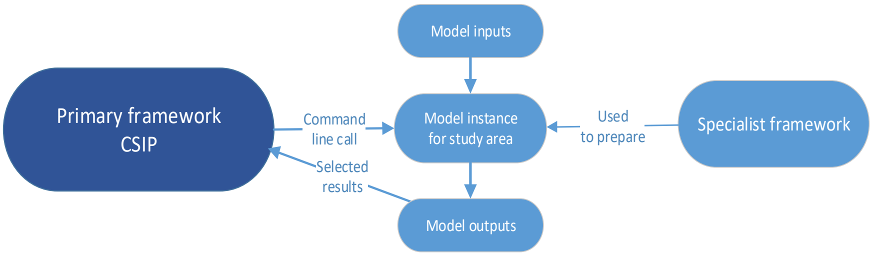

Figure 7-1: Outline of the primary framework linking to a stand-alone model instance prepared with a specialist framework.

7.2 Supporting data

7.2.1 Available data sources for modelling

Some specific sources of available data are listed in Table 4-1. The data sources are organised around the classes from Table 7-4. We also identify how the data can be obtained, and whether there are web services to obtain the data.

7.2.2 Formats and standards

A number of standards and common formats are available in the land and water space, which will be useful for this project:

- The SOS (Sensor Observation Service) provides a standard for time series observations.
- WaterML (Water Markup Language) provides an OGC (Open Geospatial Consortium) standard for water-related observations. The standard provides minimum requirements for delivering data, but the particular data within the schema can vary depending on the provider. For example, different software systems deliver WaterML-compliant flow data with different attributes for flows. Some further standardisation for such data would be desirable.
- NetCDF is a common file format for providing grid and array data (e.g., time-series) in a compact format in meteorology and hydrology, and sometimes in hydrology (e.g., for TopNet). The files contain metadata describing the entries in the file.
- For geospatial data, there are some common formats- including shapefiles (ESRI, but largely open), GeoTiff, and so on, which most geospatial libraries can access, manipulate, and display.
- XML and JSON files (and associated schema) offer text-based approaches for storing structured data, and they are often used in web design. Such files can be compressed well, to make for more compact binary files and for faster data transfer.

There are well-developed approaches for accessing information stored externally, which can be utilised in this project. In the data science community, current developments are focusing on how data is organised and the meaning of data items. At the simplest level, this entails data dictionaries for agreed terms, and at a more complex level entails formal ontologies for representing conceptual relationships between data items, which can be further developed to translate between multiple ontologies. This is an active area of development within the New Zealand and international community. The approach in this project will be to build early systems without adopting ontologies or dictionaries, which is appropriate when there is only a small number of datasets. This allows focus to be maintained on delivering a functioning product, and then allows for progression to adoption of dictionaries and ontologies as they become more readily available.

Table 7-4: List of specific data sources to support modelling.

Type	Subclass	Provider	Name/Description	Web services	Availability	Comment
Land	Topography	University of Otago	15 m topography NZDEM_SoS_v1.0	Yes, OGC	Koordinates, non-proprietary	
Land	Soil	Landcare Research	S-MAP.	Yes	Proprietary for commercial use	Partial national coverage. Landcare also have proprietary mapping to additional parameters.
Land	Soil	Landcare Research	Fundamental Soil Layer	Yes, OGC services	Koordinates, non-proprietary	Becoming superseded by S-MAP
Land	Geology	GNS	QMAP Surficial geology 1:250000	Yes, GeoServer, OGC services	Liberal Creative commons BY licence.	Difficult to relate surficial geology to hydrological parameters or aquifers
Water Features	Drainage features	NIWA	REC/Digital drainage network	Yes. OGC	Koordinates for version 1. NIWA GIS server or manual download of ESRI geodatabase for version 2.	REC2 web service requires access permission.

Type	Subclass	Provider	Name/Description	Web services	Availability	Comment
Water Features	Aquifers	GNS Science	NZ Aquifer Potential Map 1.0	No	Creative Commons BY 3.0 licence	https://www.gns.cri.nz/Home/Our-Science/Environment-and-Materials/Groundwater/Database-and-tools/Maps
Water Features	Lakes	LINZ	Lake boundaries, names from topo data	Yes, OGC	Koordinates	Linked to REC reaches.
Water Features	Aquifer hydraulic properties	GNS Science	Earth Beneath Our Feet	No	Limited use	http://data.gns.cri.nz/ebof/
Water Features	Estuaries	Envirolink Tool	ETI database. Database of estuary shapes and key geomorphic and hydrological parameters.	No	Text (csv) file for data, available through the ETI app.	Extended from the Coastal Explorer. Also, includes some trophic indicator tools. Currently about 450 estuaries.
Other	Point sources	NIWA	Point source loads, compiled from regional councils	No	Not formally available	Gap for national open data
Land use	Land Cover	Landcare Research/LINZ	Land Cover Database	Yes, OGC	Koordinates	
Land use	Land use class	AsureQuality	AgriBase	No	Proprietary	Has other attributes such as stocking rate. Proprietary. By property.

Type	Subclass	Provider	Name/Description	Web services	Availability	Comment
Land use	Land use class	MPI	FarmsOnline	No	Not available externally due to privacy.	
Land use	Land use suitability	Landcare	NZLRI Land Use Suitability	Yes, OGC	Koordinates	
Land use	Cadastral parcels	LINZ	Cadastral property boundaries	Yes, OGC	Koordinates, Linz data service	
Land use	Titles	LINZ	NZ Property Titles	Yes, OGC or Geodata	Linz data service	Ownership fields removed. Partial coverage only.
Environmental Observations	Meteorological observations	NIWA	CliFlo. Time series climate observations	Yes, Curl queries	NIWA web access. Browser or automated queries.	Need subscription. Data must be at least 2 days old
Environmental Observations	Meteorological observations	NIWA	NIWA climate network	Yes	EcoConnect	Proprietary
Environmental Observations	Meteorological observations	Councils	Climate data	Yes, SOS	Separate councils	Variety of formats (about 4) in NZ
Environmental Observations	Meteorological observations	NIWA	VCSN (Virtual Climate Station Network)	Not known	Proprietary	
Environmental Observations	Hydrology	NIWA	Flow data from NIWA network	Yes, SOS	Not known	Still in process, should be ready end 2017. Currently hydrowebportal.niwa.co.nz manual export.

Type	Subclass	Provider	Name/Description	Web services	Availability	Comment
Environmental Observations	Hydrology	Councils	Flow data	Yes, various WaterML2 implementations	Separate councils	Variety of formats (about 4) in NZ
Environmental Observations	Hydrology	GNS Science	National Groundwater Monitoring Programme	Yes	Limited use	http://ggw.gns.cri.nz/ggwdata/disclaimer.jsp?returnTo=%2Fggwdata%2F
Environmental Observations	Stream water quality	NIWA	National Rivers water quality network	Yes	hydrowebportal.niwa.co.nz	
Environmental Observations	Stream water quality	Councils		Not known	Individual councils	Disparate datasets.
Environmental Observations	Stream water quality	StatsNZ	Water quality data, national dataset	Not known	StatsNZ download. Includes RC data.	Curated dataset. Tends to be older data due to curation and approval lags.
Environmental Observations	Lake Water Quality	MfE	Lakes Water Quality database	No	NIWA, MfE	Possibly served through NIWA GIS services soon
Environmental Observations	Groundwater Quality	GNS Science	National Groundwater Monitoring Programme	Yes	Limited use	http://ggw.gns.cri.nz/ggwdata/disclaimer.jsp?returnTo=%2Fggwdata%2F
Existing model outputs	Statistical model for current environmental state	NIWA	RiverTools	No	RiverTools web service	Predictions of current state for more than 100 variables on REC network.

Type	Subclass	Provider	Name/Description	Web services	Availability	Comment
Existing model outputs	Climate model scenarios	NIWA	Climate model run outputs, regional models.	No. NetCDF arrays.	Charges for models that have been run previously, based on extraction effort. Need accounts on supercomputer account, for modelling community. New models planned to be set up on web portal, made freely under the Deep South Challenge.	Large data files. Excel and csv files can be extracted manually. New models still being set up and run.
Existing model outputs	Climate model scenarios	IPCC	Global climate model runs. CMIP archive.	Yes. ESGF have RESTful API.	Openly available. Need free account.	Other mirror sites could be easier to access data (e.g., Earth System Grid Federation).
Existing model outputs	Flow predictions	NIWA	Pre-run TopNet model results	No	Individual arrangements	Large data files. Excel and csv files can be extracted manually.

Type	Subclass	Provider	Name/Description	Web services	Availability	Comment
Water use	Abstractions	MfE	Abstractions consents national database.	Council server data can be queried using sql queries. LAWA have these.	2013-14 https://data.mfe.govt.nz/table/3613-primary-use-and-source-of-consented-freshwater-takes-201314/. Updated council server data can be queried using sql (LAWA have a set of these for abstractions data)	Quite a few tricky details with data quality when taking sql queries. Summaries also on LAWA website. Takes data also available from council servers for some councils – project proposed to automate this.
Water use	Irrigation	MfE	National irrigation layer of irrigated land	Not known	Will be released shortly via the MfE data service.	
Production and economics	National IO data	StatsNZ	National Accounts input-output table	No	2013 tables available for download. http://www.stats.govt.nz/browse_for_stats/economic_indicators/NationalAccounts/input-output%20tables-2013.aspx	Previous version was 2007, published in 2012. Significant lags.
Production and economics	Forest productivity	Scion	300 and site indexes to model productivity for different sites in NZ	No	Not available publicly due to IP issues	Developed using a set of 32K “permanent sample plots” maintained by Scion for ~100 years

Type	Subclass	Provider	Name/Description	Web services	Availability	Comment
Production and economics	Forest productivity	Scion	Forest Investment Finder (FIF) to develop GIS layers of 300 and site indexes	No	Not available publicly due to IP issues	Data would have to be generated on a request basis
Production and economics	Forest productivity	Scion	Forecaster splits productivity by log quality, which is defines what gets exported or goes to domestic mills	No	Not available publicly due to IP issues	Data would have to be generated by running the software for various sites. A layer could potentially be developed.
Production and economics	Forest productivity	MPI	National Exotic Forest Description (NEFD)	No	Excel spreadsheets are available online	Source of volumes per management regime and regions. The NEFD is a bit rough bit it gives you a pretty good idea of regional productivity differences
Production and economics	Forest financials	Scion	Forest Investment Finder (FIF) to develop cost and revenue layers	No	Not available publicly due to IP issues	Data would have to be generated on a request basis
Production and economics	Forest financials	AgriHQ	Forest management costs	No	Access granted to data available online	Paid access to online dataset or to monthly publications
Production and economics	Forest financials	Motu	Forestry costs and profits for various regions in NZ	No	Excel spreadsheets are available online	Freely available

Type	Subclass	Provider	Name/Description	Web services	Availability	Comment
Production and economics	Forest financials	MPI	Historical Radiata pine prices	No	Excel spreadsheets are available online	Freely available
Production and economics	Land values	PropertyIQ	Land values for various parcels under different land uses	No	Access granted to data available online	Paid access to online dataset
Production and economics	Farm monitoring survey data	Beef and Lamb	Sheep and beef survey	No	https://www.dairynz.co.nz/publications/dairy-industry has summary statistics for a range of production and financial indicators for 8 farm classes, in spreadsheet form. Results classified into quintiles for performance benchmarking http://beeflambnz.com/information/on-farm-data-and-industry-production/benchmarking-data/	MPI has monitor farm data until 2012 which covers all ag sector. Now not collecting this data.
Production and economics	Farm monitoring survey data	DairyNZ	DairyNZ Economic Survey, based on DairyNZ DairyBase	No	https://www.dairynz.co.nz/publications/dairy-industry. Summary reports available online.	May be possible to get extracts (not identifying properties).

Type	Subclass	Provider	Name/Description	Web services	Availability	Comment
Production and economics	Production statistics	StatsNZ and MPI	National agricultural production statistics	No	Regional tables downloadable. Not sure about access to finer scale information.	Aggregated regional data available. About 80% or properties. Excludes hort and viticulture. For pastoral, numbers of animals and births, not produce.

7.2.3 Data gaps and availability constraints

Several data gaps were identified by the Governance and Technical Groups. High-priority items include:

- Information on farm performance (for production and economic models, actual data for cropping systems). For example, monitor farm information is not collected and is now out of date, and data held by individual industries is not available publicly for an interoperable modelling system.
- Information on land use and practice/treatment/mitigations, especially at fine scales. Unavailability of such information, will severely hamper the ability of the model system to deliver on the programme objectives of linking from farm to catchment scale. If such data is not available, then representative assumptions based on surveys or analysis of representative farms need to be applied to individual properties or parcels, which introduces uncertainty and undermines the robustness of the analysis.

Other gaps include:

- Spatial data at fine spatial scales (~10m), for use in fine-scale models (e.g., LUCI applied at fine scale, or for setup of spatially-explicit OVERSEER blocks).
- Database on planning rules and restrictions, to guide land use evolution modelling.
- Water management rules (reservoir management, flow restrictions for irrigators). This limits the ability to simulate water resources and environmental flows.
- Standardisation of flow and water quality observations data.
- Standardisation of consented and actual abstractions data.
- Standardisation and availability of council climate observations.
- A maintained database of point source loading.
- Monitor farm data.

Some key data sources are proprietary, including S-MAP, VCSN, AgriBase, NIWA climate network data. Other data, such as outputs from pre-run TopNet models, are of uncertain availability and proprietary status. Farms Online data is not available at the individual property level.

There was also a request from the Governance Group for a one-stop-shop for environmental data to feed models. This has some attraction in terms of efficiency of retrieval of information and simplicity. However, it comes at a cost of maintaining and updating the sources of information, making it available in the appropriate way, and data protection. For the current project, we are taking the approach that data can be provided through standardised distributed data sources accessible through web services and associated API, to enable flexibility of data ownership and a clear authoritative source of data. Local copies of data (which may be transformed into model-specific formats) can be used to get data in the same location as the computing resource, and to avoid repeated retrieval of external data. Ultimately, data warehousing may be appropriate, but that is the beyond the scope of this programme and is not essential for this programme.

8 Proposed two-year development plan

The proposed two-year plan will deliver a selected set of models in the chosen interoperability framework, including links to data where available. The aim is to demonstrate that the framework can meet the programme goals, to justify further investment in adding models. To meet this aim, it will be necessary to include some key models of interest, link them, and provide visualisation components, in addition to using available of available datasets.

8.1 Proposed models

We propose to implement three sets of models in the framework, and demonstrate their application for a selected location. For each of these model sets, we propose to begin with implementing basic models, then to build complexity as time and resources permit, following an iterative pathway of delivery and extension.

This proposal does not preclude parties from providing further models to the system, including alternative models to some of the proposed model components, developing their own integrated models in the system, or linking the CSIP system to models provided as services by other frameworks (for example, a receiving water body model in Deltares).

These models and their integrated forms will be available to a range of modellers, including parties outside the providers or immediate stakeholders, subject to a licensing constraints that might be common to all users (for example, access permissions to a particular model component).

The proposed approach somewhat artificially separates water quantity and quality processes. Successful completion of the above applications will enable more thorough integration in Stage 3 of the programme.

8.1.1 Steady state contaminant and production and model

1. Mean annual N leaching determined from look-up tables by land-use and typologies, along with mitigation reductions, applied to a selected catchment (e.g., Ruataniwha or Hurunui). Farms subdivided by typology and linked to the REC network. Simple accumulation down the drainage network.

2. Add linkage to OVERSEER representative models with modification of selected inputs, as an alternative source model for appropriate areas.

3. Link to groundwater decay model, simplified model from the SAM (Smart Aquifer Models) programme and CLUES surface water routing.

4. Conduct spatial optimisation, linked to lookup tables of FARMAX results and other relevant data.

5. FARMAX and OVERSEER integration. This is a fairly large task in its own right, but is considered to be an important building block for future farm systems – leaching interoperability.

8.1.2 Dynamic contaminant

1. Link to TopNet pre-run outputs of runoff generation, with temporal disaggregation of loads and daily routing.
2. Link to flows from the water resources model (see below)
3. Add linkage to APSIM models to generate time-series of leachate losses, and route on a daily basis with decay.
4. Link to groundwater lag-decay model (Rotan catchment) or another simplified model from the SAM programme.
5. Link to MODFLOW-based groundwater model.

8.1.3 Water resources model

1. Daily soil moisture balance and crop model with local groundwater store to derive irrigation demand, runoff and recharge generation for each property.
2. Daily stream flow routing driven by rainfall-runoff model, with specified abstraction rules linked to demand and environmental flows.
3. Add surface-groundwater linking (from SAM programme or alternative).
4. Addition of reservoir storage and release module.

8.2 Data sources

We propose to use web services and standards-based server queries to extract base data, where they are available. Specific data types that can be obtained in this way include:

- Topography
- Drainage network (i.e., REC2)
- Lakes
- Soil
- Climate observations and VCSN (Virtual Climate Station Network) data
- Land cover
- Property boundaries
- Irrigation layer
- Flow observations

For each of these sources, CSIP will be linked to the standard data services, using OGC-based queries, and results for a particular study area cached locally in the relevant standard format for the projection, study area, and time-period of interest. Further translation/adapter services will be developed to convert these standard formats to the model-specific formats, units, and variable names where necessary, and to perform basic data quality checks (e.g., range, completeness).

Other data will need to be extracted on a manual basis, with further translation.

8.2.1 Geoprocessing and Visualisation

CSIP does not have core functionality for visualisation and geoprocessing. However, there are many free Java-based libraries and web services that could be used for these tasks. For initial work, we propose to use:

- JFreeChart for charts

- Google APIs, plus visualisation frameworks such as D3 (JavaScript)

- Geomajas for Web GIS, or Geotools with Geoserver, for map generation, interaction, display, and spatial data manipulation. There are also simpler solutions for pure map display.

Adoption of a core set of tools does not preclude the use of other client side tools or web services (e.g., rShiny (R mapping) or ESRI Arc services), because they can be provisioned through web services approaches.

8.3 Work plan

The work plan has been constructed to enable iterative development and refinement, and to ensure deliver of example applications while managing risks (Figure 8-1). We therefore propose monthly team meetings and quarterly review meetings during the programme, with a full governance meeting at the one year mark. The quarterly meetings will allow for evaluation of progress against the stated tasks, demonstrations of delivery, and re-evaluate the next steps, overall progress and suitability of the overall design. Figure 8-1 also identifies the key parties to be involved, although this still needs to be refined and negotiated.

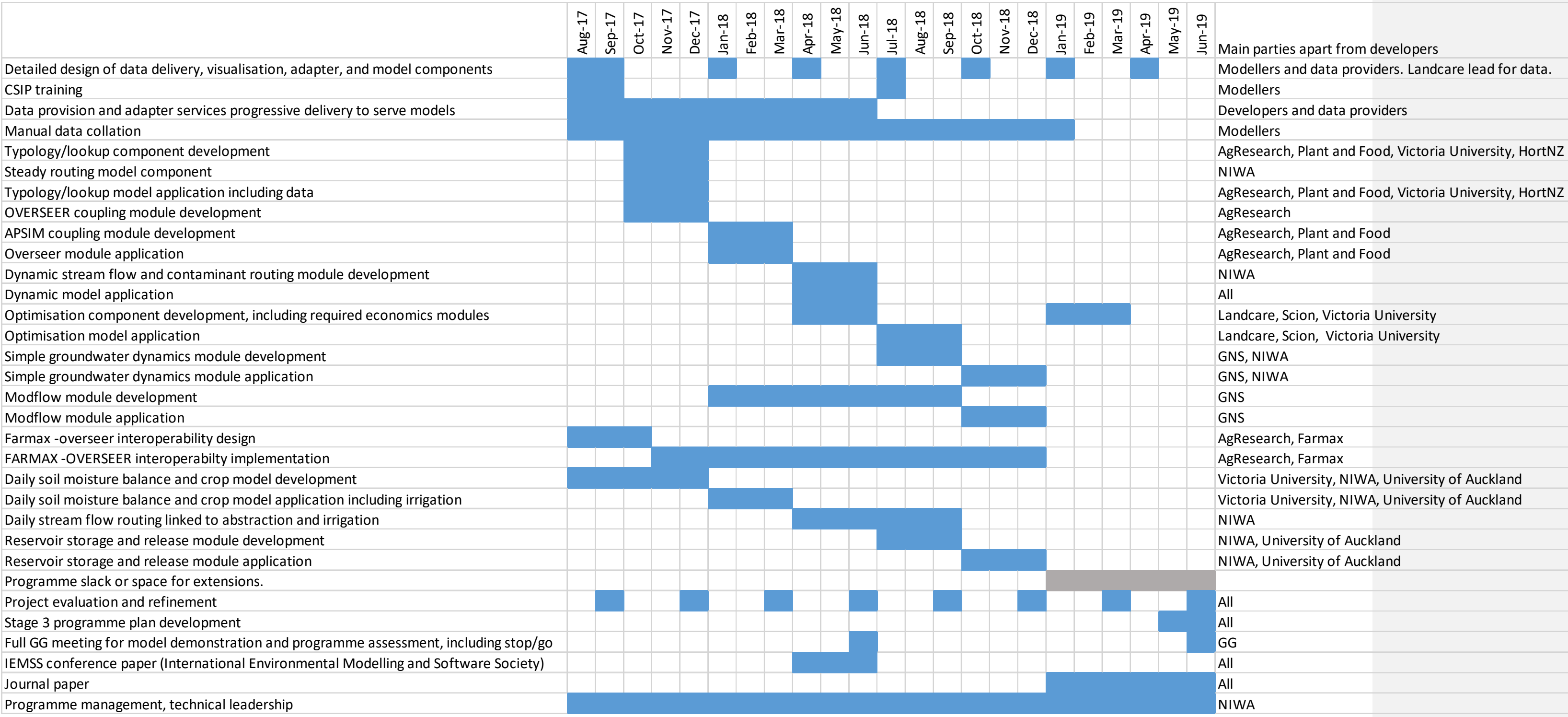

Figure 8-1: Gantt chart showing indicative timeline for Stage 2 and indicative main parties involved.

8.4 Resources

There are some key task areas that need to be resourced.

1. Linking data sources to model framework. There are some data sources already available through web services. The data sources need to be linked to the model framework through standard services, translated into a form suitable for model input, and stored. This is primarily a job for a data scientist.

2. Provision of models to the framework. In several cases, model components will need to be extracted from existing integrated models, or adapted from available code bases, to fit the needs. This will require input from a range of model developers with specialist expertise (contaminant generation, rainfall-runoff, water resources, farm financials and production, economics), along with a key developer to set the models up as services with an appropriate interface.

3. GIS visualisation libraries and user interface. A web developer with familiarity with user interfaces and GIS libraries will be required to develop the applications. Some assistance will also be required from a core developer to link OGC services to the CSIP framework.

4. Application of the models to the case study. This will require domain specialist modeller (not necessarily coders) to supply reasonable data and parameters, and reality-check the results.

5. Leadership and coordination role.

6. Technical writing for paper presentation.

Indicative full-time equivalent (FTE) allocations for external human resources are shown in Table 8-1

Table 8-1: Indicative FTE requirements.

Role	FTE per year	Funding source
Systems developer, data scientist, and web programmer with GIS expertise	1.5	Primarily Challenge, but with some co-funding. Possibly outsource some aspects.
Model provision	1	Co-funding
Model application	0.5	Co-funding
Leadership and coordination	0.2	Challenge
Communication, meetings	0.3	Co-funding
Total	3.5 (1.5 developer, 2 scientist)	See below.

We expect to also fund some international and local travel for training, international conference presentations, and meeting attendance (20k per year).

Total funding requirement, based on $270k per annum for developers and $300k per annum for scientists, is $1,025,00 per annum. With anticipated Challenge funding of $350k per annum, there is a co-funding requirement of $675 k per annum, which could be provided by co-funding equivalent to $100k per year from a notional seven parties.

Other needs which have **not** been resourced include:

- Organisation of data into standard formats and available through web services;
- Funding for model licences; and
- Organisation of IP and governance.

8.5 IP and data provision considerations

There are some key IP considerations which need to be addressed, these are listed below.

8.5.1 Model IP and ownership

- Access to OVERSEER, as an embedded model;
- Access to APSIM for commercial applications;
- TopNet licensing (will NIWA apply IP constraints; and
- FARMAX licensing.

CSIP does not have a well-developed programming interface for managing licences or user fees. We propose that means to achieve this be investigated through the first two years of the programme. It is anticipated that suitable libraries and interfaces will be provided because this is a common concern for web-based systems.

We propose that models or components that are created as part of this programme through direct or co-funding will be open source with no licence restrictions.

The framework software itself, any associated libraries for visualisation and user interaction, and data formats will be free and open source with liberal licences, so there is no IP constraint in that regard.

8.5.2 Data availability and ownership

Some key data sources are currently private or proprietary, or both (Section 7.2). Key data sources which could provide difficulties include:

- S-MAP,
- VCSN,
- Permissions to Council flow and water quality data,
- Land use (AgriBase, FarmsOnline, industry databases),
- Linkage between farm titles and land use and
- Farm financials datasets from industries.

Some existing applications of CSIP have privacy and security controls for data, due to privacy concerns about soil conservation service advice to individual farmers. We propose that technical methods to address these requirements be investigated over the first few months of the programme. As with proprietary models, we expect that suitable methods will be available, because data privacy is a common concern for web applications.

8.6 Risk management

A list of potential risks and mitigations from the programme proposal are listed in Appendix U.

Overall, the programme does involve significant risks (see Table 8-2), which is inevitable given the complexity of the topic and rapidly-evolving technology. The risks have been minimised through careful selection of the framework, strong governance, and an incremental staged development approach.

Table 8-2: Key programme risks and proposed mitigations.

Hazard	Likelihood and consequence.	Mitigation
Resource constraints limit task completion. The project is ambitious and resource requirements are uncertain.	Resource constraints are likely to be limited, with reduced depth or completeness of outputs.	Work programme is incremental with iterative assessment of progress and goals.
Recruitment of developers and data scientists.	Moderate likelihood in early stage, leading to need to revise project scope.	Governance Group to commit resources at an early stage for this aspect. Outsource some work components.
Key component models are not available due to licence restrictions.	Moderate likelihood, will reduce attractiveness of the final product.	A key item for the GG. Continue with IP role for GG. Backup approach is to use lookup tables only for OVERSEER. Develop irrigation module within this programme rather than third-party.
Key data sources are not available, or not provided as a web service.	Moderate likelihood, resulting products less attractive.	Manual data extraction where need be instead of automated. Case study in an area that has already been studied. Provide default or dummy data if not available from authoritative source.
CSIP and OMS development ceases or becomes proprietary.	Unlikely. Would have large consequence.	Have chosen system with a low likelihood of this happening. Switch to another system using similar components.
Insufficient support, documentation and training available.	Unlikely. Moderate to high consequence.	Iterative programme assessment to determine whether this is a problem.
Long-term maintenance and development cost not supported.	Moderate likelihood. Could result in system folding.	Assess for Stage 3. Need strong Governance Group. Adoption of web service and modular approach means that components could be re-used in new system.
The system becomes dominated by a single provider, resulting in reduced trust and engagement.	Low likelihood, moderate consequence.	System is open source, free, and flexible.
Interoperability technology becomes superseded.	Moderate likelihood in long term, moderate consequence.	Have adopted flexible emergent web services approach to modelling, which can be re-purposed in different frameworks. Framework does not rely on an intricate API. Core system developers are very agile, adapting to new technologies.
System becomes more difficult to use than anticipated or has key technical limitation.	Low-moderate likelihood, can potentially derail delivery.	Review progress and choices on a regular basis. Engage professional systems developers.
Alternative proprietary systems are developed and capture the market, especially for specific applications.	Moderate likelihood, could undermine the programme.	Continue open approaches with key support and expertise used. Allow for use of models developed in other frameworks, via web services.

9 Conclusions and recommendations

Based on a systematic approach, this report presents a proposal for implementing a set of land-water models within an interoperability framework that addresses needs of the key users, along with accompanying rationale. The report documents the outcome of the first stage of the three-stage Interoperable Modelling Systems for Integrated Land and Water Management programme.

The proposed framework is CSIP/OMS3, developed by Colorado State University with the US Department of Agriculture as a cornerstone user.

The proposed work programme for Stage 2 (2 years) will put a selected set of models into the framework and set up the models in a selected catchment. This will demonstrate the use of the framework to provide integrated models pertinent to the objectives of the Governance Group and Challenge. The set of models encompasses a range of temporal resolutions, linking from farm to catchment scale, water quality and water resources, production and economics. An iterative approach has been adopted to ensure staged delivery of demonstrated results, including frequent evaluation and reconsideration of the development pathway.

The proposal entails significant financial and human resources beyond those provided directly by the Challenge. It is suggested that Challenge funding will be key to providing core developer and data scientist support, and overall programme coordination and leadership, while co-funding will contribute primarily to model provision and applications.

The next step is for the Governance Group to approve the work programme and arrange co-funding. They will also need to resolve some key IP and data provision constraints that have been identified.

10 Acknowledgements

The authors wish to thank Tim Davie and Mike Beare for chairing the Technical Group meetings, and the Governance Group for responding to the needs survey. We also wish to thank, Ballance AgriNutrients, DairyNZ, HortNZ, and Ravensdown for providing direct co-funding for this project.

11 References

Chikumbo, O., Goodman, E., Deb, K. (2014) Triple bottomline many-objective-based decision making for a land use management problem. *Journal of Multi-Criteria Decision Analysis*.

Daigneault, A., Greenhalgh, S., Samarasinghe, O. (2016) Economic impacts of multiple agro-environmental policies on New Zealand land use. *Environmental and Resource Economics*.

Doole, G. 2015. A flexible framework for environmental policy assessment at the catchment level. *Computers and Electronics in Agriculture,* 114: 221–230.

Garcia, O. (1990) Linear programming and related approaches in forest planning. *New Zealand Journal of Forestry Science,* 20(3): 307–31.

Herzig, A., Ausseil, A. and Dymond, J. (2013) Spatial Optimisation of Ecosystem Services. In *Ecosystem Services in New Zealand – Conditions and Trends*. J.R. Dymond (Ed). Manaaki Whenua Press, Lincoln, New Zealand.

Lehmann, N., Briner, S., Finger, R. (2013) The impact of climate and price risks on agricultural land use and crop management decisions. *Land Use Policy,* 35: 119–130.

Manley, B., Papps, S., Threadgill, J., Wakelin, S. (1991) Application of FOLPI A linear programming estate modelling system for forest management planning. Forest Research Institute. FRI Bulletin No. 164. Rotorua, New Zealand.

Monge, J., Warren, P., Richardson, J. (2016) Integrating forest ecosystem services into the farming landscape: A stochastic economic assessment. *Journal of Environmental Management,* 174: 87–99.

Musshoff, O. and Hirschauer, N. (2009) Optimizing production decision using a hybrid simulation-genetic algorithm approach. *Canadian Journal of Agricultural Economics,* 57: 35–54.

Nativi, S., Mazzetti, P., Geller, G.N. (2013) Environmental model access and interoperability: The GEO Model Web initiative. *Environmental Modelling & Software*, 39: 214–228. http://dx.doi.org/10.1016/j.envsoft.2012.03.007

Rabotyagov, S., Campbell, T., White, M., Arnold, J., Atwood, J., Norfleet, M., Kling, C., Gassman, P., Valcu, A., Richardson, J., Turner, R., Rabalais, N. (2014) Cost-effective targeting of conservation investments to reduce the Northern Gulf of Mexico hypoxic zone. *Proceedings of the National Academy of Sciences* 111(52): 18530-35.

Rendel, J., Mackay, A., Smale, P., Vogeler, I. (2016) Moving from exploring on-farm opportunities with a single to a multi-year focus: Implications for decision making. *Journal of New Zealand Grasslands,* 78: 57–66.

12 Glossary of abbreviations and terms

API	Application Programming Interface. A set of standard commands or protocols to enable some software to interact with other software.
APSIM	Agricultural Production Systems Simulator A modelling framework and set of models for dynamic simulation of soil-production dynamics at point scale, with development largely in Australia.
bespoke	Written or adapted for a specific user or purpose (customised).
CAEDYM	A lake eutrophication model, developed in Western Australia.
CENW	A forest growth model including carbon and nutrient cycling, developed in New Zealand.
CLUES	Catchment Land Use for Environmental Sustainability. A catchment model used in New Zealand for predicting contaminant losses, socio-economic indicators, and estuarine status. Maintained and managed by NIWA.
component	A part of a software system which performs a defined and specific task.
CSIP	Cloud Services Infrastructure Program. A system developed by Colorado State University for running models as web services.
Delft3D	A hydraulic model for streams, lakes and estuaries developed by DeltaRes.
Delta Shell	A framework developed by DeltaRes for environmental models, especially spatial simulation models for water.
DELWAQ	A water quality simulation model for streams, lakes, and estuaries, developed by DeltaRes.
Eco Lab	A numerical programming environment developed by DHI Denmark for ecological simulation.
ESGF	Earth System Grid Federation. A UK web portal for environmental data.
FARMAX	A farm systems model developed by Farmax.
FEFLOW	A groundwater model using finite-element numerical techniques, owned by the Danish Hydraulic Institute.
FIF	Forest Investment Finder. A forestry economics and spatial planning model developed by Scion.
FIFM	Framework for Interoperable Freshwater Models. An MBIE project running from 2011 to 2014.
GeoJModelBuilder	A framework for coupling models and running them as web services with a mapping interface. Developed in China.
Geonamica	A spatial simulation framework developed by the RIKS Institute in the Netherlands
GeoServer	An open source server for sharing geospatial data. It publishes data from any major spatial data source using open standards.

GFlow	A groundwater model based on the analytical element numerical method. Developed by the Haitjema Software consultancy.
GoldSIm	Engineering and environmental A dynamic simulation programme which has been applied to environmental problems. https://www.goldsim.com/Web/Products/Modules/ContaminantTransport/
GUI	Graphical user interface.
IFIF	Instream Flow Incremental Methodology. Model for assessing habitat associated with different flows in rivers.
integrated model	A model that has a number of sub-models of parts of a system integrated to represent the larger system.
interoperable	Models, data sources, or software that can exchange information using mutually recognised formats or mechanisms.
IP	Intellectual property.
Irricalc	An irrigation simulation programme developed by Aqualinc.
Koordinates	A repository for geospatial data, holding a large number of New Zealand spatial datasets, and accessible through a web portal.
LAM	Land Application and Management modelling approach for economic and environmental systems, developed by Graeme Doole (University of Waikato).
LINZ	Land Information New Zealand
LUCI	Land Utilisation and Capability Indicator. An integrated catchment model and framework developed by Victoria University.
LUMASS	Land Use Management Support System. A spatial system dynamics framework including spatial optimisation, developed by Landcare Research.
MfE	Ministry for the Environment
MGM	Matrix of Good Management. A system of loss rates and mitigation measure effectiveness values used by Environmental Canterbury.
MIKE	A model suite from DHI (Danish Hydraulic Institute).
MODFLOW	Modular groundwater flow model, developed by the United States Geological Survey.
MPI	Ministry for Primary Industries
MT3D	A groundwater mass transport model linked to MODFLOW. Recently extended to surface-groundwater modelling.
MyLand	An integrated model for silviculture management and planning, developed at Scion.
NetCDF	A compact data format for array and time series data, commonly used for climate, earth systems, and oceanographic data.
NPS-FM	National Policy Statement for Freshwater Management.
NuBalm	A nutrient balance model for forest plantations.

NZFarm	Landcare Research model for assessment of environmental and economic implications of land management policies.
OGC	Open Geospatial Consortium. An international standards body for geospatial data.
OLW	Our Land and Water Science Challenge
OMS3	Object Modelling System version 3. A modelling framework developed by Colorado State University.
OpenMI	A model coupling standard developed in Europe.
orchestration	Operation of software components in a co-ordinated and interacting manner
OVERSEER	A programme for fertiliser management and nutrient loss estimation from New Zealand farms and horticulture.
Pasture Growth Forecaster	A pasture growth forecasting tool developed by DairyNZ.
REST	Representational State Transfer. A protocol for requesting and receiving information via the web. A web service using REST is called a RESTful service.
RMA	Resource Management Act
SAM	Smart Aqifer Models. An MBIE research programme led by GNS, investigating surface-groundwater model simplification and uncertainty.
SedNetNZ	A New Zealand version of the SedNet erosion model, developed by Landcare Research.
Server	A computer set up to perform some function (for example, running calculations or providing data) for another computer (the Client).
SMWBM	Soil Moisture Water Balance Model developed by the Williamson Water Advisory consultancy
SOS	Sensor Observation Service. An OGC standard for time series observations data from environmental sensors.
Source	eWater Source catchment model
SPASMO	A plant-soil dynamics model, developed by Plant and Food Research.
TopNet	A catchment scale hydrological model developed by NIWA. An extension of the TopModel hydrological model which includes, among other things, stream routing of flows.
TRIM	Tukituki River Model. A dynamic periphyton growth model developed by NIWA and applied in the Tukituki River.
WaterML2	Water Markup Language version 2. A standard for hydrological time series data.
WCS	An OGC standard for providing spatial coverages (grids) through the web.
web service	A processing task conducted by a computer via the world-wide web, typically used to provide or process data or perform some calculations.

WFlow	A spatially-lumped hydrology model developed by DeltaRes.
WFS	Web Feature Service. An OGC standard for providing vector spatial data through the web.
WISE	Waikato Integrated Scenario Explorer. An integrated spatial simulation model developed using the Geonamica software for the Waikato Region.
workflow	Set of tasks conducted in a co-ordinated and repeatable way.
World Wide Web (or Web)	A system of internet servers that support documents formatted in HTML (HyperText Markup Language).

Appendix A Governance and Technical Group members

Governance group

Name	Organisation
Tim Davie	Environment Canterbury. Chair and representing Regional Councils
Richard McDowell	OLW
Ken Taylor	OLW
Serina Callachan	Ministry for the Environment, project coordinator
Sarah Bromley	Plant and Food Research
David Burger	DairyNZ
Warwick Catto	Ballance Agri-Nutrients
Bryce Cooper	NIWA
Glyn Francis	AgResearch
Suzie Greenhalgh	Landcare Research
Matt Harcombe	Beef and Lamb New Zealand
Chris Keenan	HortNZ
Brent King	Greater Wellington Regional Council
Diana Mathers	Foundation for Arable Research
Alister Metherell	Ravensdown
Kelly Palmer	Ministry for the Environment and Iwi adviser
Caroline Read	OVERSEER Ltd, observer
Gerald Rys	Ministry for Primary Industries
Basil Sharp	University of Auckland
Mara Wolkenhauer	University of Otago

Technical Group

Name	Organisation	Role/Expertise
Tim Davie	Environment Canterbury	Chair
Sandy Elliott	NIWA	Technical Lead. Catchment modelling.
Serina Callachan	Ministry for the Environment	Project co-ordinator
Nic Conland	Independent scientist. HortNZ representative.	Catchment modeller
Chris Daughney	GNS	Groundwater modeller
Hans Eikaas	DairyNZ	Environmental modeller
David Eyers	University of Otago	Geospatial modeller, computer scientist
Alex Herzig	Landcare Research	Geospatial modeller
Bethanna Jackson	Victoria University of Wellington. Also, representative for Ravensdown.	Hydrologist and environmental mathematician
Juan Monge	Scion	Resource economist
Asaad Shamseldin	University of Auckland	Hydrological modeller
Tarek Soliman	Landcare Research	Economist
9Jo Sharp	Plant and Food Research	Agricultural systems modelling
Gabriella Turek	NIWA	Environmental software developer
Iris Vogeler	AgResearch	Agricultural systems modeller

Appendix B Objectives survey of Governance G9roup

See table on the next page.

Organisation and respondent	What questions do they see the interoperable model system addressing?	What benefits would a successful interoperable model system bring to your organisation	What are key things that the system should be able to predict, and at what spatial and temporal scales	Are there any particular models you would like to see catered for by the framework	Key supporting datasets the framework should utilise, especially in your area of interest	Are there any key gaps in models, data, or interoperability that need to be addressed?	Do you have any further comments or feedback you would like to pass on?
NIWA (Bryce Cooper)	What are the effects of changing land use/implementing mitigations on the downstream water quality and the economic effects of those changes by sector, including employment? And the converse, what is the optimal mix of land uses to achieve the water quality limits downstream while maximising economic benefits.	Efficiency in bringing datasets and models together to tackle particular issues	Water flows, water quality, farm and catchment economics in response to changes in response to land use and mitigations. Spatial and temporal scales - well, we want everything of course but how realistic is that here?	TopNet, CLUES	Land cover, land use, land title, soils, stocking rate, fertiliser use, topography, river network, climate, river flow, water quality	Heaps! Models are only as good as the data and the representation of the processes that are within them.	Concern about 'beyond the project' - such frameworks require long-term funding to keep them current and useful.
Landcare (Suzie Greenhalgh) (transferred from earlier comments)	Look at the implications of policy, resource constraints, technology changes, etc. on production and environmental quality. At least look at implications of actions on the land for water needs, including climate change. Should look at wider values or ecosystem services. Also, identification of how to move to the highest value use.		Cover all major land uses. Scale from local to catchment.	NZFarm optimisation, LUMASS land use optimisation, SedNet. Get LUMASS capabilities into interoperability standard.	S-map. Other data available through the LRIS portal.	Getting hydrology models to work across scales. Key difficulty with obtaining information on farm performance (MAF monitor farm equivalent). Several existing models and datasets aren't compliant with interoperability standards.	
DairyNZ (David Burger) (transferred from earlier comments)			Catchment to national scale, farm and seasonal resolution, impact of dairy and land use scenarios on water quality, including effect and cost of mitigations.		Daily climate data.		
Ravensdown (transferred from earlier comments)			Two tiers for timescales: Annual-seasonal and daily.	LUCI. Leverage value by linking with other frameworks using standard formats and protocols.		In relation to LUCI, need more work on attenuation, groundwater residence times, in-stream processing, farm systems, tactical farm management, better coupling to forest models. Difficulty providing spatially explicit information at small scales using data sources provided at coarser scales. Better information on the implications of attenuation for the farmer. Better knowledge for the farmer of timescales for environmental response.	
AgResearch	Improved ability to link land-environment models and datasets more effectively. For example, run FARMAX and OVERSEER as a single pass, link current models to national datasets (APSIM, to S-map, APSIM to daily climate data). Ability linking multiple existing models and datasets in risk and land use suitability frameworks			FARMAX, OVERSEER, APSIM	S-map, daily climate data.	Ability to have a good testing regime for models; Harmonisation of concepts of spatial units (e.g., 'blocks' between models); Resolution of IP/licensing issues will be critical.	

Organisation and respondent	What questions do they see the interoperable model system addressing?	What benefits would a successful interoperable model system bring to your organisation	What are key things that the system should be able to predict, and at what spatial and temporal scales	Are there any particular models you would like to see catered for by the framework	Key supporting datasets the framework should utilise, especially in your area of interest	Are there any key gaps in models, data, or interoperability that need to be addressed?	Do you have any further comments or feedback you would like to pass on?
Horticulture New Zealand	Sporadic, uncoordinated accounting frameworks being developed that cannot be utilised in a commercial environment	An ability to coordinate our investment with other investors to establish a plausible and recognised basis for resource allocation over time.	The relationship between loads and concentrations of contaminants to water and the water quantity accounting system needed to allow for more efficient allocation.	Source, OVERSEER, APSIM, SPASMO, SedNet, MODFLOW, FEFLOW, urban catchment models	Root zone reality flux meter results over time, sediment discharge data from Muddy Waters programme, S-MAP, NIWA climate data, Council SOE data	Integration of consents and compliance data, bacterial transport, linking urban and rural in peri-urban environments, public and private data sharing agreements	
Greater Wellington	There isn't a recognised information system that helps decision makers (national scale, catchment scale or property scale) understand the links between land use/land management and their impacts on environmental, social, economic and cultural values. Nor is there a recognised system to explore how changes in land use/land management practice/treatments might impact those values.	Regional Councils need to explore potential futures (a range of scenarios) with their communities. This is about the direction of travel rather than the detail (as many models(ers) tend to fall back on) to achieve the 'accuracy'. Interoperable in the limit setting context when exploring potential management and policy options is about agility and speed. Can this platform deliver this? Could be a useful base modelling system to use, locally customise or build upon, for exploring alternative future catchment management and consequences of those alternatives	At a broad level, the system needs to cover an assessment of key indicators of environmental, social, economic and cultural (though these may be better evaluated through an adaptable Iwi assessment framework?) values. Needs to be able to explore alternative future development extents and practices/treatments. However, the specifics would very much depend on the purpose of such a platform and the questions being investigated/decisions being supported. There needs to be a high awareness of user needs, the questions needing addressing and an understanding and acceptance of what is fit for purpose. The challenge is to think outside of the traditionalist mind set.	NA	NA	Some of the challenges to overcome with interoperable models to get to this are about agility, speed of delivery, speed of models, level of detail and complexity that individual especially biophysical models go into. The agility, flexibility and adaptability required to meet some of the objectives within a national model seems a large challenge. There is also an issue of appropriate complexity and the readiness to come to an acceptable level of inaccuracy for given purposes. Need land use and land use practice/treatment data	I'm concerned about the exclusion of urban land use and land use practice/treatment from the project. This leaves a significant gap in the utility of the model for council use and the complete understanding of mixed and urban dominated catchments.
MfE	Many model packages for separate problems. Cumbersome to organise models to work together for integrated assessment	In terms of informing decision making it will: allow more efficient use of stakeholder resources, allow more robust evidence to be presented. An obvious example of where this may be used is in limit setting processes	Uncertainty in terms of model outputs, predict a range of variables, allow a range of scenarios to be assessed. Should ideally have the ability to predict at the national through to sub-regional scales, i.e., ability to adjust resolution. Daily time scale would be nice.	OVERSEER is the only one I'm familiar with	NA	It needs to address 'ki uta ki tai', so from the atmospheric variables through to what is leaving the freshwater domain at the bottom of catchments.	NA

Organisation and respondent	What questions do they see the interoperable model system addressing?	What benefits would a successful interoperable model system bring to your organisation	What are key things that the system should be able to predict, and at what spatial and temporal scales	Are there any particular models you would like to see catered for by the framework	Key supporting datasets the framework should utilise, especially in your area of interest	Are there any key gaps in models, data, or interoperability that need to be addressed?	Do you have any further comments or feedback you would like to pass on?
FAR	Addressing the problem that farm system and catchment models do not communicate seamlessly	confidence that modelling exercises that have an end impact on the farm business (limit setting for example) are drawing on all the information together and linking it together to get robust results	NA	OVERSEER is the only one I'm familiar with.	NA	there is a lack of information and actual data from cropping systems.	As a small sector, we have little contact with modelling work that is or has been done by regional councils and CRI's through MBIE programmes. We are in a black hole and being there, probably not able to effectively contribute to this programme of work. I always assumed that models had interoperability in the first place. Silly me!
MPI	Making the first real steps in getting better interoperability between currently popular models used in NZ at relevant spatial and temporal scales and incorporation of financial data.	Enable more complex modelling to address more complex policy questions at various spatial and temporal scales to address social, economic an environmental issues.	Should be scalable, enable scenario analysis, be simple to use, at a farm to catchment scale, predict out a minimum of daily and max of 100 years.	FARMAX, OVERSEER, MyLand, CLUES, NZFarm	Agricultural Production Survey data, Census data, Soils database, climate data and drought index, climate change scenarios at a NZ regional level, LCDB, need to ensure a process of updating databases on a regular basis.	There are variable quality of data at spatial and temporal scales and there need to be a process to test data quality. There also needs to be consideration of how uncertainty is propagated through interoperable models and what levels are acceptable.	Need to see real progress in interoperability. i.e., getting acceptable numbers and graphics out
ECan	A framework for freshwater accounting; ability for stakeholders involved in limit setting processes to all use same base data and frequently same agreed models	Allow contention in limit setting to be on agreeing values rather than disputing science on whether values/objectives will be achieved.	nutrient transfer from farm to catchment scale at seasonal basis; water flows at daily (catchment scale); economics is a "nice to have"	NA	Regional council monitoring data (through LAWA?); OVERSEER outputs	Single source data	

Appendix C Bespoke Coupling of Stand-Alone Models

Introduction

This section introduces the key terms and concepts related to bespoke coupling of stand-alone component models, with emphasis on the main strengths and weaknesses of this approach. Comparison is made to other approaches described elsewhere in this report, such as the use of an off-the-shelf interoperability framework or a fully integrated model.

Stand-alone models are individual self-contained software packages that model specific environmental processes or sub-systems. These stand-alone models are developed to be mathematically, temporally and spatially complete for a specific primary physical/chemical system, e.g., groundwater or surface water. For the remainder of this chapter on bespoke coupling approaches, the term 'model' or 'component model' will be used for stand-alone models that are linked together. These models will typically require information from what are considered secondary physical/chemical systems, but will introduce simplifying assumptions for simulating these; for example, groundwater models may utilise simplified surface water systems and surface water models may utilise simplified groundwater systems.

Coupling refers to the linking or interfacing of different stand-alone models. Tight and loose coupling could be utilised. Loose coupling (Belete et al., 2017) refers to the process of exchanging input and output files between different component models that are run independently, perhaps at different times, by different parties and/or on different machines, and usually in a linear sequential fashion. Within loose coupling, the parameters of each component model are defined through independent calibration procedures. Tight coupling refers to a system by which individual models exchange input and output data in a more intimately-joined way (for example, two-way exchange of data between models in each time-step, and use of common computer memory) and often within a larger software system and on a single computer. In tight coupling, parameters may be determined through a joint calibration/inversion procedure.

Bespoke coupling refers to the loose coupling of models using codes or approaches that are developed for a specific application. In other words, the coupling of the models is *bespoke* because it is not achieved using an off-the-shelf interoperability framework such as those discussed in other sections of this report. Bespoke coupling has the advantage of being highly flexible, because it is not bound by any hard-coded restrictions on how coupling between models is to be performed (as might exist in a previously developed interoperability framework). On the other hand, bespoke coupling can have the disadvantage of being time-consuming, because the coupling solutions must be individually developed and tested for the application at hand. In contrast, an interoperability framework may have created generalised solutions for coupling that suit a wide range of studies.

Intended Purpose

Virtually any type of system can be addressed via bespoke coupling, provided the required stand-alone models are available and the project's timeline allows for the development and testing of the bespoke coupling approach. Increased recognition of the truly holistic natural and anthropogenic environmental issues has introduced a bespoke modelling age, as distinct disciplines come together to jointly address environmental management requirements – by pressing together their discipline-derived computer models. Selected recent examples of bespoke coupling undertaken in New Zealand are discussed below. Due to its bespoke nature, use of this approach is targeted towards modelling experts and code developers. Thus, once created, the system would likely remain in the domain of expert users for on-going maintenance and operation.

Models

Any type of stand-alone model can be considered for incorporation into a bespoke coupling framework - the only requirement is that data files can be utilised as model input and output. Limitations beyond the individual models own limitations may arise within the coupling framework, for example, due to the differing spatial and temporal scales of the component models.

Data

The model coupling occurs through exchange of data. The majority of effort for bespoke coupling lies within the translation of data between different stand-alone models. This may involve, for example: coordination of system translations, unit translations, file format translations; integration or fractionation of model outputs across different time steps; and selection, integration or fractionation of model outputs across different spatial locations or areas.

Institutional support

Institutional support for bespoke coupling is a challenge. This is because, by its very nature, bespoke coupling involves hand-building a system of interacting stand-alone models for a specific problem or application. This will require domain experts familiar with each of the different stand-alone models, and these experts may reside in different organisations. Bespoke coupling also requires development of particular pieces of code to link the models together, and the coding experts may be from an additional organisation, who may also not be familiar with the particular requirements and foibles of the individual stand-alone models. Thus, bespoke coupling of stand-alone models requires the input of many experts and no one institute will necessarily take ownership of on-going maintenance of the system.

To-date, because of budgets, time-lines and the technical challenges involved in bespoke coupling, the final modelling system has generally not been user-friendly (for example, by lacking a well-designed GUI). This means that operation of the modelling system will not be easily picked up by end-users at institutions separate from those of the original developers. The low adoption rates at end-user institutes means that overall there will be fewer organisations that have knowledge of how to operate the system, and maintain or adapt it for future use.

Technical aspects of integration

Examples for Illustration

As noted above, bespoke coupling can involve virtually any number or type of stand-alone models. Thus, to provide some concrete examples for discussion, the remainder of this report focuses on two applications of bespoke coupling presently underway in New Zealand. This section discusses the technical details of bespoke coupling in both cases, whereas the following section presents overall experiences, strengths and weaknesses in their development and use. For other examples of bespoke coupling the reader is directed to Belete et al. (2017) and references therein.

Coupled Groundwater-Surface Water Model, Southland

The first example to be discussed in the remainder of this report pertains to coupling of groundwater and surface water models for parts of the Southland region for and in collaboration with Environment Southland (Rawlinson et al., 2015). The modelled area encompasses a large part of the

Southland Plains and is shown in Rawlinson et al. (2015). The work is not yet finalised but the main technical aspects of model integration can be presented here. This work is being undertaken in conjunction with Environment Southland's *Fluxes and Flows* project within the *Water and Land 2020 & Beyond* programme (Environment Southland, 2017).

The original plan for this work was the loose coupling of three models (Figure 1). Workshops were held including modellers from NIWA and GNS Science and Environment Southland staff to conceptualise and plan the loose coupling between the groundwater model (implemented in FEFLOW) and the surface water flow model (implemented in TopNet) and the surface water transport model (implemented in CLUES). These workshops defined the necessary types of interactions between these models, the points of interaction, and the development, formatting and exchange of appropriate input and output files. Preliminary targets for calibration between the models were also tentatively set and the number of iterations for exchange of information between each model was planned.

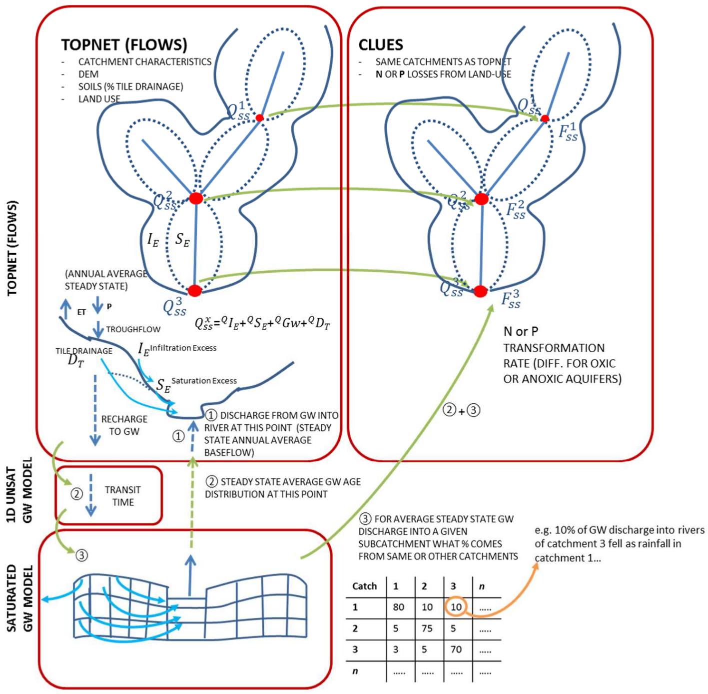

Figure 1: Schematic illustration of the technical aspects of loose coupling between groundwater and surface water models for the Environment Southland *Fluxes and Flows*. Q represents river flow and F represents the flux of a contaminant of interest (nitrogen or phosphorus). Source: Rawlinson et al. (2015).

Collaborative Modelling Project, Ruamāhanga, Wellington

The second example to be discussed in the remainder of this report pertains to the Collaborative Modelling Project (CMP) presently being undertaken by Greater Wellington Regional Council as a component of its Ruamāhanga Whaitua process (Greater Wellington Regional Council, 2016). The

CMP involves bespoke coupling of a wide range of models representing surface water hydrology, groundwater hydrology, ecology, farm systems and economics (Figure 2). The model domain encompasses the Ruamāhanga River catchment in the Wairarapa Valley. As for the preceding example from Southland, the Ruamāhanga modelling work is still underway. Hence, the final results cannot be presented but the technical and experience-of-use aspects of model integration can be discussed in general terms here.

Similarly, to the Southland example, the development of the bespoke coupling requires ongoing workshops and communications between modellers and staff from a multitude of organisations.

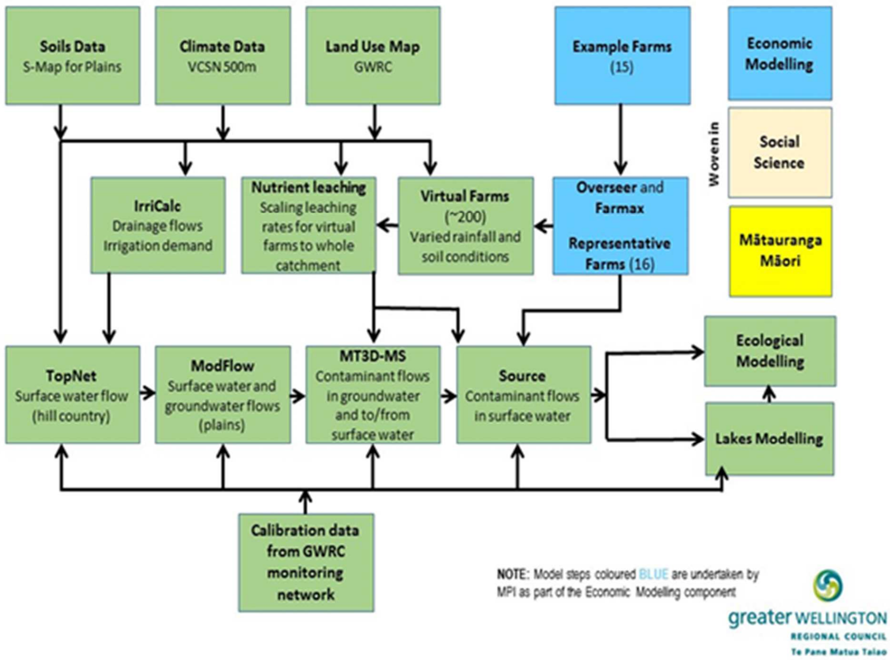

Figure 2: Schematic illustration of the stand-alone models and coupling links for the Ruamāhanga Collaborative Modelling Project . Source: Greater Wellington Regional Council (2016).

Model coupling

For the Southland example, the CLUES aspect of the bespoke coupling was deferred to allow some further development work, and the coupling was limited to lose coupling of the FEFLOW and TopNet models. Specifically, TopNet would be used to calculate recharge to groundwater (at the appropriate time step). This would be passed to the groundwater model and used as input on the top surface of the groundwater model. The groundwater model would then be run and used to predict groundwater inflow to streams at each node, i.e., at each place where TopNet needs to know baseflow.

The model coupling for the Ruamāhanga example is more complicated because it involves a greater number of individual component models (Figure 2). The models and their coupling is as follows:

- Consistent input data were used across each model that needed common data, for example, TopNet, IRRICALC and the Nutrient leaching components all made use of the same climate data from NIWA's Virtual Climate Station Network (VCSN).
- TopNet (NIWA, 2016) was used to model the surface water flows from the mountains and hills to the MODFLOW and Source models which model surface and groundwater flows across the Wairarapa valley floor.
- IRRICALC was used to calculate rainfall recharge, rainfall runoff or 'quick flow', and irrigation water demand. Irrigation demand modelling relied on irrigated area and was linked to surface and groundwater consents databases to water take locations (Aqualinc, 2016). The IRRICALC water demand, recharge and runoff outputs were then imported as inputs for MODFLOW and Source.
- The OVERSEER and FARMAX models were used together to develop farm financial and nutrient budgets for a number of example farms operating in the Ruamāhanga Whaitua. OVERSEER was then used to scale up the results from the example farms to the range of rainfall and soil types across the catchment. OVERSEER and FARMAX were also used to model on-farm mitigation options for N and P.
- The OVERSEER and dynamic SedNet models provided the nutrient and sediment loadings to Source (Jacobs, 2016) and MODFLOW/MT3DMS in order to simulate transport of contaminants through the surface water and groundwater systems (OVERSEER, 2016). Groundwater-surface water nutrient load fluxes were exchanged between MODFLOW and Source.
- Surface water and groundwater flows and contaminant concentrations from TopNet, Source and MODFLOW/MT3DMS are used by the lakes models to simulate water levels, contaminants and ecological conditions of Lakes Wairarapa and Onoke.
- Selected outputs from all models are used in a Bayesian Belief Network to evaluate surface water ecological conditions and variables important for social impact and cultural assessments.
- Social impacts and impacts on Māori cultural values are assessed using expert interpretation of outputs from multiple models, local data and expertise (e.g., water quality and ecological conditions, census or economic data and local knowledge and Mātauranga).

For both of the above-mentioned examples, the coupling is to be achieved via hand-built codes specifically constructed for each step. Feedbacks between the models are not directly catered for. Instead, the data exchange between the coupled models is to occur for a specific number of iterations and then stop. Any requirement to add another component model would necessitate extensive coding and re-design of the modelling system.

Data import and export

In the Southland study, NetCDF (network common data form) files are used to transport data between the groundwater and surface water models. NetCDF files are scientific-oriented self-describing data formats, which have the advantage of being very compact, which facilitates transfer between different members of the modelling team. The content of these NetCDF files follows the existing structures produced by TopNet and includes information such as catchment identification number, time dimension, and other model variables such as flow. Python scripts had to be created to read data from the TopNet output NetCDF file and convert it into the formats required as input by the groundwater model. The conversion steps involved translating between time units and time codes (e.g., UTC to NZ time), and apportioning calculated flows between TopNet catchments and MODFLOW cells. Once the groundwater model has been run, a new NetCDF file was created that required the calculation of baseflow to TopNet catchments. The Waimatiku catchment was selected as a test case for loose-coupling (work is presently underway). Within this catchment, the groundwater model domain contains 52 TopNet catchments, with 11 of these linked to river nodes of the two streams within the catchment. These 11 TopNet catchments are able to transmit fluxes between surface water and groundwater.

In the Ruamāhanga study, exchange of data between models was generally undertaken using csv files. The content of the csv file was selected by discussion among the modellers to ensure that information would be interpretable and complete. The csv file format is not especially compact and so it was often necessary to create compressed zip files for data exchange between models. Again, for the groundwater model interface: pre-processing of the provided input files into the format for the groundwater model was required, and post-processing of the groundwater model outputs into locations and data values required by subsequent models.

A challenge in both studies arose from the different approaches for spatial discretization used by the different models. For example, TopNet uses the sub-catchment as its base geographic unit, whereas FEFLOW uses triangular elements (Figure 3). Thus, for the Southland study it was necessary to develop a scripted methodology that would assign FEFLOW's triangular elements to the TopNet catchments, and also to apportion flows from different TopNet catchments to different FEFLOW elements wherever the boundaries did not coincide (i.e., one FEFLOW element might extend across the boundary of more than one TopNet catchment). This same issue of non-correspondence of boundaries of spatial units occurred in the Ruamāhanga study. For example, the groundwater model used square cells instead of triangular elements, but these still needed to be interfaced with TopNet catchments. The same difficulty arises when interfacing the groundwater or surface water models with other models that use land parcels, farms or even paddocks as their main spatial unit.

The Southland and Ruamāhanga studies also had to deal with non-coinciding spatial coverage for different models. As shown in Figure 3 the Southland groundwater model covered only a portion of the area represented by the surface water model. This meant that flows had to be calculated wherever a river crossed into the groundwater model are, then assigned as a boundary condition. The same challenge was faced in the Ruamāhanga study.

Resolving the above-mentioned issues for spatial discretization and spatial coverage required extensive communication between the groundwater and surface water modellers early in the studies, including setting of provisions for accounting for these differences during data transfer between the component models. These challenges could potentially be addressed to a large extent by adopting an off-the-shelf interoperability framework instead of building a bespoke coupling solution because it would already utilise an interoperable spatial framework.

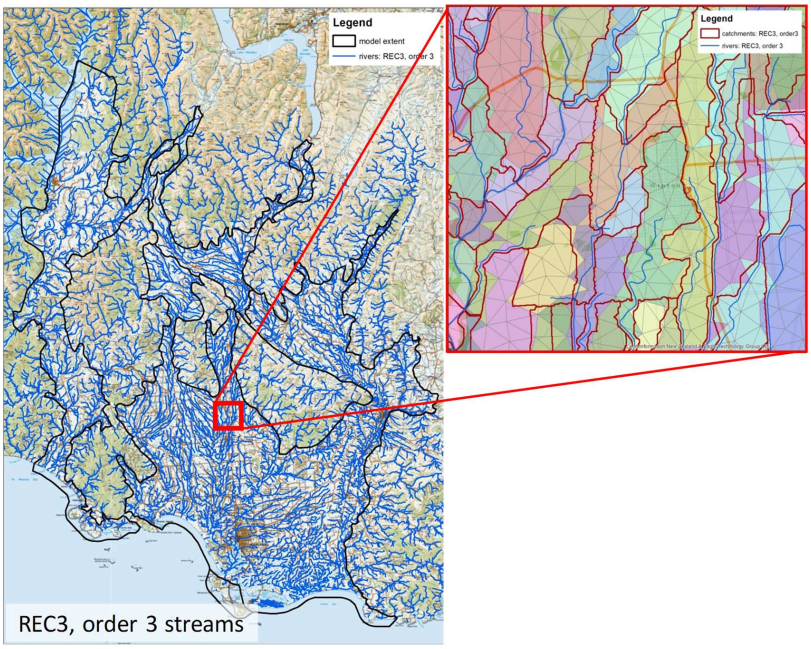

Figure 3: Different systems used for discretization of the model domain by TopNet and FEFLOW for the Southland groundwater-surface water model. Left pane shows the river network depicted by TopNet in comparison to the model domain for FEFLOW. Right pane shows the base geographic unit used by TopNet is the catchment (red outlines) whereas FEFLOW uses triangular elements (grey lines); colour coding shows how collections of FEFLOW elements were defined to approximate the TopNet catchments. Source: Rawlinson et al. (2015).

Visualisation

GIS was the primary tool used to visualise outputs in the Southland study which involved models that incorporate geospatial elements (e.g., MODFLOW, FEFLOW, TopNet). Selecting the GIS package was a straight-forward matter that was decided by discussion amongst the modellers and regional council stakeholders early in the project. Aside from GIS, the Southland study involved a variety of graphing methods to depict outputs.

The Ruamāhanga study is developing its visualisation and reporting approach to build on the idea of a ‘pyramid of information’. Outputs from each model will generate large amounts of detailed information, an important base for developing, reviewing and passing information between the models. These detailed outputs are to be integrated across models and disciplines to help provide decision-makers with simplified information to assist decision-making. The detailed nature of information at the base of the pyramid allows that final presentation to take any number of forms, including narratives, mapping, symbols (e.g., arrows, smiley faces), graphs and tables.

User interface

No graphical user interface was produced for the Southland or the Ruamāhanga study. This is not surprising given the bespoke nature of the coupled modelling in both studies: it would be unusual for the expert modellers to require or be given time to construct a user interface that would be self-explanatory, robust and simple enough to pass off to other, less technical users. Thus, interaction with the models in each study was largely by command line interface and the hand-built codes developed to pass information back and forth and configure and run the individual stand-alone models. This lack of a graphical user interface is likely a feature of most bespoke coupled modelling systems for the aforementioned reasons.

Metadata, scenario management, provenance, auditability

The Southland project used NetCDF files to transfer information between the groundwater and surface water component models. The NetCDF files were structured to include metadata such as model version used, date the model was run, etc.

Metadata was not routinely captured in electronic form in the Ruamāhanga project. Information related to the model versions used, the conditions under which scenarios were run, etc. was typically only described in the emails that accompanied exchange of input and output files between different members of the modelling team. In some cases, the csv files also included embedded explanatory comments.

In general, there is nothing to prevent the routine electronic collection of metadata as part of a bespoke coupled modelling system. However, defining exactly what metadata to capture and how to do so is a responsibility of the modelling team for each project, whereas an off-the-shelf interoperability framework may have already been set up to record such information as default.

Underlying language and computing requirements

Any programming language(s) can be selected for the development of the scripts that tie together the different models in a bespoke coupling project. Python was the language used for most of the coupling scripts in the Southland and Ruamāhanga projects by the groundwater modellers. Python was selected because it has many different frameworks and resources already developed and freely available for programmers. Additionally, Python is readily interfaced with GIS systems, which assists with visualisation of the modelling outputs; and with PostgreSQL databases, which assists with data management. In both projects, PostgreSQL databases were utilised by the groundwater modellers to underpin the coupling architecture.

The computing resources required for running a bespoke coupled modelling system will depend on the models included. Where the models require significant computing power, then so too will the coupled modelling system. The computing resources required will also depend on the approach taken for calibration and uncertainty analysis (see below). The Southland and Ruamāhanga projects involved use of several hundred processors on a Linux cluster during calibration and uncertainty analysis.

Uncertainty methods

Bespoke coupling of stand-alone models allows for bespoke approaches for uncertainty analysis, and some flexibility in terms of what model inputs the modellers incorporate into the uncertainty analysis for any model output. Stitching together these uncertainty analysis outputs from one model to provide the uncertainty of inputs to the next model requires some creativity.

One constituent model in the Ruamāhanga modelling study (TopNet) used a generalized likelihood uncertainty estimation (GLUE) Monte Carlo-based approach (Bevan and Binley, 1992). The GLUE approach relies on the generation of multiple realisations of model inputs, running the model with each realisation, and then filtering those realisations for model outputs that match historical data. Predictive outputs from each model run are then collated into a "predictive probability distribution". This analysis assesses the uncertainty of the estimated flow time series based on the uncertainty in rainfall, evapotranspiration and soil parameter model inputs. This GLUE analysis took months of model run times to complete on a super computer.

In contrast, the MODFLOW/MT3DMS component model of the Ruamāhanga modelling study needed to assess the combined impact of the uncertainties of the TopNet, OVERSEER, IRRICALC and Source component models. This was required because these models all provided inputs to the MODFLOW/MT3DMS model, and therefore their uncertainties impacted on the model output uncertainties. The uncertainties for each of these models were derived in a range of ways (e.g., Shepard et al., 2013). Because of the more complex model interdependencies and the associated combined model run times, a Monte Carlo analysis that integrated all these models was not computationally practical. Instead, an approximate, but quicker, linear Bayesian uncertainty analysis (PREDUNC) was adopted to describe the impact of model inputs and model parameter uncertainty on the model "prediction variance" (Doherty, 2015). PREDUNC is a linear algebra-based approach that accounts for system uncertainties, including aquifer heterogeneity and model input variability, and the calibration constraints imposed on that heterogeneity via a distributed parameter set. This method was also used to assess the spatial disposition of parameter reliability in the MODFLOW/MT3DMS model.

The Monte Carlo based approaches and the linear uncertainty analyses described above define the opposite ends of the spectrum of model uncertainty analysis in terms of effort and accuracy. There are other methods that may be applicable that fall between these two end points, e.g., hybrid methods such as "Null space Monte Carlo" (Tonkin and Doherty, 2005), or ensemble Kalman filtering (Kalman, 1960). The computational time and resultant costs associated with the more complex of these methods can preclude their use.

Management of IP and proprietary material

IP arrangements must be developed on a project-by-project basis wherever bespoke coupling approaches are used. These arrangements must consider the background IP associated with each stand-alone model, as well as with any IP created through the process of coupling. Typically, the bespoke coupling scripts are written within an open-source environment, however, stand-alone component models are often proprietary.

Quality assurance

Methods for quality assurance may include standardised testing or benchmarking of the coupled modelling system, documentation and model version control, and internal and external reviews. Bespoke approaches for model coupling require consideration of these quality assurance elements on a project-by-project basis, due to the lack of use of a stand-alone, previously developed interoperability modelling framework.

Typically, bespoke coupling approaches consist of quality assured stand-alone models that are linked by data reformatting and exchange. The stand-alone models are quality assured via their standard methods, and it is not considered that any additional uncertainty or quality reduction is introduced by this data-exchange method.

In the Southland example, quality assurance of the coupled groundwater-surface water model is being addressed in a number of ways. First, the two component models (FEFLOW for groundwater and TopNet for surface water) have both been subjected to extensive peer review and been successfully used in a variety of studies previously. Second, the modelling team is working with regional council partners to identify the model design requirements and criteria for successful performance and calibration. Third, the final model design, method of operation, and outputs will be documented in reports co-authored by all involved organisations and peer-reviewed by experts external to the project.

The Ruamāhanga Collaborative Modelling Project involves quality assurance methods like those described in the Southland example. The constituent stand-alone models have all been previously described, tested and used in New Zealand. The collaborative approach to designing model requirements, selecting models and developing their coupling provides for ongoing and iterative review of individual and coupled model performance. Using a number of subject matter and modelling experts brings a range of perspectives and collaborators can challenge one another and hold each other to professional account. The process of using the outputs of one model as inputs for another has provided opportunities to identify and respond to unexpected results or missing data as the project develops. Reports are peer-reviewed by experts external to the project.

Experience with Use of the Framework

The following section describes the experiences from the Southland and Ruamāhanga coupled modelling projects, based on information from Gyopari (2016) and supplemented with reflections from the modellers and regional council staff involved in both projects. The emphasis in this section is on the technical and implementation hurdles that are reasonably likely to arise in any project that involves bespoke coupling of component models.

Does it meet the intended purpose?

Both the Southland and Ruamāhanga coupled modelling projects were still underway at the time of writing of this review. Thus, it is not presently possible to determine how well these modelling systems met the original intended purposes or to what extent their use has improved economic or environmental outcomes or processes, or improved efficiency or effectiveness of decision-making. However, initial experiences of the modelling team are summarised by Gyopari (2016).

Barriers to use

Building bespoke solutions for coupling component models is a widely-used approach, and IP constraints will vary on a project-by-project basis. Prior to the initiation of technical work in any such project, there is a need to carefully consider the IP of the stand-alone models and the codes that may need to be built to couple them. Agreement on IP management must be reached by all project partners, including end-users, so that the coupled modelling system is open for discovery, extension, modification and application to other projects afterwards. This allows follow-on value to be obtained from the often-large effort required to construct a bespoke coupling methodology.

Aside from IP issues, other barriers to use include: model licensing constraints, model computing requirements, on-going costs with model maintenance, and dependence on specific staff contributing organisations, with minimal apparent contingency for staff absences/turnover.

As noted above, the Southland and Ruamāhanga projects are still underway and so there has not yet been any effort to extend these coupled modelling systems to other catchments. However, there is intent from both Environment Southland and Greater Wellington Regional Council to apply elements of the coupled modelling system to other catchments in their regions. Certainly, both councils will capitalise on their learnings from these projects in relation to the difficulties that can be encountered during construction of coupled models.

Flexibility, extensibility and adaptability

The bespoke coupling approach typically has several strengths and weaknesses related to flexibility of use:

- The bespoke approach has the advantage of being completely flexible in its design, in terms of individual models chosen for inclusion and the methods adopted for their integration. This means that the bespoke approach can be applied to any project, providing the required models are available or can be built. In contrast, an off-the-shelf interoperability framework may be limited in its architecture or the specific component models that it can accept.
- Bespoke coupling of models may allow for greater flexibility in overcoming numerical instabilities, especially compared to fully integrated models.
- The bespoke coupling approach also preserves the ability of the constituent stand-alone models to operate independently. This may be an advantage if the modellers or end-users wish to explore scenarios that require the use of only one (or few) component models, because the individual models may run relatively quickly and be quite numerically stable.

Conversely, the bespoke coupling approach may have strengths or weaknesses related to extensibility and adaptability:

- Although the bespoke coupling approach is in principle suitable for any application, it has the disadvantage that it may require lengthy code and method development work at the start of the project (e.g., for exchange of data between models). Due to the bespoke nature of such coding, it may not be possible to easily adapt scripts from one project to the next.

- Bespoke coupling can become particularly complicated if the individual models are based on very different underpinning assumptions or process representations. For example, an unsaturated zone might be included in a surface water model but not in a groundwater model, and this difference may make them quite difficult to couple.
- It is possible that the stand-alone models will use different approaches for discretizing the spatial domain. Some examples are given in the preceding sections (e.g., see Figure 3). This may necessitate the time-consuming creation and testing of new methods that translate between different spatial scales or approaches for discretisation.
- It is also possible that the stand-alone models will operate at different time steps. Again, some examples are given in the preceding sections, and these may need to be dealt with through bespoke coding solutions. It may be relatively straight-forward to integrate from smaller to larger time steps (e.g., converting a model's output from daily time step to a weekly average), but disaggregating time steps requires more assumptions.

Ease of use in particular applications

Overall, the ease of developing and using a bespoke coupling approach depends on how well the goals of the project are articulated and managed, given the above-listed challenges to flexibility, extensibility and adaptability (see Gyopari, 2016):

- The project's vision, goals, budget and timelines must be clearly conveyed and well-understood by the modelling team and the modelling team must match the modelling complexity with the project needs. This may require specific and early effort in the project from a modelling coordinator or a professional facilitator.
- The modelling team also needs to build a clear, collective understanding of how collaboration will occur within the project. Professional development in collaborative processes may be beneficial for a team composed of experts with different domain expertise and/or from different organisations.
- There may be a very high level of task dependence during the set-up and running of the coupled modelling system. For example, modellers from one organisation might need to run a model that generates output that is then used as input for a different model that is operated by a different organisation. In such cases the second model cannot be run until the first model is completed. This interdependency must be accounted for in planning project timelines, but even so can compromise delivery if unanticipated complexities in running one stand-alone model then leads to cascading delays in running of other models in the system.

Bespoke coupling introduces a need to be particularly clear about the information needs from the models to enable careful matching of the selected models to these information needs. A clear understanding of purpose and level of modelling that is fit-for-purpose needs to consider elements such as:

- What decisions are being made and what information is required to inform those?
- What natural and anthropogenic systems need to be characterised?

- What changes to those systems need to be provided for and characterised?
- What temporal and spatial scales are information required?
- What are end users' technical abilities to use the models and their information?

Suitability for the needs of the OLW programme

The bespoke-coupling approach is already being used in New Zealand so there are some examples of bespoke coupled models already available (or being built) that can be used for comparison to other interoperability approaches.

References

Aqualinc (2016) Groundwater recharge and irrigation demand modelling. Ruamāhanga collaborative modelling project. *Prepared for Greater Wellington Regional Council.*

Bandaragoda, C., Tarboton, D.G., Woods, R. (2004) Application of TopNet in the distributed model intercomparison project. *Journal of Hydrology,* 298: 178–201.

Belete, G.F., Voinov, A. and Laniak, G.F. (2017) An overview of the model integration process: From pre-integration assessment to testing. *Environ. Modelling & Software*, 87: 49–63.

Beven, K.J., Binley, A.M. (1992) The future of distributed models: model calibration and uncertainty prediction. *Hydrological Processes,* 6: 279–298.

Brunner, P., Simmons, C.T. (2012) HydroGeoSphere: A Fully Integrated, Physically Based Hydrological Model. *Groundwater*, 50(2): 170-176. doi:10.1111/j.1745–6584.2011.00882.x.

Doherty, J. (2015) Calibration and Uncertainty Analysis for Complex Environmental Models. *Watermark Numerical Computing*, Brisbane, Australia. ISBN: 978-0-9943786-0-6.

Environment Southland (2017) Water and Land 2020 & Beyond. http://waterandland.es.govt.nz/

Greater Wellington Regional Council (2016) The Ruamāhanga Modelling Project. http://www.gw.govt.nz/assets/Environment-Management/Whaitua/Ruamāhanga/Collaborative-Modelling-Resources-BrochureA4WEB.PDF

Gyopari, M. (2016) The Ruamāhanga Whaitua Collaborative Modelling Project (CMP): Interdisciplinary Team Science Reflections. *Report prepared by Earth in Mind Limited/Greater Wellington Regional Council for Ministry for the Environment.*

Kalman, R.E. (1960) A new approach to linear filtering and prediction problems, Transactions of the ASME – *Journal of Basic Engineering,* Series D, 82: 35–45.

OVERSEER (2016) Nutrient Budgets: supporting New Zealand's primary industries. https://www.OVERSEER.org.nz/

Rawlinson, Z.J., Toews, M., Daughney, C., Zammit, C., Kees, L. and Moreau, M. (2015) Development of a regional steady-state groundwater flow model loosely-coupled to a surface water model for the Southland region. *Conference of the New Zealand Hydrological Society*, Hamilton, New Zealand, December, 2015.

Shepherd, M., Wheeler, D., Selbie, D., Freeman, M. (2013) OVERSEER: accuracy, precision, error and uncertainty. In: Accurate and efficient use of nutrients on farms. (Eds L.D. Currie and C L. Christensen). http://flrc.massey.ac.nz/publications.html. *Occasional Report*, No. 26. Fertilizer and Lime Research Centre, Massey University, Palmerston North, New Zealand.

Tonkin, M.J., Doherty, J. (2005) A hybrid regularized inversion methodology for highly parameterized environmental models. *Water Resources Research*, 41: doi: 10.1029/2005WR003995. issn: 0043-1397.

Tschritter, C., Rawlinson, Z.J., Barrell, D.J.A., Alcaraz, S. (2015) Three-dimensional geological model of Environment Southland's area of interest for freshwater management. *GNS Science Consultancy Report*, 2015/123.

United States Geological Survey (2017). MODFLOW and Related Programs. https://water.usgs.gov/ogw/MODFLOW/

WASY-DHI (2017a) FEFLOW. https://www.mikepoweredbydhi.com/products/FEFLOW

WASY-DHI (2017b) MIKE SHE. https://www.mikepoweredbydhi.com/products/mike-she

Appendix D Delta Shell

Intended Purpose

Problem types

To assist the development and implementation of freshwater policy and management, there is increasing need to integrate and operate multiple model types, using a range of different data sources and types, and have models to exchange data with external models or visualization tools in a co-ordinated and integrated manner. Multiple efforts have been made at trying to develop frameworks and standards allowing the linking and exchange of data of a range of components by different vendors, such as OpenMi Gregersen, Dijsbers et al. (2007), ESMF (Collins, Theurich et al. 2005), and many more. Often, such frameworks mainly target the integration aspects (data exchange), with models running independently of one another.

Delta Shell is an interoperable integrated modelling environment and framework, designed with emphasis on hydrological modelling (i.e., hydrology, hydraulics, water quality, morphology). It comprises several core components to address the whole model application domain, such as scientific model data handling (e.g., data types, units of measure), GIS data handling (e.g., support of OGC simple feature and coverage standards), data storage (project and file storage), spatial data visualisation (maps, graphs, tables), and interoperable modelling (e.g., OpenMI).

Models

Delta Shell provides specific support for DELTARES[7]' hydrological models, such as SOBEK 3, Delft 3D Flexible Mesh (FM) and others. However, by way of custom plug-ins, Delta Shell can cater for a range of model-specific domains, including but not limited to hydrology, water quality, morphology, climate and more, with full integration across spatiotemporal scales and resolution (Figures 1 and 2). Delta Shell also provides specialized libraries required to describe all mathematical, physical, and other aspects of data types used by the models, GIS domains, data storage with import and export in multi-dimensional file formats (NetCDF, HDF and others), relational databases and more. It provides a graphical user interface where all required components can be visualized and analysed (Donchyts and Jagers 2010).

[7] DELTARES is an independent Dutch institute for applied research in the field of water and subsurface, with main focus is on deltas, coastal regions and river basins.

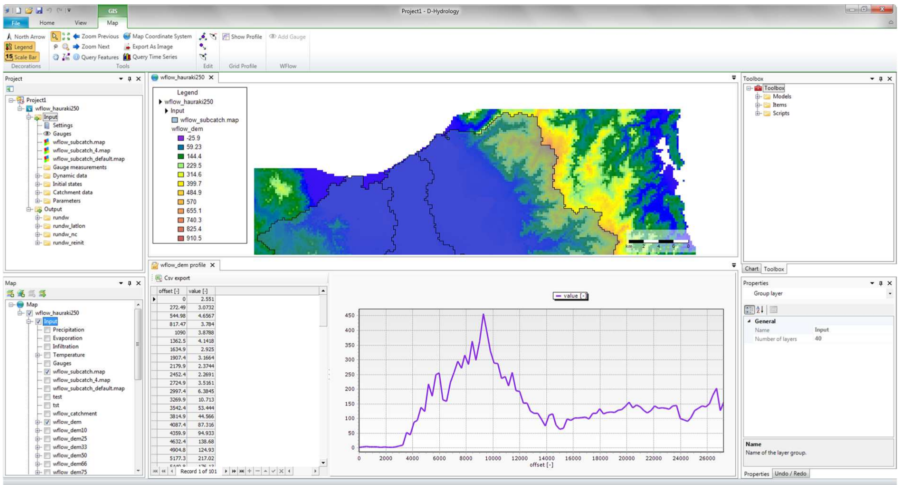

Figure 1: Hydrological models embedded in Delta Shell.

Institutional Support

Delta Shell is being developed as part of the Next Generation Hydro Software (NGHS) project by DELTARES. The development is staged and different modules of the project are released according to a predefined schedule (https://publicwiki.Deltares.nl/display/nghs/Development). It can be expected that DELTARES will continue the development and maintenance of the software beyond the end of this project, as it is their replacement platform for the predecessor FEWS (Flood Early Warning System).

Data

To extend the functionality of the Delta Shell platform beyond its native datatypes, the user(s) should implement an interface called IDataProvider (Donchyts and Jagers 2010). Delta Shell subsequently searches all implementations of that interface in all plug-ins at start-up and shows any data types provided by them to the user in such a way that any custom entity defined, such as time series, rasters, tables and more. Delta Shell also provides its own native default data types which can then be shared between plug-ins, and thus be used in the project along with all other data types.

Technical aspects of integration

Model coupling

Delta Shell supports three interfaces for model coupling

- Delta Shell internal IModel interface (plug-in).
- Basic Model Interface (BMI); enables model coupling of most computational Delta. Shell cores outside the GUI, e.g., to be used on clusters.
- OpenMI 1.4 and 2.0 (OGC standard).

Data import and export

Import

Delta Shell provides access to vector data (i.e., *.shp, *.mdb) and raster data (i.e., *.asc, *.bil, *.tif, *.iff, *.map). It also provides dedicated import functions for time-dependent grids, time series (i.e., *.csv), and NetCDF Regular 2D Grid.

Export

Delta Shell offers support to save project settings and model data in a hybrid SQLite database and NetCDF data format.

GIS functionality

Delta Shell supports the display and editing of model data (i.e., Hydro Region: basin, network) on top of a background raster map. Upon import of vector features, the user has to select which element (e.g., channels, bridge, culvert, etc.) of a Hydro Region the respective feature represents. The core functionality is implemented based on a range of open-source GIS libraries, including GeoAPI.NET, NetTopologySuite, Proj.NET, SharpMap and others for lightweight GIS applications, and includes support for geographic projections. For more functionality, IFeature, Coverages, Network Library and additional advanced mapping functionality has been embedded into Delta Shell. Additional functionality can be provided by custom-made plug-ins.

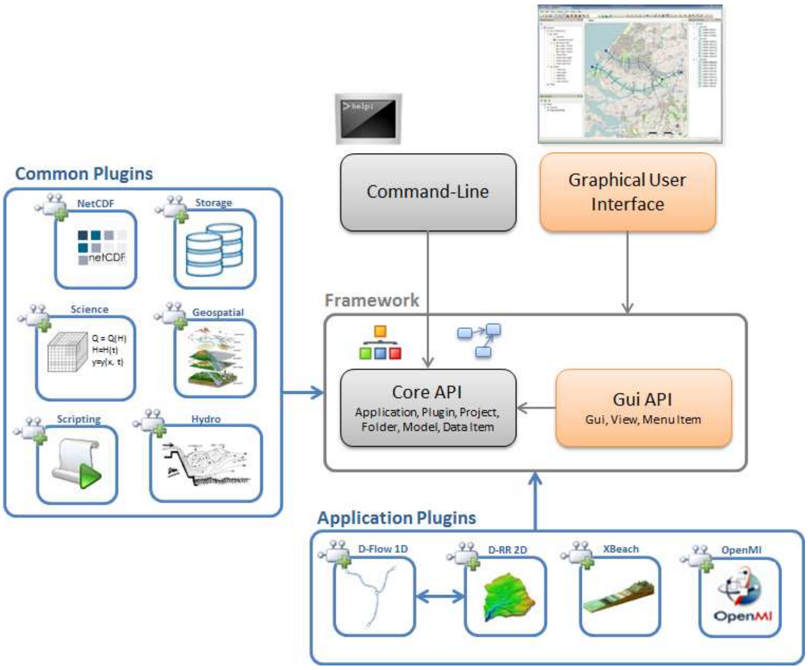

Figure 2: **Delta Shell schematic overview framework.** Source: https://oss.deltares.nl/web/delta-shell/develop.

Project storage

Delta Shell uses a hybrid data store of mixed SQLite and NetCDF file format for storing projects on the file system. Based on the specific type of data, it decides where to store values of the objects used in the projects. While many objects are stored in relational databases using NHibernate object-relational mapping libraries, some are stored in NetCDF with only meta-information stored in the relational database. Here, NetCDF file names always remain in sync with the corresponding records in the database. An XML file providing the database mapping is required when a plug-in introduces a new data type.

Visualisation and User Interaction/Interface

Delta Shell provides a GUI for displaying and editing model data (e.g., meshes) as well as for model coupling. Additionally, an interactive command line application is available for running configured and linked (coupled) models.

Visualisation in Delta Shell can be customised to the needs of the user(s). It may be that an organisation has a range of users with different needs for visualisation of different data. This can be managed by having one system, with multiple levels of user-group interfaces. For technical staff, they may want to dig into the details, including the scripting, data storage, management, implement new routines and analysis. Their screen may look like the screenshot below (Figure 3).

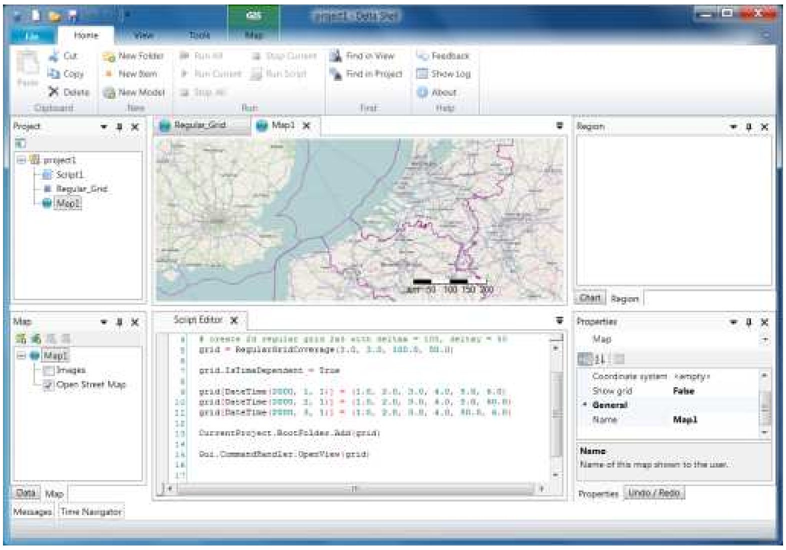

Figure 3: One means of visualisation in Delta Shell. Source: http://oss.Deltares.nl/image/image_gallery?img_id=345503&t=1383298647214.

For middle management, this level of detail may be of less interest, and for top management, the overall situation may be what is of interest (Figure 4). By having multiple user group access, the visualisation and operations that can be performed can be controlled.

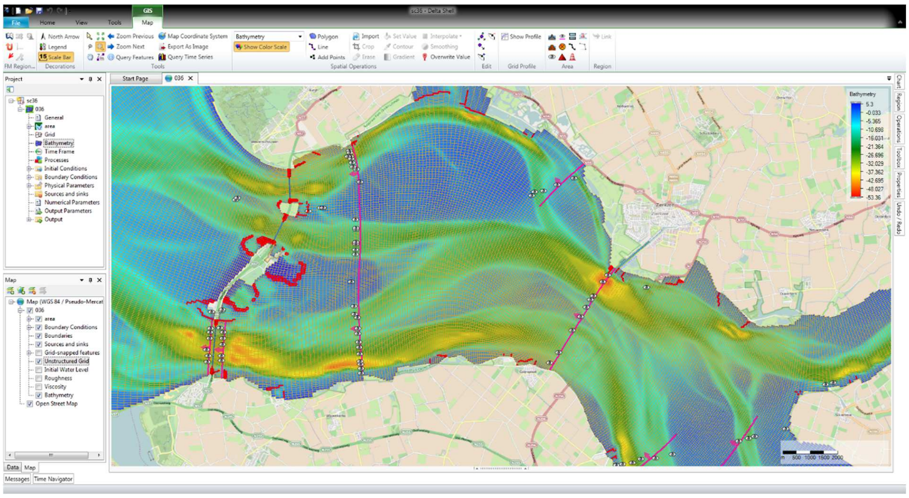

Figure 4: **Visualisation of results in Delta Shell.** Source: https://www.Deltares.nl/app/uploads/2015/10/Delft3D-Flexible-Mesh-Eastern-Scheldt-Scaloost-model2.png.

Metadata, scenario management, provenance, auditability

Delta Shell users can embed Rich Text Format data and documents in projects to describe the relevant data types in the project. This can include descriptive metadata, structural metadata, and administrative metadata.

For scenario management, users can implement multiple scenarios through the project tree. Setting up those scenarios is accommodated by the Python scripting toolbox, with which the users can easily and accurately set up scenarios in a reproducible fashion, run the scenarios, and store the results.

Auditability: The development of DS follows the DQMS (DELTARES Quality Management System) for software development. This includes, but is not restricted to, the use of versioning systems, issue tracking system, documented code reviews, V-model test approach, continuous integration environments which can all be tracked and presented at any time. Because the system runs by scripts, it is repeatable and easily documented.

Underlying language and computing requirements

Delta Shell has been written in C#, and is currently limited to use on MS Windows systems.

Uncertainty methods and calibration

Delta Shell currently offers both a Python scripting toolbox environment in which uses can freely experiment with their own or open source available optimisation and / or uncertainty analysis codes. Additionally, Delta Shell models may be run in OpenDA for calibration and parameter estimation. However, it is not clear whether that functionality is restricted to Delta Shell internal models (i.e., based on the IModel interface) or whether it is also available for all parts of a coupled model using BMI or OpenMI.

Management of IP and proprietary material

At the time of writing, Delta Shell is closed source software and the most recent freely downloadable and usable version was released in 2013. However, according to the Deltares website, the development follows a defined schedule until 2018 (s. institutional support and https://oss.Deltares.nl/en/) and in 2017 the Delta Shell framework is to be released under an open source software licence.

Proprietary model support: as long as a proprietary model is setup to support any of the Delta Shell-supported coupling interfaces (i.e., internal plug-in, BMI, or OpenMI) on the given host operating system, it should be possible to include them in an integrated modelling exercise.

Quality assurance

Versioning is implemented by using SVN. Unit and integration testing is done within the IDE of the software developers as well as on the continuous integration server. System testing is done manually by carrying out test scripts by human testers, as well as dedicated testers within the development team. Verification testing is carried out on the continuous integration server by running regression tests against predefined validated output data. Acceptation tests are carried out before the release of a new software produce, based on DS in close collaboration with the client. Internal reviews are enclosed within the DQMS and involved software code reviews within the software development process and documents reviewing process for both design and user documentation. External reviews are carried out by an external party in case a client asks to do so.

Experience with Use of the Framework

Does it meet the intended purpose?

In DairyNZ, Delta Shell serves as the primary platform for hydrology and water quality models. The intention is to expand on the current setup, which contains the models for Hauraki Plains in Northland and Waituna in Southland (Figure 5), to include water quality models across dairy catchments throughout the country. The platform provides a flexible, user friendly interface to operate models and run scenario analysis for water quality mitigation measures. This provides support for DairyNZ’s work with dairy farmers and the various regional councils that are involved in limit setting and plan change exercises.

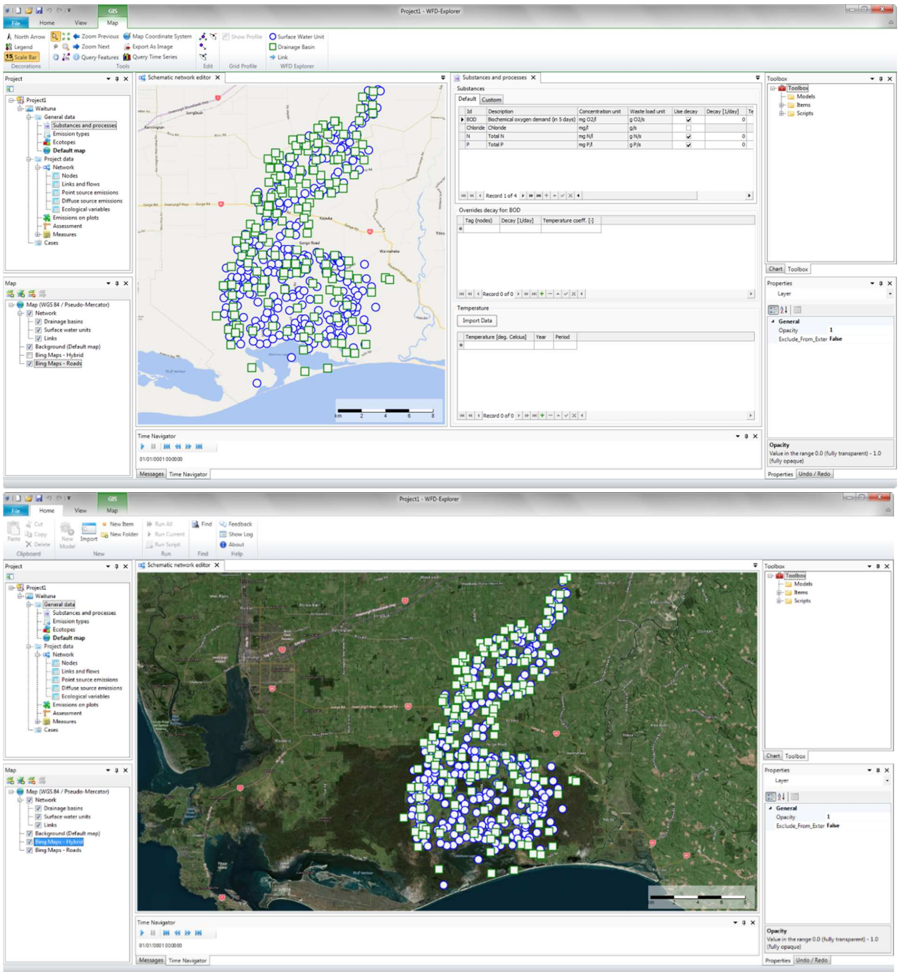

Figure 5: Waituna models in Delta Shell.

The following section gives examples of uses of Delta Shell within NZ, including some shortcomings of the framework and personal and end-user experiences.

NIWA scientists have been working with the Deltares DFLOWFM (Flexible mesh hydrodynamics) and DWAQ (Water quality) models in Delta Shell over the past 18 months. They are currently using the DFLOWFM and DWAQ models on three projects. These projects are: Manukau Harbour nutrient modelling study, Whanganui waste water dilution study and Queen Charlotte Sounds hydrodynamic modelling. The client feedback NIWA have had on the new software has been positive, especially on the GIS appearance and layout of Delta Shell and DFLOWFM software.

DairyNZ uses Delta Shell as the platform for their Waituna and Haruaki Plains water quality models. Expert support was provided by DELTARES for setting up the models. The experience in running the model was generally positive due to the very visual and neat interface, the tidy file storage, the layout of the boundary conditions, and the visualisation on the bathymetry grid. Even when run at a high resolution the model is quick to run. Challenges included the best way to approach grid development, and the frequent crashing of RGFGRID. The capacity to create higher resolution gridding in shallower regions would also be beneficial.

eCOAST (a NZ-based marine and freshwater consultancy), used Delta Shell for a simple two-dimensional modelling of currents and sea level in an embayment in Fiji. While they found the interface very nice and a big improvement on the Delft3D interface, the model frequently crashed, and the crossing of the dateline in the case study also caused problems.

Hopefully more modules will be released online in the coming year (coupled waves, sediment transport etc.).

Barriers to use

Currently, only an early version of Delta Shell is freely available and the source code is not available. However, if Delta Shell is going to be released as open source, as planned, IP barriers would be removed. Furthermore, in its available version, Delta Shell specifically targets the hydro domain. Data import and export and its general GIS functionality is limited. Again, this might be attributed to the age of the available test version.

Flexibility, extensibility and adaptability, and Ease of use in particular applications

Extending the system to new models or additional models is a straight-forward procedure with only some need for programming experience. However, the development of custom plug-ins requires advanced programming skills and significant effort and resources. Scenario options, or ranges, can be configured, and then accessed and executed via web interfaces. Delta Shell’s common components, i.e., geospatial, science, core (application), GUI, and Hydro, are extendable through custom-developed plug-ins. This offers a flexible way of extending the core framework without having to change the source code of the common components themselves.

Suitability for the needs of the OLW programme

Delta Shell is a potential candidate because of its hydro domain-driven design and built-in capabilities for hydrological modelling. Furthermore, it is one of the few frameworks that supports GUI-based coupling of standards for coupling (BMI and OpenMI) compliant models, one of the prime objectives of the programme.

In the past six months, newer releases of both Delta Shell and DFLOWFM have improved the” stability” of both tools. These improvements have increased the potential for use for commercial applications. However, there is a lack of good and complete documentation for the models and plug-ins that Delta Shell can support.

References

Collins, N., G. Theurich, G., DeLuca, C., Suarez, M., Trayanov, A., Bakaji, V., Li, P.,Yang, W., Hill. C., da Silva, A. (2005) Design and implementation of components in the earth system modelling framework. *International Journal of High Performance Computing Applications*, 19(3): 341–350.

Deltares (2017) Delft3D Flexible Mesh Suite – 1D/2D/3D Modelling suite for integral water solutions. *Delta Shell Draft User Manual Version 1.2,* Revision 50392.

Deltares (2017) Delft3D Flexible Mesh – Product description – Content description. *Delta Shell Technical Reference Manual Version 0.2,* Revision 46248.

Donchyts, G., Jagers, B. (2010) Delta Shell - an open modelling environment. *International Environmental Modelling and Software Societ*y, Ottawa, Canada, iEMSs.

Gregersen, J. B., Dijsbers, P.J.A., Westen, S.J.P. (2007) OpenMI: Open modelling interface." *Journal of Hydro informatics,* 9(3): 175–191.

Appendix E eWater Source

Intended Purpose

Problem types

The Source catchment modelling platform was developed by eWater in Australia, which was a partnership of government agencies, researchers and consultants and now operates on a consortium basis. The Source platform has core functionality in hydrology, pollutant generation and water management rules [8].

Source was developed to provide a flexible structure for hydrological modelling. It can be customised via plug-ins, which provide flexibility in adapting the model to a variety of water resource issues, such as integration with other models. At the same time, the standard distribution of Source comes with a number of hydrologic models to manage different applications for model use (the operating types are – Catchments, Rivers and Operations).

Models

Source represents river systems as a series of sub-catchments that are connected to nodes, which are then interconnected with links. Nodes are locations where flow or constituents enter a stream network, reporting locations, or stream confluences. Links, which define river reaches, act to store water and to route or process water and constituents passing between nodes.

Functional units (FUs) are areas within a sub-catchment that have similar behaviour in terms of runoff and/or constituent generation. The FUs are typically classified as unique combinations of the sub-catchment attributes (such as farming systems and soil types). Contaminant loading from each FU are provided by contaminant generation models built into the software and can also use external inputs from models such as OVERSEER [9], SPASMO, APSIM or a similar process based generation model representing that activity at the functional unit scale.

Source allows for different sub-models to be selected to provide the functions for rainfall runoff and drainage generation within each FU at time steps from sub daily through to annual. A rainfall runoff model that has been applied in several cases in New Zealand is the Soil Moisture Water Balance Model (SMWBM, Williamson Water Advisory), a daily conceptual rainfall-runoff model. The model takes inputs of daily rainfall and PET (which is provided as a time series for each sub-catchment for each day of the simulation, and can be obtained from NIWA's VCSN model) and physical parameters that control the movement of water within the model. Each FU therefore produces a daily time series of surface runoff to the stream and a daily time series of drainage to a conceptual groundwater store (one for each FU).

Each FU therefore produces time series of surface runoff to the stream via quick flow (surface runoff) and baseflow (via groundwater discharge). Consultants have recently developed modules for dynamic exchanges of groundwater; a plug-in version is now in the publicly available version of Source.

Constituents (for example, nutrients) can be exported from each FU in each sub-catchment using representative concentrations for storm events, dry-weather flow. Decay within the groundwater store is also accommodated.

[8] (Welsh, et al., 2012)
[9] (Wheeler et al., 2013)

In some cases, Source has been coupled loosely to other models, the coupled models providing a data series for flow of constituents with GLEAMS (NZTA, Jacobs), SedNet (HARC), APSIM (Ecological, Wairakei Estate), OVERSEER (HortNZ, Jacobs), and MODFLOW (Jacobs, Ecological, WWA). There are working examples of dynamic coupling between MIKE11 (Jacobs), MODFLOW and Source for recharge expressions, However, these have only been trialled in a test environment (Jacobs 2014).

The newly developed groundwater interaction plug-in[10] helps to predict the exchange of water and a contaminant (nitrogen) between rivers and the underlying groundwater systems. It determines the exchange flux of water between a river and the underlying aquifer for each link of Source at each time-step. The estimated flux accounts for interactions between groundwater and surface water along the entire length of the link. The direction of the flux can either be from the river to aquifer or vice versa, that is, the river either loses water to the groundwater system or it gains water from the groundwater system.

These can be either losses from the river to the groundwater or gains from the groundwater to the river. The fluxes are represented as a total volumetric loss or gain in a river reach (represented by a link) for each time-step in the Source platform. This new functionality allows for improved prediction of baseflow.

Data

The Source model data is can be provided as:

- Time series (e.g., daily time-step gauging station data).
- Polygons, polylines or points (GIS Visualisation layers).
- Imagery (Aerial and satellite imagery 3).
- Grid (Digital Elevation Model or Climate data-series).
- Point (Gauging station or point source nodes).
- Polyline (stream network).

The Source platform allows for a variety of data formats[11]

Institutional Support

Source has a wiki including a user guide[12] and technical documentation. There is also a training programme[13], which has been run in NZ several times.

Support is also available through the eWater Online community[14] and there is an annual Source conference.

[10] https://wiki.ewater.org.au/display/SD37/Groundwater+interaction+module

[11] https://wiki.ewater.org.au/display/SD41/Data+file+formats

[12] https://wiki.ewater.org.au/display/SD41/Source+User+Guide+4.1

[13] http://ewater.org.au/products/ewater-source/source-training/

[14] https://wiki.ewater.org.au/display/SC/Forum

Technical aspects of integration

Model coupling

Source has different models for runoff generation, constituent generation, filtering, in-stream water quantity and in-stream water quality. These are accessed through the command screen menu for the component models or plug-ins .dll's from the Plug-in manager menu.

With regard to model coupling – there are three ways that this can work.

1. A model can be re-coded to operate in Source as a plug-in, so that it effectively becomes an integral part of the model software. The SMWBM is an example of this – an externally coded model was re-coded in C# Visual Studio .NET so that it could be run within Source.

2. Source can produce a time series of outputs, which are then read by an external model, for example a hydraulic model like MIKE-11.

3. Dynamic coupling, where for example Source runs for a number of time steps, it feeds outputs into an external model and waits (e.g., MODFLOW) which then runs for a number of time-steps before feeding outputs back into Source – a cycle which is then repeated over and over again.

Functions

Source allows for functions to be developed to produce synthetic data or process data series as a built-in process calculation. This allows for arithmetic expressions to be introduced to a node, link or component model output.

Using functions is one of several ways of adding data in a Source model. Used in conjunction with variables, they provide a robust and flexible way of data input. There are six types of variables available in the Source platform:

1. Pattern variable – creates a dataset of repeating time-dependent values (e.g., daily or monthly pattern). Value used is based on the current simulation date and time.

2. Time series variable – uses a time series data source as input. Value used is based on the current simulation date and time.

3. Piecewise linear variable – creates a lookup table that interpolates a Y value based on an X value.

4. Bilinear variable – similar to a piecewise linear variable, but allows you to look up and interpolate between values in two dimensions.

5. Modelled variable – uses a model output, e.g., the downstream flow of an inflow node.

6. Context variable – for applying the same function in a number of locations that use modelled variables.

Visualisation

Time series outputs model elements such as links, sub-catchments, and functional units can be viewed graphically in charts, numerically in tables or as summary statistics, and they can also be filtered and manipulated using transforms. With custom charts, you can also view and statistically analyse multiple results within and between runs, compare results to external data sources and automatically update with the latest results each time a model is run.

User interface and web interface

Source has a dedicated GUI for data formatting, model development, simulation, and result viewing. These interfaces make enable the user to change model inputs to represent water resource scenarios and extract maps and statistical analyses for displaying outcomes.

One of the eWater originators has developed a development manager called 'Veneer', which is a system for linking Source to other applications, including web browsers, scripting tools and other graphical user interface based tools. Veneer works by hosting a HTTP server within the main Source application and providing access to key data via a RESTful API using JSON encoded data. Veneer can be used to:

1. Build customised reports and visualisations for Source models, which can be run live, alongside the model, or packaged for later publication to the web.

2. Build tailored Decision Support Systems with custom, HTML based user interfaces.

3. Perform ad-hoc scripting tasks while the Source user interface is open, using a tool like R or Python.

Veneer itself runs as a plug-in to Source and sets up a local service that makes some of the model data available to be consumed by another program. The other program will typically be a web browser and the data exchange formats are chosen to be easy to consume using Javascript visualisation libraries.

Metadata, scenario management, provenance, auditability

The Source framework facilitates setting up scenarios, which are stored as a model file and can be attached to the base model.

Once run, every scenario generates a discrete result set. Scenario results can be examined and analysed using the Results Manager, which opens when a run completes. All data and models are stored locally and the operator is responsible for the data provenance and associated data integrity.

The Veneer plug-in enables Python-based scripting-based access to Source functions, which enables models to be set up, run and visualised in a consistent and repeatable fashion.

Underlying language and computing requirements

Source is built using TIME, a general-purpose modelling system in the .net environment (vb.net and C# are supported).

The Invisible Modelling Environment (TIME) is a software development framework for creating, testing and delivering hydrological and environmental simulation models and as a foundation for plug-ins for Source[15]. TIME includes support for the representation, management and visualisation of a variety of data types and underpins a range of spatial and temporal modelling systems including several eWater models.

Plug-ins are coded in Visual Studio .Net – usually either C# or VB. They do access specified "access points" into the Source software, via calling of specific software objects in the Source programme. Source was built using the TIME libraries and hence it is quite common for plug-ins for Source to call

[15] (J.M. Rahman, S.P. Seaton, J-M. Perraud, H. Hotham, D.I. Verrelli, and J.R. Coleman; 2003)

software objects from the TIME.dll, although it is also very normal for plug-ins to call software objects from other .dll's that are included with Source.

This requires all machines to operate in a windows operating environment.

Uncertainty methods and calibration

Source has a tool for automated calibration using a 'calibration wizard'. For most model calibrations, a manual approach is required for the first cut for flow and constituents.

There is, however, an optimisation tool called 'Insight' which provides a method for objective function and scenario optimisation. This allows for more efficient evaluation of model options than the traditional manual trial and error approach that is often used.

The main aim of Insight is to optimise decision rules for multiple objectives. The framework allows optimisation across single or multiple scenarios. The optimisation tool enables a more thorough examination of potential planning scenarios and the resulting trade-offs between desired outcomes.

Source models can also be run through a command-line or scripts, enabling the use of third-party calibration and programmes such as PEST.

Management of IP and proprietary material

The eWater toolkit is controlled source software for the free versions. All users, including public release users, get access to the full library of plug-ins. There is a licence fee for the full version which provides full access to the code and wider user support. Government institutions are given free access to the Source software licence.

Quality assurance

There is a formal release programme with an open source beta version available to review changes and determine bugs as part of the eWater community. Notes are available for the production release detailing changes and 'software highlights'.

Experience with Use of the Framework

Does it meet the intended purpose?

The eWater Source platform has been used across several catchments in New Zealand, these are summarised in Table 1. The table shows the catchment for which the application was made, the organisation undertaking the modelling, the variables modelled, the component models and the modelling purpose.

Table 1: **Summary of eWater Source applications in New Zealand.**

Location	Owner(s)	Constituents	Models[1].	Status
Whangamarino Wetland	Dept. of Conservation	Concentrations and loads for E-Coli, TSS, TP, TN, DIN, DRP, Flow, level	SMWBM, MIKE11	Calibrated and used for hearing
Tukituki River	HortNZ, Consortium of Primary Sector	Concentrations and loads for E-Coli, TN, DIN, NO3, TP, DRP, Flow	SWMBM, OVERSEER	Calibrated and used for hearing and validation for TRIM2 by HBRC
Selwyn Waihora	CPW, Primary Sector Gp	Concentrations and loads for E-Coli, TN, NO3, TP, DRP, Flow, groundwater concentrations for TN	groundwater Network model, SMWBM, OVERSEER	Calibrated and used for hearing and CPW allocation management
Waipaoa River	GDC, HortNZ, FED, Wi Tu Trust	Concentrations and loads for E-Coli, TN, NO3, TP, DRP, TSS, Flow	SMWBM, OVERSEER, SPASMO	Calibrated and used for hearing
Ruamāhanga River	GWRC	Concentrations and loads for E-Coli, TN, TSS, DIN, NO3, TP, DRP, Flow	MODFLOW SMWBM OVERSEER TopNet MIKE11	Calibrated and used for allocation setting and NPS FW
Heretaunga Plains rivers	HBRC	Concentrations and loads for E-Coli, TN, TSS, DIN, NO3, TP, DRP, Flow	MODFLOW SMWBM OVERSEER	Calibrated and used for allocation setting and NPS FW
Ruahuwai Catchment, Waikato River	Wairakei Estate	Concentrations and loads for E-Coli, TN, Chlorophyll A, Clarity, Turbidity, NO3, TP, DRP, Flow	MODFLOW SMWBM APSIM	Calibrated and used for allocation setting and NPS FW
Rangitaiki River	BoP RC	Concentrations and loads for E-Coli, TN, TSS, DIN, NO3, TP, DRP, Flow	MODFLOW SMWBM APSIM	Calibrated and used for allocation setting and NPS FW
Kaituna River	BoP RC	Concentrations and loads for E-Coli, TN, TSS, DIN, NO3, TP, DRP, Flow	MODFLOW SMWBM APSIM	Calibrated and used for allocation setting and NPS FW

[1] Source does not incorporate all these models, but has been linked to their outputs or served as inputs to those models.

Barriers to use

The main barriers for use of Source in NZ is the availability of resources and a local community of users. This has improved recently with multiple parties undertaking modelling work. The other barrier has been a lack of regional data to harness the water quality modelling elements in the model.

Source requires buy-in to the consortium to access source code and access to advanced features. This could be a barrier to adoption of the system widely in New Zealand.

Some of the key models that have been used with Source in New Zealand are held by consultants, and are not available as community plug-ins. This includes SMWBM and the network groundwater model.

There are other problems with adapting traditional thinking around hydrological solutions to the mass balance approach in the Source catchments model. This requires allocation or routing prior to the gauge locations for each catchment. The benefit is that it contributes to the hydrological understanding between groundwater and surface water and can help early diagnostics for integrations.

An important constraint of the Source modelling platform is that it is primarily a hydrological model with a node-link spatial construct. Where complex flood inundation is involved, a hydrodynamic model is required to quantify the flow dynamics in a hydraulic sense (velocities, volumes, inundation depths). This can be overcome for some management situations where the flow can be coupled with a MIKE11 model, however, if dedicated flood predictions are required a hydrodynamic model is probably better suited. Similarly, models for contaminant sources that work on fine grid would not be suitable for source.

Flexibility, extensibility and adaptability

Source is not a single hydrological model. It is a range of models and tools that have been incorporated into a single flexible adaptable environment. Source can easily be customised by users to address specific local problems, or can be pre-configured for modelling situations.

In practice this has meant that solutions can be developed along the broad understanding of the catchment hydrology, data availability and companion models. In recent models this has allowed APSIM to provide temporal series for base flow concentrations for each functional unit within a sub-catchment. This allows a series for historical data to be imported into the base model and is a considerable advancement for modelling lag effects. This solution can be incorporated as a 'Plug-in' with a dynamic call in for each model step across the model's intrinsic data or pre-calculated outside the model and imported as a data series.

Ease of use in particular applications

Source operates through a high-level graphical interface, more like a manager's "workbench" or "dashboard" than a traditional hydrologic or hydraulic model. The interface is deceptively simple as it begins with a schematic view of the watershed or river which is built by dragging icons from a palette to create a conceptual view. At this high-level scale, simple scenario model can be used to quickly and easily explore with stakeholders the practical way a river catchment operates, without significant data requirements.

Within the standard menu; rainfall-runoff models, pollutant export models, sediment/nutrient retention (filter) models, link routing models, link decay models, nodal process models, are all selected from a wide range of alternatives and can be assigned uniformly to a sub-catchment or globally across the model. Alternatively, the component models can be selected for each functional unit within the entire model domain.

There is a large body of Source users, (ca. 3000), a small number of which (<30 in Australasia) are developing Plug-ins. There is a constantly growing library of Plug-in solutions.

Suitability for the needs of the OLW programme

Source provides the opportunity to dynamically couple a wide range of component models, which represents a significant change for catchment management New Zealand where regulatory approaches for decision making around allocation and trading could be integrated into a single decision support framework. Such capabilities are relevant to the needs of OLW.

While Source does not include production, economic, or social models, which are part of the OLW programme needs, they can be used in conjunction as the inputs/outputs are easily transferable.

The node-link structure and hydrological basis of Source would probably be a limitation in relation to the class of model that can be incorporated into Source.

Web Resources

http://ewater.org.au/products/ewater-source/
http://ewater.org.au/casestudies/
Other resources are available to registered users including a user forum and user guidance.

References

http://www.malleecma.vic.gov.au/resources/salinity/7.-key-tools-strengths-and-weaknesses

Australia and India partner to manage water for scarcity. Australian Government - Department of Foreign Affairs and Trade. Australian Government. 26 April 2013.

Aid Program Performance Report 2012−13 Australian Mekong Water Resources Program (PDF). Australian Government - Department of Foreign Affairs and Trade. Australian Government.

Hydrological Modelling using eWater's Source. Australian Government - Murray Darling Basin Authority. Australian Government.

Great Barrier Reef Report Card 2011 (PDF) Reef Water Quality Protection Plan. Australian Government, Queensland Government.

Environmental water delivery: Namoi River. Department of Environment. Commonwealth Environmental Water Office, Australian Government.

Appendix F GeoJModelBuilder (GJMB)

GJMB is a geoprocessing workflow tool that integrates OGC (Open Geospatial Consortium) Sensor Web services, geoprocessing services, and OpenMI-compliant model components (see separate review on OpenMI).

Intended Purpose

GJMB provides a way to run models, conduct processing of spatial data, and integrate sensor observations, accessed through the internet and in a structured workflow. GJMB implements modelling as a service (MaaS) based on OGC based sensor and geoprocessing web services as well as OpenMI compliant (local) models and other 'scriptable' local processing resources. The integration of the OGC Sensor Web additionally provides automated monitoring and notification capabilities, which can be configured to trigger geoprocessing and/or modelling components, e.g., to provide particular analysis and evaluation of events defined from monitoring. In short, GMB provides:

- Integration of sensor web and geoprocessing functionality to enable event-driven processing and modelling.
- Event-driven workflow enactment.
- Integrated modelling and environmental monitoring.

Models

In general, the system handles geospatial processing algorithms and integrated models implemented as OGC Web Processing Service and OpenMI (v1.4) compliant models respectively.

Models implemented in the system and described in (Peng et al. 2015) are:

- GeoPW (about 200 geoprocessing services on the web, mostly based on the GRASS GIS functionality).
- Hargreaves runoff model (OpenMI component).
- TOPMODEL runoff model (OpenMI component).
- SOSReader senser observations processor (OpenMI component).

These components were used to implement, for example, a workflow for a watershed runoff simulation. Additionally, GJMB supports the following OGC Sensor Web Enablement services for combining environmental monitoring with modelling:

- Sensor Observation Service (SOS).
- Sensor Event Service (SES).
- Sensor Planning Service (SPS).
- Web Notification Service (WNS).

Data

See the *Models* and *Data import and export section.*

Institutional Support

GJMB is developed by the GeoPW team from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS), Wuhan University, China.

Technical aspects of integration

Model coupling

GJMB uses output/input data linkages for coupling OGC web services, and locally available processing resources, such as OpenMI compliant models and other locally installed software accessible via scripts.

Data import and export

GJMB provides the following OGC web services for data provision, which provides a rich set of tools for working with spatial data:

- Web Feature Service (WFS).
- Web Coverage Service (WCS).
- Web Processing Service (WPS).

Additionally, GJMB enables the integration of locally available data that is loaded into the geoprocessing workflow.

Visualisation

Input data, results, sensors, and geoprocessing services can be visualised using the integrated virtual globe (NASA World Wind) component. Interactive workflow development is carried out using a visual workflow designer.

User interface

GJMB provides a GUI for workflow development and visualisation of imagery and sensors. It is not explicitly stated how workflows can be saved and whether they can be executed outside the GUI (i.e., command line interface).

Metadata, scenario management, provenance, auditability

There are no specific facilities for scenarios and auditability, but the workflows aid reproducibility and some documentation for auditing and provenance.

Underlying language and computing requirements

GJMB is implemented in Java. It uses the Java OpenMI v1.4 implementation. It supports at least OGC WPS versions 0.4 and 1.0. It is not clear which exact services and service versions are supported from the OGC Sensor Web.

Uncertainty methods and calibration

GJMB does not have any specific capabilities for calibration and uncertainty, although such components could be used if the user provides them.

Management of IP and proprietary material

GJMB is open source software licenced under GPL v2.0.

Quality assurance

No information available for the workflow engine software itself. However, the system provides a configurable Quality of Service (QoS) control for the web services linked in the workflow.

Experience with Use of the System

At present the system has had little use.

Does it meet the intended purpose?

GJMB provides an array of tools to meet the intended purpose, but it is too soon to evaluate success of this new system.

Barriers to use

The user guide for version 2 is not in English, which would be problematic for use in the OLW programme.

Flexibility, extensibility and adaptability

GJMB provides considerable flexibility through the capability of building workflows based on standardised data and processing services. Since the software is open source, it is in general adaptable but adaptation requires significant Java programming skills.

Ease of use in particular applications

There have not been enough applications yet to judge the ease of use of this system. Since it makes use of standards and common tools and software patterns, we expect that it would be fairly easy to for a professional programmer to develop an application using this system.

Suitability for the needs of the OLW programme

GJMB is interesting for the OLW programme because it supports the linkage of standardised web-based data services, such as WFS, WCS, and SOS, with standardised geoprocessing services (WPS) and locally installed OpenMI compliant models and other geoprocessing resources accessible via scripts. However, the software is new with few applications, and there may be difficulties accessing suitable support, which may limit its suitability for widespread use across the programme.

References

Yue, P., Zhang, M., Tan, Z. (2015) A geoprocessing workflow system for environmental monitoring and integrated modelling. *Environmental Modelling & Sofware*, 69: 128–140.

Zhang, M., Bu, X., Yue, P. (2017) GeoJModelBuilder: an open source geoprocessing workflow tool. *Open Geospatial Data, Software and Standards*, 2:8. DOI 10.1186/s40965-017-0022-7.

Appendix G Land Use Management Support System (LUMASS)

Intended Purpose

Problem types

LUMASS is designed to provide support for two high level aspects of land management: i) land use impact assessment and ii) spatial planning. The former aspect is supported by LUMASS's spatial system dynamics modelling framework, and the latter is supported by LUMASS' spatial optimisation framework.

Models

LUMASS facilitates the development of raster-based spatially explicit system dynamics models. The framework is agnostic of model domain and has been used to implement the soil erosion model SedNetNZ (Dymond et al. 2016), the bio-physical forest growth model CenW (Kirschbaum 1999), cellular automata-based spread models (e.g., DaisyWorld, Neuwirth et al. 2015), as well as general image processing tasks (i.e., pre-processing for modelling).

Additionally, LUMASS supports modelling optimal allocation scenarios of arbitrary resources to fixed spatial units. This has been used for modelling land-use configuration scenarios (Herzig et al. 20W16), the assessment of resource-use efficiency of land-use at the catchment scale (Herzig et al. submitted), and the selection of efficient soil erosion mitigation strategies within budgetary constraints at the farm scale (Northland & Auckland RC).

Data

LUMASS supports multi-band raster data (import, export, 2D display, processing, creation and editing of raster attribute tables), polygon vector data (import, export, display, attribute table editing). Additionally, LUMASS supports the import and stereo display of 3D point clouds (*.csv files).

For reading and writing 2D raster and vector data, LUMASS uses the GDAL/OGR library. 3D raster data may be read from and written to an open source rasdaman database (rasdaman.org), including its required metadata. This enables the accessibility of image data via rasdaman OGC WCS, WCPS, and WPS.

Raster attribute tables and stand-alone tables (*.dbf, *.csv, *.xls) can be imported into SQLite databases and can be accessed by the SQLProcessor model component to perform SQL-based database operations as part of a LUMASS model.

Institutional Support

LUMASS development is supported by Landcare Research SSIF funding, MBIE contestable funding, and commercial contracts (e.g., Northland & Auckland RC).

Technical aspects of integration

Model coupling

LUMASS models are built from atomic processing components. Individual components represent mathematical equations and fundamental algorithms working on multi-dimensional arrays. The components are concatenated via their respective output and input data to processing pipelines. These can be grouped and nested to build a complex hierarchical processing workflow, which can operate on multiple time scales. Repetitive and conditional execution of components is achieved by way of dynamic component properties which may be changed at runtime depending on intermediary modelling results. Feedback is enabled through shared (sequential) access of array data (i.e., a stock variable in system dynamics terms) by different individual components.

In addition to internal LUMASS components, arbitrary external (non-interactive command line) applications may be integrated into the above described workflow to provide additional functionality. However, external components do not enjoy the benefits of the internal pipeline architecture, which provides sequential and multi-threaded processing of the data.

Data import and export

See the Data section.

Visualisation

LUMASS visualises vector and raster data in a 3-D display area, which supports two distinct interaction approaches for 2-D and 3-D data respectively. The (default) 2-D interaction approach provides typical 'GIS-functionality' for zooming, panning, and querying 2-D data, such as raster layers and polygon vector layers. The user may optionally display a scale bar and a coordinate frame. Point clouds (i.e., 3-D point data) may be vest viewed using the 3-D interaction approach and can be also rendered in different stereo modes for a true 3-D visual experience.

Tabular data such as raster and vector attribute tables, as well as a number of standalone data types (e.g., *.xls, *.sqlite, *.csv, *.dbf) is displayed in a tabular view, which also allows the user to manipulate the data.

User interface

LUMASS comes with a GUI-based desktop application, which provides an integrated modelling framework and a 3-D capable display area for the visual inspection of raster and vector data. Tabular data may be viewed and manipulated in dedicated table views.

The modelling framework allows the user to build complex hierarchical spatial system dynamics models and geoprocessing workflows using an interactive visual (i.e., icon-based) programming environment. Properties of individual model components can be displayed and edited interactively. Additionally, users can freely place and format text on the modelling workbench to document and annotate model components. Users may also create their own toolbars and toolbar buttons or tools to run specific LUMASS models. Beside the desktop application, LUMASS also provides a simple command line application for running LUMASS models or optimisation scenarios in cluster or server environments.

Metadata, scenario management, provenance, auditability

LUMASS models (and each component individually) can be saved as an XML representation on disk. This representation comprises all individual model parameters including references to the input and output data. The format is used internally to import LUMASS components into another LUMASS model or to recreate the entire model logic while loading a model from disk.

Underlying language and computing requirements

LUMASS and all of its underlying libraries are written in C++. CMAKE build tools are used to facilitate cross-platform compilation. LUMASS has been built and run on 64bit Windows and Linux systems. Binaries are provided for 64bit Windows.

Uncertainty methods and calibration

LUMASS models and the optimisation component support uncertainty and sensitivity assessments by facilitating Monte Carlo simulations. LUMASS does not provide parameter calibration or estimation functionality.

Management of IP and proprietary material

LUMASS is open source software and available on bitbucket https://bitbucket.org/landcareresearch/LUMASS

Quality assurance

LUMASS is tested internally. A formal internal or external review processes is not established.

Experience with Use of the System

Does it meet the intended purpose?

Several models from different science domains, i.e., soil erosion modelling (SedNetNZ, Dymond et al. 2016), eco-physiological modelling (CenW, Kirschbaum 1999), ecosystem services assessment (Herzig et al. 2016), and image processing have been implemented as proof-of-concept for the modelling framework. Several functional weaknesses have been identified along the way and consecutively addressed to improve the overall capabilities of the framework. Furthermore, the interactive visual model development environment has been found to be useful to facilitate the integration of domain specific experts without advanced programming skills, into the development process and the application of a particular model.

Barriers to use

Documentation and sample data is currently under development to reduce the 'lack of documentation' barrier.

Flexibility, extensibility and adaptability

The main functionality provided by LUMASS, i.e., spatial optimisation and raster-based spatial system dynamics modelling are implemented as generic frameworks. Both frameworks can be applied to different science domains and applications.

Extending the LUMASS software requires advanced C++ knowledge and significant effort by new developers.

Ease of use in particular applications

See other material in this section.

Suitability for the needs of the OLW programme

LUMASS does not target the integration of large complex external models. Rather, models are constructed by visually assembling in a Graphical User Interface (GUI) individual components that provide atomic functionality ranging from general mathematical equations working on image pixels and pixel neighbourhoods to specific functions targeting land-management, e.g., cost-distance surfaces. These components, representing either a process or data, can be further grouped into aggregated components to build complex hierarchical models. External executables (e.g., a command line application or script) can be integrated into the model workflow as long as they are non-interactive and input and output data can be read and written by LUMASS respectively (i.e., loose coupling). However, these components do not enjoy the benefits of the internal pipeline architecture, which provides sequential and multi-threaded processing of the data. To extend the LUMASS modelling framework, developers have to implement additional components according to the LUMASS API and possess advanced C++ programming skills.

LUMASS provides some useful features, which align well with the programme objectives. On the science domain level, it provides capabilities in the areas of soil erosion modelling, ecosystem services assessment, as well as trade-off analysis and optimal resource allocation. Furthermore, it may potentially provide access to further land domain models, such as forest growth, weed spread, etc. On the technical level, LUMASS provides processing capabilities for multi-dimensional datasets, including sequential and multi-threaded processing of large data sets. The LUMASS engine command line application enables the execution of LUMASS models and optimisation scenarios on cluster environments for multi-scenario simulations. Furthermore, because LUMASS provides a computational engine that can execute any model developed within the LUMASS framework and because LUMASS models can be completely described by a single XML file (with appropriate parameters set), it enables 'bring the model to the data' architectures.

Two important shortcomings need to be addressed to make LUMASS fit for the programme requirements: i) the implementation of unit tests for individual components, and ii) the adaptation of the model framework to support an agreed-on interoperability standard (e.g., BMI, OpenMI). The latter would make any model developed within the LUMASS framework interoperable according to this standard. Hence, it would allow LUMASS models (i.e., a LUMASS engine library running a LUMASS model) to be coupled with other standard compliant components as part of an integrated model. In turn, LUMASS would also be able to integrate external standard compliant models into its own workflow.

The lack of unit testing is going to be addressed successively in the next development phase of LUMASS. This phase will integrate LUMASS step by step into the 'production process' of selected Landcare science outputs and goes hand in hand with training of targeted staff. Increased effort will be undertaken to roll-out LUMASS externally to grow the user-base and attract external developers.

References

Dymond, J.R., Herzig, A., Basher, L., Betts, H.D., Marden, M., Phillips, C.J., Ausseil, A-G.E., Palmer, D., Clark, M., Roygard, J. (2016) Development of a New Zealand SedNet model for assessment of catchment-wide soil-conservation works. *Geomorphology*, 257: 85—93.

Herzig, A., Rutledge, D. (2013) Integrated Land Systems Modelling and Optimisation. In: Piantadosi, J., Anderssen, R.S., Boland, J. (eds.) MODSIM2013, 20th International Congress on Modelling and Simulation. *Modelling and Simulation Society of Australia and New Zealand*, December 2013: 880-886. http://www.mssanz.org.au/modsim2013/C8/herzig.pdf

Herzig, A., Dymond, J., Ausseil, A-G.E. (2016) Exploring limits and trade-offs of irrigation and agricultural intensification scenarios in the Ruamāhanga catchment, New Zealand. *New Zealand Journal of Agricultural Research*, 59(3): 216—234.

Herzig, A. (2017) LUMASS – User Guide, Version 0.3. *Manaaki Whenua – Landcare Research New Zealand Ltd.*
https://bitbucket.org/landcareresearch/LUMASS/downloads/LUMASS_UserGuide_v0.3.pdf

Herzig, A. (2017) LUMASS – Spatial Optimisation How To, Version 1.1. *Manaaki Whenua – Landcare Research New Zealand Ltd.*

https://bitbucket.org/landcareresearch/LUMASS/downloads/OptimisationHowTo_1.1.zip

Herzig, A., Nguyen, T.T., Ausseil, A-G.E., Maharjan, G.R., Dymond, J.R., Arnhold, S., Koellner, T., Rutledge, D., Tenhunen, J. (submitted) Assessing Resource-Use Efficiency of Land Use. *Environmental Modelling & Software.*

Kirschbaum (1999) CenW, a forest growth model with linked carbon, energy, nutrient and water cycles. *Ecological Modelling*: 118: 17—59.

LUMASS website: https://bitbucket.org/landcareresearch/LUMASS

Neuwirth, C., Peck, A., Simonovic, S.P. (2015) Modeling structural change in spatial system dynamics: A Daisyworld example. *Environmental Modelling & Software,* 65: 30—40.

Appendix H OMS3 and CSIP

Intended Purpose

Problem types

OMS3[16] (Object Modelling System) is the third iteration of OMS which was developed and is being used primarily in the agro-environmental modelling space. The main focus is on hydrology, but the framework itself is more general and not limited to that knowledge domain. OMS is also one of the core components of CSIP (Cloud Services Integration Platform) which is a "*SoA (Service-oriented Architecture) implementation to offer a **Model-as-a-Service** framework, Application Programming Interface, deployment infrastructure, and service implementations for environmental modelling*".

Models

A number of hydrology models have already been set up set up using a range of hydrological and water quality model components including Thornthwaite Monthy Water Balance Model[17], Precipitation Runoff Modelling System[18] (PRMS/OMS), AgroEcoSystem-Watershed Model (AgES-W)[19], and JGrasstools. A number of hydrology models have already been set up with a number rousing a range of hydrological and water quality model components including Thornthwaite Monthy Water Balance Model Thornthwaite Monthy Water Balance Model[20], System Precipitation Runoff Modelling System[21] (PRMS/OMS), AgroEcoSystem-Watershed Model (AgES-W)AgroEcoSystem-Watershed Model (AgES-W)[22], and JGrasstools.

Additionally, many models are available through CSIP[23],[24].

Data

The API specifically handles elementary data types exchange between models (single values, arrays and tables) but does not otherwise require any specific complex data format. OMS3 handles geospatial data and time series.

Institutional Support

There is mature and ongoing support for OMS3 and CSIP, led by Colorado State University in collaboration with US Department of Agriculture (USDA). There is also a large base of users in Italy (University of Trento). OMS3 has not been adopted by a New Zealand organisation, but it was selected, trialled, and promoted for uptake in an Interoperable Freshwater Models (IFM) project[25].

Technical aspects of integration

Model coupling

Coupling is done by modifying the code of a model or by writing a wrapper around the code so that it conforms with the OMS3 API (Application Programming Interface). Once the API has been

[16] https://alm.engr.colostate.edu/cb/wiki/16961
[17] https://alm.engr.colostate.edu/cb/wiki/17027
[18] https://alm.engr.colostate.edu/cb/wiki/17003
[1919] https://alm.engr.colostate.edu/cb/wiki/17029
[20] https://alm.engr.colostate.edu/cb/wiki/17027
[21] https://alm.engr.colostate.edu/cb/wiki/17003
[2222] https://alm.engr.colostate.edu/cb/wiki/17029
[23] https://alm.engr.colostate.edu/cb/project/csip
[24] https://alm.engr.colostate.edu/cb/wiki/17106
[25] https://teamwork.niwa.co.nz/display/IFM/Framework+for+Interoperable+Freshwater+Models

implemented, simulations linking components are constructed by writing simulation scripts which control the data flow and sequencing of model components. Data to CISP deployed models is supplied in a JSON formatted structure.

Data import and export

Databases (spatially enabled or not), web service supplied data (spatial or not), etc. can all be accessed by writing the appropriate wrappers. There is also support for importing and exporting time series.

Visualisation

The OMS3 console, which is what is used to run the simulations, has basic graphing functionality. Third-party tools can be used for displaying graphical and spatial data, but this requires programming effort.

User interface

The OMS3 console provides simple access to framework core features such as simulation management (by editing the simulation files), output analysis, or documentation generation.

The simulation script provides some ability to provide metadata and repeatability, and metadata can be incorporated into source code and extracted, but there is no formal system for scenario management and auditability.

Underlying language and computing requirements

OMS3 is written in Java, and can be run on all computing platforms. In addition, the web functionality of CSIP enables access to model run on remote machines via web services.

Uncertainty methods and calibration

OMS3 does not have specific method for calibration or uncertainty, except that it allows for ensemble hydrological modelling (running a number of hydrological models and comparing the results).

Management of IP and proprietary material

OMS3 is Open source under LGPL 2.1, and the code is readily available.

The system does not have a means for managing proprietary data or model components, although the source code of individual components could be kept private and the web basis of CSIP would enable some protection of proprietary material.

Quality assurance

OMS is versioned software with professional software development support.

Experience with Use of the System

Does it meet the intended purpose?

OMS3 meets the purpose of the collaborators, and is used extensively by the USDA to deliver key models to end-users.

Barriers to use

No IP barriers.

Flexibility, extensibility and adaptability

OMS3 has the ability to combine models in a flexible and adaptable way as it was designed as a general interoperability framework.

Ease of incorporating new models depends on the state and nature of the target model. Fortran, C and C++ models are relatively easy, Windows models (C#, .NET) can be challenging and require more specialist programming skills.

Ease of use in particular applications

In the IFM project, some simple models were set up successfully for OMS3. It appeared that setting up complex models would require significant programming input. Applications for USDA involve programmers support from Colorado State University.

Suitability for the needs of the OLW programme

OMS3 could be a good option for the OLW programme. Most applications for OMS have been developed in the agro-environmental domain, which is relevant to the programme needs, but no economic components have been added to our knowledge. The framework is already fairly mature and is constantly being developed and is used and improved by a range of national and international collaborators. The availability of the OMS driven CISP framework (also open source) is an additional plus, given the extended scope in nature and location of the models that may be target for inclusion in OLW. The OMS developers are very open to collaboration, as was found in the FIFM project, and this is an important aspect of successful adoption of a framework.

References

OMS website https://alm.engr.colostate.edu/cb/wiki/16961

Also, see list of publications on same website

Appendix I OpenMI

Intended Purpose

Problem types

OpenMI is not a framework per se, but rather a software component interface definition, which is now also an OGC (Open Geospatial Consortium) standard. That is, OpenMI it is an API with a set of clearly defined methods of communication between various software components, be it models, databases, or other types of tools (visualisation and analysis). It was initially developed to support interoperation of time-stepping models in Europe, especially hydrological and hydraulic models, but it is not restricted to this model domain.

Models

The API is domain agnostic; hence it can be used with to couple any two software components as needed. It is primarily aimed at static and dynamic simulation models.

It provides synchronous as well as asynchronous coupling, as well coupling between data represented on different grids and time domains.

A range of models has been modified to comply with OpenMI, most of them simulation models in the hydrological and hydrodynamics areas. A list of compliant models, projects and initiatives is maintained on the OpenMI website[26].

Data

OpenMI does not incorporate any particular data source, because it is a standard (a prescription for doing things) not an actual model. OpenMI allows for quantitative and descriptive data and spatial elements (points, lines, polygons, multi-polygons); elements of the same type can be combined into an element set (e.g., river network, grid). The main data types are elementary data types for each spatial element, although the standard can be extended to allow for more complex objects.

Institutional Support

OpenMI is supported by the open and not-for-profit OpenMI Association, which was established once initial project funding ended. OpenMI is still considered to be a relevant and important standard, which is accommodated by interoperability frameworks such as OMS and Delta Shell (they can use OpenMI-compliant models). The Pipistrelle implementation of OpenMI was developed by HR Wallingford, and is open source.

There have been two implementations of the standard in .Net and Java respectively for version 1.4, and one in Java for version 2.0. A Java Software Development Kit (SDK) and a Graphical User Interface (GUI) for version 2.0 are available for download (dated 2010) and is part of the FluidEarth project. Although there was some recent activity (code updates) on the Java SDK software download website[27], activity around these existing implementations has been very low. Development of OpenMI implementations does not appear to be a centralized effort.

[26] http://www.openmi.org/openmi-around-the-world
[27] http://sourceforge.net/projects/fluidearth

Technical aspects of integration

Model coupling

Coupling is described by the API. Both ends of the coupled pair must implement the API. Each component has an update method (for dynamic models). Pull-driven sequencing of model component runs at each update step, whereby one component does not update until all the components upon which it depends have updated. The update sequence for a time step is initiated by a trigger component. Various message-passing strategies to deal with complex inter-dependencies such as loops.

Version 2 allows for loops with sequencing of components specified externally, but this is somewhat experimental.

Data import and export

No specific import and export methods are provided for by the standard.

Visualisation

OpenMI is a standard, so it does not have an interface. However, a GUI is provided by the Pipistrelle implementations.

Metadata, scenario management, provenance, auditability

OpenMI does not have any relevant features.

Underlying language and computing requirements

The Java implementation can be run anywhere, whereas .NET implementation can only be run effectively in Microsoft Windows.

Uncertainty methods and calibration

OpenMI does not address uncertainty or calibration

Management of IP and proprietary material

The two existing implementations (Java and .NET) are open source, so is Pipistrelle.

Quality assurance

OpenMI has been adopted as an OGC standard, which has high requirements of specification. However, OpenMI does not have any particular features for versioning, testing and peer review.

Experience with Use of the System

Does it meet the intended purpose?

Several models have been coupled successfully using the OpenMI API, including coupling of models from different science providers in Europe. Also, the standard has been adopted by the OGC standards authority, and serves as one of the few standards for model coupling in the hydrologic and hydraulic simulation space. So, the standard has met its original goal. However, there seems to be waning enthusiasm for active development of conversion of models to be OpenMI compliant.

Barriers to use

There are few barriers to use. However, models must be adapted to conform to the standard. Also, since the emphasis is primarily on the mechanics of coupling, the user must prepare their own model interface and utilities, which would entail considerable development. Coupling of models using OpenMI entails a penalty in terms of performance if information is exchanged between components frequently, which may be a barrier for parties interested using OpenMI of computationally intensive models.

Flexibility, extensibility and adaptability

OpenMI can accommodate a range of simulation models, and the standard allows for extension (for example, adding different data types).

Ease of use in particular applications

OpenMI has been used in a range of applications, especially the old 1.4 version (see http://www.openmi.org for references). It involved considerable effort to modify existing models to be compliant with the standard with full dynamic coupling.

Suitability for the needs of the OLW programme

The API is quite extensive but it all comes down to the actual implementation of the API. For the scope of this project it is best to consider existing frameworks and models implementing the API (e.g., Seamless, Fluid Earth, ESMF etc.).

References

The OpenMI website http://www.openmi.org includes a long list of references describing OpenMI and its applications.

Appendix J Catchment Land Use for Environmental Sustainability (CLUES)

Intended Purpose

Problem types

The development of CLUES (Woods et al. 2006; Elliott et al. 2016) was initiated by New Zealand government agencies with a goal of being able to rapidly identify the impacts of land use on water quality and socio-economic indicators, at catchment to national scale and with spatial resolution to REC sub-catchments. This was in order to inform policy and planning in New Zealand. Intended users are catchment modellers in universities, research institutes and regional and central government charged with freshwater management, especially at a planning level, although there has subsequently been some use by environmental consultants.

Models

CLUES has model components for: contaminant generation; accumulation and decay in a stream network and lakes; an estuary mixing model; and an economic indicator component. CLUES provides steady state, spatially distributed predictions of mean annual loads of total nitrogen, total phosphorus, sediment and *E. coli*, and concentration of nutrients in streams and rivers throughout New Zealand (268,000 km^2) with a sub-catchment resolution of 0.5 km^2. A number of pre-defined land uses are used. CLUES also incorporates and estuary mixing model to provide estimates potential nutrient concentrations for estuaries. The socio-economic indicators of CLUES are (farm economics, energy, greenhouse gas, and infrastructure revenue), which are derived from analysis of farm reporting, are applied to the land uses in the catchment to derive catchment-wide economic metrics.

CLUES has a fixed set of models. It uses a simplified version of OVERSEER Nutrient Budgets (Roberts and Watkins 2014) for pastoral nutrient losses, a meta-model of SPASMO(Rosen et al. 2004) soil-plant model results for nitrogen loss from horticulture and cropping, other source coefficients and decay coefficients from adaptations of the SPARROW catchment model (Alexander et al. 2004; Elliott et al. 2016), and simple estuary mixing models(Luketina 1998; Gillibrand et al. 2013). The socio-economic indicatory model is described in Harris et al. (2009).

Data

CLUES is based on polygons (sub-catchments and estuaries) and linear spatial features (stream reaches) connected into a network. Tabular data describing the spatial elements is stored in text files and spatial geodatabase tables.

Institutional Support

CLUES was initially established through central government and regional council funding, with science provision from NIWA, AgResearch, Landcare Research, Plant and Food Research, and Harris Consulting. NIWA is providing ongoing funding, with co-operation from AgResearch and OVERSEER Ltd.

Technical aspects of integration

Model coupling

CLUES calls various sub-models from vb.net, OVERSEER, and Sparrow, with components called as dll's from the main code, while other model components are written within the main vb.net program. The components are called in a fixed sequence, with use of text files for major data exchange (apart from those entailed in the dll's). There is no need for feedback between model components. All the component models are already within CLUES; adding further components involves modification of the main monolithic code.

Data import and export

CLUES uses ESRI file geodatabases for spatial data and associated tables, with some text-based tabular data and parameter files. No particular data standards are used, except that internal representations of the stream network adopt the ArcHydro data model and library. All data is stored locally on the user's machine.

Visualisation

Model inputs and outputs are visualised as 2-D catchment maps, without a temporal component. The maps can easily be modified since CLUES is built within ArcGIS. Attributes of model elements (sub-catchments, streams, and estuaries) can be viewed as data tables of selected elements.

User interface

CLUES has tools for selecting the study catchment area, scenario management, land use and mitigation setup, run control and scenario management, output display, and data or map export. A custom control panel format is used rather than a conventional ArcGIS toolbar or toolbox to make the system suitable for users with only a basic GIS background. Users familiar with GIS can also use the core set of ArcGIS features for data editing and geo-visualisation.

Metadata, scenario management, provenance, auditability

The system has facilities for setting up scenarios, which are stored as part of the ESRI map document and associated data. All data and models are stored locally, and the user is responsible for information on the particular model application such as scenario definition, imported data, and mitigation setup. The distributed models follow a basic versioning numbering system.

Underlying language and computing requirements

The model is written in vb.net, and is limited to PC's with ArgGIS installed.

Uncertainty methods and calibration

CLUES does not have any representation of uncertainty and does not have a calibration facility. The Sparrow model, which provides some key coefficients, is calibrated externally to CLUES.

Management of IP and proprietary material

ArcGIS is proprietary and closed-source. The CLUES code is also closed-source, but there is no charge for use by Regional Councils and researchers.

Quality assurance

There are no formal systems for testing or external review, but new versions are tested internally before release, the underlying research is published, and the CLUES model has been documented in a peer-reviewed publication (Elliott et al. 2017). The OVERSEER dll provided by AgResearch is difficult to validate, as not all the assumptions are exposed, placing reliance on testing by AgResearch.

Experience with Use of the Framework

Does it meet the intended purpose?

Meetings held in March 2012 with various stakeholders identified that the intended purpose of CLUES has been met. Since then, CLUES has been extended to incorporate further elements and refinements in response to needs of users. The range of successful applications provides evidence of suitable application and uptake of the model. The geospatial aspects have been useful in terms of communicating environmental risks, differences in sources, and responses to scenarios in a way that can be easily understood by a wide audience. This last point is pertinent to the National Policy Statement for Freshwater Management (NPS-FM) which has provisions for public participation in catchment planning and management. The model is suitable for use by regional council staff with moderate GIS expertise, and has been used in this manner in some cases, although applications are most often contracted out to external parties. The model calculations are fast enough to enable interactive modelling. As an indication, run time is typically a minute for a 1000 km2 catchment. Also, default input data are provided at national scale, which makes for rapid initial model setup.

At the regional scale, the simple structure of the model and visual representation of results have been useful for communicating the implications of land use change to decision-makers, identification of hot-spots of contaminant generation, and linking of stream water quality to upstream contributing areas (Semadeni-Davies et al. 2009). At the national scale, model results for future scenarios provided useful information on effects of land intensification and dairy expansion for an independent government commissioner(Parliamentary Commissioner for the Environment 2014). Mitigation assessment in Southland highlighted the importance of accounting for the potential and current extent of mitigation measures to temper previously high expectations of future water quality improvements(Semadeni-Davies and Elliott 2011; Hughes et al. 2013).

It was originally hoped that groundwater could be added to CLUES, but this has not been achieved. Also, it was hoped that the system could 'zoom in' to sub-farm scale resolution, but this was not done apart from displaying N leaching grids; this limits the ability of the system to link from sub-farm scale to catchment scale.

Barriers to use

CLUES is closed-source with code managed by NIWA, so that other parties cannot modify or extend the models or interface themselves. CLUES rely on a licence to use OVERSEER. The land-use layer provided with CLUES is not being updated, due to licence constraints associated with AgriBase or alternative national land use layers. The user can, however, import their own land use, so this limitation is not important in practice. CLUES rely on ArcGIS, which most of the intended users already own. However, this presents a barrier for smaller consultants or community groups. Agreement with NIWA is required for use of CLUES in consulting projects, although this has not been a serious constraint.

Flexibility, extensibility and adaptability

Considerable effort is required to add or modify model components, requiring detailed coding by a programmer with close knowledge of the overall programme. Similarly, addition or modification of interface components requires programming. Changing from the original VBA version to VB.Net, as required by changes to ArcGIS, entailed considerable effort. Effort is also required to update data sources and convert them to the required format for CLUES. Such improvements and maintenance are generally provided through NIWA funding without external support, which represents a risk to ongoing development.

Ease of use in particular applications

CLUES comes with default input data and is rapid to run, which has been useful for applications with default inputs. It is also fairly straight forward to run mitigation scenarios, although separate pre-processing of inputs are required to set up complex mitigation scenarios. Running the model at national scale (Parliamentary Commissioner for the Environment 2014) proved time-consuming as each region needs to be run separately and the results combined as separate post-processing, and this limited responsiveness to scenarios posed by the PCE. Setting up new land-use at national scale is time-consuming, as this requires a sequence of spatial pre-processing steps that the CLUES itself does not cater for, which introduced delays. In several applications, users requested modifications to CLUES, which entails new programming. Examples include refining the model spatially (Waitaki for ECAN), adding forest harvesting terms (Tasman District Council) and adding other model components (e.g., sediment modelling for Waikato Regional Council, Northland Regional Council and Auckland Council),

Changes to OVERSEER versions has proved troublesome, because the resulting changes to source terms have implications for other model coefficients, requiring re-calibration (which takes place externally to CLUES).

Suitability for the needs of the OLW programme

CLUES has limited applicability to the objectives of the interoperable models project in OLW programme. The underlying code for integration is not open source and is proprietary. The interface is tightly coupled to the calculation engines, and the engine components are not modularised in an interoperable fashion. Only steady-state predictions are provided, whereas the programme calls for the capability for dynamic simulations. There is no clear linkage from individual properties (or sub-property resolution) to the catchment scale. However, some of the principles and components of CLUES are relevant. For example, linking OVERSEER into the catchment scale, a map-based foundation, use of national spatial datasets, and rapid run-times using simplified budget-based catchment models and economic indicators are desirable aspects of CLUES that could be translated into a future interoperability system.

Web Resources

https://www.niwa.co.nz/freshwater-and-estuaries/our-services/catchment-modelling/clues-%E2%80%93-catchment-land-use-for-environmental-sustainability-model

References

Alexander, R.B., Smith, R.A., Schwarz, G.E. (2004) Estimates of diffuse pollution sources in surface waters of the United States using a spatially referenced watershed model. *Water Science and Technology*, 49(3): 1–10.

Elliott, A.H., Semadeni-Davies, A.F., Shankar, U., Zeldis, J.R., Wheeler, D.M., Plew, D.R., Rys, G.J., Harris, S.R. (2016) A national-scale GIS-based system for modelling impacts of land use on water quality. *Environmental Modelling & Software*, 86: 131–144. http://dx.doi.org/10.1016/j.envsoft.2016.09.011

Gillibrand, P.A., Inall, M.E., Portilla, E., Tett, P. (2013) A box model of the seasonal exchange and mixing in Regions of Restricted Exchange: Application to two contrasting Scottish inlets. *Environmental Modelling & Software*, 43: 144–159. 10.1016/j.envsoft.2013.02.008

Harris, S., Elliott, S., McBride, G., Shankar, U., Semadeni-Davies, A., Quinn, J., Wheeler, D., Wedderburn, L., Hewitt, A., Gibb, R., Parfitt, R., Clothier, B., Green, S., Cacho, O., Dake, C., Rys, G. (2009) Integrated assessment of the environmental, economic and social impacts of land use change using a GIS format - the CLUES model. *New Zealand Agricultural and Resource Economics Conference, August 2009.*

Hughes, A., Semadeni-Davies, A., Tanner, C. (2013) Nutrient and sediment attenuation potential of wetlands in Southland and South Otago dairying areas. *National Institute of Water and Atmospheric Research.*

Luketina, D. (1998) Simple tidal prism models revisited. *Estuarine Coastal and Shelf Science*, 46(1): 77-84. DOI 10.1006/ecss.1997.0235

Parliamentary Commissioner for the Environment (2014) *Water quality in New Zealand: Land use and nutrient pollution*, Wellington, New Zealand. http://www.pce.parliament.nz/publications/all-publications/water-quality-in-new-zealand-land-use-and-nutrient-pollution

Roberts, A., Watkins, N. (2014) One nutrient budget to rule them all–the OVERSEER® best practice data input standards. *Nutrient Management for the Farm, Catchment and Community, Occasional Report,* No. 27. Fertilizer and Lime Research Centre, Massey University Palmerston North, New Zealand.

Rosen, M.R., Reeves, R.R., Green, S., Clothier, B., Ironside, N. (2004) Prediction of groundwater nitrate contamination after closure of an unlined sheep feedlot. *Vadose Zone Journal*, 3(3): 990–1006.

Semadeni-Davies, A., Elliott, S. (2011) Application of CLUES to the Mataura Catchment. Impacts of land use and farm mitigation practices on nutrients. Prepared for Environment Southland. *NIWA Client Report:* HAM2011-018, National Institute of Water and Atmospheric Research.

Semadeni-Davies, A., Elliott, S., McBride, G., Shankar, U. (2009) Using CLUES to identify impact catchments - Waikato River Catchment Pilot Study. *NIWA Client Report* HAM2009-101. National Institute of Water and Atmospheric Research.

Woods, R., Elliott, S., Shankar, U., Bidwell, V., Harris, S., Wheeler, D., Clothier, B., Green, S., Hewitt, A., Gibb, R., Parfitt, R. (2006) The CLUES Project: Predicting the Effects of Land-use on Water Quality – Stage II. *NIWA Client Report* HAM2006-096. National Institute of Water and Atmospheric Research.

Appendix K Land Allocation and Management (LAM) model

Intended Purpose

Problem types

The goal of the LAM model is to provide an integrated description of a biophysical and economic system, such that the profit, production and environmental implications of alternate management options can be assessed (Doole, 2015). The model has been applied extensively at the catchment level, across both Australia (e.g., Doole et al., 2013) and New Zealand (e.g., Howard et al., 2013; Bermeo et al., 2015; Parsons et al., 2015; Doole et al., 2013, 2015, 2016, 2017a, b). The output has been used to inform policy development by stakeholder groups, regional environmental managers and central government across both countries.

Models

LAM is an equilibrium model that studies land use and land management across different spatial zones within a mixed catchment incorporating both rural and urban land uses. It is based around the partitioning of a catchment into diverse spatial units; these can vary according to land use, slope, rainfall, soil type, and so on. It has been integrated extensively with output from other models; this includes output from nutrient budgeting (such as OVERSEER) (e.g., Howard et al., 2013), attenuation data from flow-network models (e.g., CLUES) and sediment budgeting (such as NZEEM) (e.g., Doole, 2015, 2016). Other models have also been incorporated into it. One example is its inclusion of four contaminant load models in the context of the Healthy Rivers Wai Ora process (Doole et al., 2015, 2016). Output from the LAM model has also been used in other models. One example is the use of LAM output to assess policy impacts at the district, regional and national levels (Market Economics, 2015; McDonald and Doole, 2016).

Data

A chief benefit of the LAM model is its flexibility (Parsons et al., 2015). All versions have a common overall structure, but this is manipulated to suit each application based on the available information and the objectives of the analysis (Doole, 2015). The LAM model can incorporate data from most other types of models, though some manipulation may be required. Examples from previous work are illustrative. Prior application typically involves the integration of land use information from GIS databases, use of profit information from FARMAX or farm consultants, contaminant loss information from biophysical models (e.g., CLUES, NZEEM, OVERSEER), and attenuation data from flow-network models (e.g., CLUES).

Institutional Support

The LAM model is currently implemented by Waikato University and the Department of Economic Development, Jobs, Transport, and Resources in Victoria, Australia. There is no one version of the LAM model; the model is substantially adapted each time it is employed to suit the unique circumstances surrounding each application.

Technical aspects of integration

Model coupling

The LAM model is built within Microsoft Excel or GAMS. Inputs from other models are either defined as exogenous coefficients or the external models themselves are incorporated into the LAM model (Doole, 2015). Other models could be closely coupled with the LAM model, such that the diverse models are maintained in different modelling environments and update one another dynamically. The details of how this could best be achieved depend on the resources at hand and the objectives of the analysis.

Data import and export

The LAM model can incorporate data from most other types of model, though some manipulation may be required. Data is easily imported from, or exported to, text or Microsoft Excel files. The import and export of data is controlled through coding in the GAMS language.

Visualisation

Model output from the LAM model is generally presented using R, MATLAB, and/or Microsoft Excel.

User interface

The user interacts with all parts of the LAM model through coding in either Microsoft Excel or GAMS. No user interface is provided, given that the LAM model is principally a research tool and the unique circumstances of each application prevent the use of a fixed model structure.

Metadata, scenario management, provenance, auditability

Scenario management, provenance, and auditability are maintained through an established process for careful version control. This is centred on the long-term use of open-source version control software, Apache Subversion.

Underlying language and computing requirements

The model is coded in GAMS. It can typically run on any modern computer, given the efficiency of the matrix formulation used within the model (Doole, 2015).

Uncertainty methods and calibration

Uncertainty analysis is typically carried out according to structured sensitivity analysis (Pannell, 1997) and Monte Carlo simulation (Doole and Romera, 2014).

Management of IP and proprietary material

Details around model structure are freely available (e.g., Doole, 2015). The data used within the model is often subject to confidentiality restrictions.

Quality assurance

Careful version control is used to prevent errors arising from iterative model development. Internal and external review are used within each application to help verify and validate model inputs and outputs.

Experience with Use of the System

Does it meet the intended purpose?

The LAM model has been extensively applied to inform regional policy (Parsons et al., 2015; Doole et al., 2015, 2016, 2017a, McDonald and Doole, 2016) and national policy (Doole, 2013; Bermeo et al., 2015; Doole et al., 2017b).

Barriers to use

LAM is subject to its platform's (i.e., GAMS) shortcomings. Unlike some of its competitors (e.g., AIMMS), GAMS does not offer the feature to easily build a GUI. Hence, a potential LAM user would need to have good knowledge of mathematical programming, complex coding and the solvers to be used. GAMS, through some of its solvers, provides limited parallel-computing capabilities that could fully utilise the new multi-core architecture offered by most computers these days. Hence, depending on the amount of variables, parameters, constraints and type of problem (defining the solver to be used), models can be subject to long run times. However, in practice, this is rare due to the wide-scale use of general principles of good model development (e.g., provision of good starting points, use of smooth functions, limiting nonlinearity). Such short-comings constrains the spatial and temporal detail that can be included in the model. It is common practice in mathematical programming models (i.e., the broad category to which the LAM model belongs) to discretise space and time as much as possible to make the model tractable and to achieve faster solution times. This can become a barrier when the spatial correlations are critical to the problem in hand (e. g. clustering land uses to achieve higher economies of scale) or when the temporal dimension is important (e. g. forestry's erosion and leaching rates differ across time).

Flexibility, extensibility and adaptability

A key feature of the LAM model is its flexibility (Doole, 2015). It is easy to adapt and this is demonstrated in the diversity of model formulations that have been constructed in recent years. For example, Howard et al. (2013) focused on nitrogen in the Canterbury region, while Doole et al. (2016) studied nitrogen, phosphorus, sediment, and microbial losses in the Waikato River catchment. Additionally, Bermeo et al. (2015) used the LAM model to study water-quantity issues, with little attention on contaminant loss.

Ease of use in particular applications

The model has been applied numerous times across diverse circumstances within projects that are subject to tight resource constraints. Other models could be closely coupled with the LAM model, such that the diverse models are maintained in different modelling environments and update one another dynamically. The details of how this could best be achieved depend on the resources at hand and the objectives of the analysis. Nevertheless, it is likely to be a complicated process subject to complex coding and long run times.

Suitability for the needs of the OLW programme

LAM has the potential to be suitable to the programme's objectives. The main advantages offered are its tractability, transparency, and flexibility. From an economic point of view, LAM will be able to find the optimal (i.e., most profitable or least costly) combinations of land uses and mitigation strategies to reduce the impact exerted from erosion or leaching phenomena, at the block and catchment levels, which is among the Challenge's main objectives. The model's tractability and transparency advantages are the result of its wide use in peer-reviewed published material, which avoids dealing with IP restrictions around its source code.

Contrary to other similar models (e.g., NZFARM), the fact that the LAM's structure is freely available offers all of the usual advantages offered by open-source models: they can be modified, expanded and improved by any modeller with the right sets of skills. Hence, its shortcomings can be addressed depending on the problem in hand. For example, although LAM is a steady-state model that does not consider price responsiveness from varying supply-demand levels, the model could be modified to include this through the inclusion of empirical supply and demand functions. Furthermore, the fact that the LAM includes the time dimension simplistically can be improved by including a time-dependent path for some of its variables to make the model of a dynamic nature, e. g. tree productivity affecting cash flow, erosion and leaching phenomena.

However, some limitations are apparent. First, linking dynamic simulation models with such models is difficult where daily input data (e.g., climate) is used. Second, the best way to link the LAM model with output from other software is not clear *a priori*. Microsoft Excel is the most common method used, but when linking models, it is more likely that one of the established application programming interfaces would be best used. Last, models can be slow to run where they integrate data from many models that encompass a fine resolution along spatial and/or temporal domains.

References

Bermeo, S., Doole, G.J., Austin, D., Fenemor, A. (2016) Waimea Plains: economics of freshwater quantity management. *Contributed paper prepared for presentation at the 60th AARES Annual Conference,* Canberra, ACT, Australia, 2-5 February 2016.

Brooke, A., Kendrick, D., Meeraus, A., and Raman, R. (2017) *GAMS—A user's guide*, GAMS Development Corporation, Washington D. C.

Doole, G.J. (2013) *Evaluation of policies for water quality improvement in the Upper Waikato catchment*, University of Waikato client report, Hamilton.

Doole, G.J., Vigiak, O.V., Pannell, D.J., and Roberts, A.M. (2013) 'Cost-effective strategies to mitigate multiple pollutants in an agricultural catchment in North-Central Victoria, Australia'. *Australian Journal of Agricultural and Resource Economics,* 57: 441–460.

Doole, G.J., Romera, A.J. (2014) 'Cost-effective regulation of nonpoint emissions from pastoral agriculture: a stochastic analysis'. *Australian Journal of Agricultural and Resource Economics,* 58: 471–494.

Doole, G.J. (2015) 'A modelling model for determining cost-effective land allocation at the catchment level'. *Computers and Electronics in Agriculture,* 114: 221–230.

Doole, G.J., Elliott, A.H., and McDonald, G. (2015*) Economic evaluation of scenarios for water-quality improvement in the Waikato and Waipa River catchments: Assessment of first set of scenarios. Healthy Rivers Wai Ora report* HR/TLG/2015-2016/4.1, Hamilton.

Doole, G.J., Quinn, J.M., Wilcock, B.J., Hudson, N. (2016) Simulation of the proposed policy mix for the Healthy Rivers Wai Ora process. *Healthy Rivers Wai Ora report* HR/TLG/2016–2017/4.5, Hamilton.

Doole, G.J., Moleta, G., Barns, S. (2017a) Economic analysis of the re-allocation of nutrient entitlements in the Lake Rotorua catchment. *BOPRC Technical Report*, Tauranga.

Doole, G.J., Austin, D., Bermeo, S. (2017b) Economic evaluation of instruments for the management of irrigation water on the Waimea Plains. *Ministry for the Environment Technical Report*, Wellington.

Howard, S., Romera, A.J., and Doole, G.J. (2013) Selwyn-Waihora nitrogen loss reductions and allocation systems, *DairyNZ*, Hamilton.

Market Economics (2015) Economic impacts of Rotorua nitrogen reduction; district, regional, and national evaluation. *Market Economics,* Takapuna.

McDonald, G., Doole, G.J. (2016) Regional- and national-level economic impacts of the proposed Waikato Regional Plan Change No. 1—Waikato and Waipa River Catchments. *Healthy Rivers Wai Ora report* HR/TLG/2016-2017/4.5a, Hamilton.

Pannell, D.J. (1997) 'Sensitivity analysis of normative economic models: Theoretical model and practical strategies'. *Agricultural Economics,* 16: 139–152.

Parsons, O.J., Doole, G.J., Romera, A.J. (2015) On-farm effects of diverse allocation mechanisms in the Lake Rotorua catchment. *BOPRC/DairyNZ,* Hamilton.

Appendix L Land Utilisation and Capability Indicator (LUCI)

Intended Purpose

LUCI is an integrated framework considering how land use and management impact a range of landscape provisions including agricultural production, flood mitigation, water supply, greenhouse gas emissions, biodiversity and habitat protection, erosion, sediment and nutrient delivery to waterways. It is spatially explicit, working at the resolution of an input digital elevation model, typically input to be 5 by 5 m scale. It compares the services provided by the current utilisation of the landscape to estimates of its potential capability, and uses this information to identify areas where change might be beneficial, and where maintenance of the status quo might be desirable.

Problem types

LUCI was originally designed to be run in near-real time with farmers and other stakeholders to investigate multiple benefits provided by the land, to explicitly account for spatial configurations in the landscape so management change could be more efficiently targeted, and to be able to work with nationally available data where finer resolution data is not available. More specifically it was designed to investigate the potential of cumulative sub-field level changes to contribute to good outcomes at both farm and catchment scale. This focus continues, but as a consequence of the fast run times of the model combined with its fine spatial resolution, it is also used to support national scale predictions of consequences of national and local policies on environmental outcomes (x, y and z), and is being adapted to support international environmental accounting.

Models

Nitrogen, phosphorous, sediment, agricultural production. These are most developed to predict export of mass from terrestrial systems to freshwater bodies. Process and transport representations are more limited in fluvial and lake systems, and not suitable for modelling estuaries or to the coast. This method identifies nutrient sources and current or potential intercepting nutrient sinks.

Habitat suitability, habitat and/or species connectivity. Typically run at 5 by 5 m resolution.

Data

LUCI internal processing is in main based on raster/grid operations, although time series estimates of flow operate on polygon hydrological response units and stream reaches connected via a flow network and further reporting is calculated on user-specified (polygon) aggregate units. Parameters are stored in dbf files and xml files.

Institutional Support

LUCI is a second-generation extension and associated software implementation of the Polyscape framework described in Jackson et al. (2013), the ideas for which were an outcome of work by the United Kingdom’s Flood Risk Management Research Consortium (FRMRC), specifically involving Imperial College London, Nottingham University, the Centre for Ecology and Hydrology, and Bangor University. Seed funding to develop the Polyscape concept and software prototype were provided by Victoria University of Wellington (VUW), Bangor University (Wales) and a second FRMRC project phase. Central government and research councils in the UK have funded a range of completed and ongoing projects over the last five years, as have regional councils and Ravensdown within New

Zealand. Continuing funding for LUCI is provided through VUW, with significant in-kind personnel time input also from the UK's Centre for Ecology and Hydrology, Nottingham University and the University of Queensland.

Technical aspects of integration

Model coupling

LUCI is coded in Python, and has been written to be modular, with the intent of being able to both bring external models into the framework, and export internal models for use in other frameworks. However, this is not well tested in practice. Early proof of concept of ability to import external models written in other languages was successfully established by running models operating in R code and Fortran code via f2py and rpy2 within LUCI code, but since then no projects or other work have required such functionality. Internally, a range of coupling approaches are used. Some routines are fully coupled along with others that are loosely coupled or stand alone.

Data import and export

Almost all of the initial (and internal) data processing within LUCI uses ArcGIS tools, and as a consequence it implicitly supports the same raster and vector formats as ArcGIS. Others require external pre-processing. Outputs are written to user specified folders, and the user can specify output format as either a collection of shapefiles/ESRI grid rasters/dbf tables in a standard folder, or have these stored in a file geodatabase. Text files and xml files are also output providing metadata on runs. Pdf files summarising maps and other results are also output. When the desktop version of LUCI is used, all data is stored and processed locally on the user's machine. Geoprocessing servers upload user data, process operations on the server, and download output back to the user as a zipped folder with outputs as described above.

Visualisation

Maps and tables are loaded into ArcMap and also written out in a pdf report. Additional spider diagrams and time series are output for some modules. Results include maps at the native processing resolution (typically 5 by 5 m/DEM resolution), and more spatially aggregated maps and other output to help interpretation at larger scales. Default output is annual. Temporal components are developed for water flow and soil moisture accounting. In default mode, these run at hourly scale and are reported at daily scale but the code accepts any time resolution.

User interface

LUCI is delivered as a Python toolbox or as a geoprocessing service which operates from ArcToolbox. The toolbox includes pre-processing tools to load and reconcile inconsistencies in input data (rivers flowing uphill, soil and land cover data inconsistent about terrestrial/water interfaces, etc.), scenario generation tools, a range of single service tools exploring outcomes of land management interventions, trade off and optimisation tools, and a range of visualisation tools. A web interface is in development to allow users without GIS licenses and/or GIS operating experience to access results, but cost and licensing issues with input data need to be resolved before this could be made generally available.

Metadata, scenario management, provenance, auditability

LUCI is hosted in two github repositories (split so users can have full access to the main code base as appropriate, with a second hosting additional components where IP/development/third party agreements or data requires a higher level of protection). A numbered versioning system is in place. All changes to code are tracked via github processes and therefore are fully auditable. Metadata is generated during LUCI runs to keep inputs/warnings/provenance tracked.

Underlying language and computing requirements

The model is implemented in Python and has a strong dependency on ArcGIS, with most modules calling a number of ArcGIS tools via the Arcpy package. It is supported for both Windows and Linux/Unix systems, and can be installed either as a toolbox on Windows desktops, or called as a geoprocessing service from Victoria University of Wellington or AWS servers.

Uncertainty methods and calibration

LUCI has basic Monte Carlo implemented for exploration of uncertainty, which could also be used for internal calibration. However, no such internal calibration has yet taken place. Most LUCI models are parameterised from measured data or "expert judgement" related to soil, topography, and land cover combinations and results to date have not shown a need for further calibration. Pastoral N and P representations have been externally calibrated to a large database of OVERSEER results.

Management of IP and proprietary material

ArcGIS is proprietary and closed source. A governance board for LUCI is currently investigating how IP should be managed and the extent to which the framework will be open vs closed source. Currently open source access to the core framework is given to all partners in testing and development, with a subset of tools more carefully protected to respect third party agreements and data.

Quality assurance

A process for testing both development and release versions is in place, with an automatic test suite with a variety of different options/datasets running. Fine detail of the protocols and automatic test suites are evolving. A variety of recent peer review papers on LUCI exist, but the developers have not yet published a paper formally detailing the software as per a particular release version, which is increasingly considered necessary for best practice. As per CLUES, New Zealand N and P predictions have some reliance on OVERSEER, where not all assumptions are exposed.

Experience with Use of the Framework

Does it meet the intended purpose?

Yes

Barriers to use

Decisions on the ways in which the evolving LUCI framework will be delivered and maintained for operational use and the extent to which it will be open source and/or freely available are not finalised. A freely available geoprocessing service with the most established tools is about to be released, but there is limited server infrastructure to support processing so only a few users will be able to access the tool at any given time. LUCI N and P export for pastoral land cover is calibrated to OVERSEER, and version changes to OVERSEER in the future may affect compatibility of predictions between models. The model currently has the ability to read in farm management detail via provision of an OVERSEER output xml, but changes in the way OVERSEER is delivered in the future pose a risk to this.

Predictive capacity is a function of the quality of input data provided, and digital elevation, land cover and soils data are always necessary. Access to ArcGIS can be a barrier for some users. Use of LUCI requires either users to be able to access and operate ArcGIS and also obtain the necessary data and license agreements to provide required inputs, or licensing agreements to be in place allowing delivery of a web service. Costs and third party licensing issues therefore pose a further barrier to use.

Flexibility, extensibility and adaptability

Modification requires a good knowledge of Python. The framework was intended to be flexible, extendable and modular from the start of its development, but it is not yet mature or tested enough to confidently say it has achieved those criteria. Most external funding is directed to achieving specific outcomes so there are challenges in continuing the additional resources required to keep the framework flexible going forward; for the time being this is underwritten by internal funding from VUW and partners.

Ease of use in particular applications

LUCI has been developed with the intent of being easy to use and set up, and to have minimal data requirements, capable of operating using only nationally available data. However, recent changes in New Zealand to access data and data licensing requirements place an additional burden on users as they need to carry out a number of third party negotiations to obtain enough input data to run the model. It is fast running in the context of its fine spatial resolution, but there is an unavoidable computational overhead to this use.

Significant effort has been made to reduce the amount of time a user might otherwise have to spend in reconciling inconsistencies between datasets, e.g., warnings, automatic pre-processing and corrections are made internally in LUCI in its pre-processing toolbox.

Suitability for the needs of the OLW programme

LUCI is not yet a mature enough framework to be used as a “main” framework within the OLW programme, nor can some of the evaluation criterion be answered with certainty (e.g., to what extent is it open source) until its development path into the future is more clear.
It does explicitly and computationally efficiently link sub-field and farm scale to catchment (and national) scale and is modular. Proof of concept trials suggest it is easy to integrate code in other languages where stable interfaces to these languages have been developed for Python, for example R and Fortran. More generally though significant coding effort would be needed to move to a more general interoperable system.

Estuarine and coastal environments are not currently supported.

It only partially meets the criteria on temporal resolution. Water quantity calculations can be carried out and output at any temporal resolution, but N, P and sediment results are monthly at best.

Web Resources

http://www.LUCItools.org/

References

Sharps, K., Masante, D., Thomas, A., Jackson, B., Redhead, J., May, L., Prosser, H., Cosby, B., Emmett, B., Jones, L. (2017) Comparing strengths and weaknesses of three ecosystem services modelling tools in a diverse UK river catchment, in press at *Science of the Total Environment*, available online at Science Direct.

Trodahl, M.I., Jackson, B.M., Deslippe, J., Metherell, A.K. (2017) Investigating Trade-offs Between Water Quality and Agricultural Productivity Using the Land Utilisation and Capability Indicator (LUCI) – a New Zealand Application, in press at *Ecosystem Services*, available online at Science Direct.

Emmett, B.A., Cooper, D., Smart, S., Jackson, B., Thomas, A., Cosby, B., Evans, C. et al. (2016) Spatial patterns and environmental constraints on ecosystem services at a catchment scale, available online at *Science of the Total Environment.*

Clark, M.P., Schaefli, B., Schymanski, S.J., Samaniego, L., Luce, C.H., Jackson, B.M., Freer, J.E., Arnold, J.R., Moore, R.D., Istanbulluoglu, I., Ceola, S. (2016) Improving the theoretical underpinnings of process based hydrologic models. *Water Resources Research,* 52: 2350-2365, doi: 10.1002/2015WR017910.

McIntyre, N., Ballard, C., Bruen, M., Bulygina, N., Buytaert, W., Cluckie, I., Dunn, Wheater, H. including Jackson, B.M. (2014) Modelling the hydrological impacts of rural land use change. *Hydrology Research,* 45(6): 737–754.

Fraser, C.E., McIntyre, N., Jackson, B.M., Wheater, H.S. (2013) Upscaling hydrological processes and land management change impacts using a metamodelling procedure. *Water Resources Research,* 49(9): 5817-5833*.*

Robinson, D.A., Jackson, B.M., Clothier, B.E., Dominati, E.J., Marchant, S.C., Cooper, D.M., Bristow, K.L. (2013) Advances in Soil Ecosystem Services: Concepts, Models, and Applications for Earth System Life Support. *Vadose Zone Journal,* 12(2).

Ballard, C., Bulygina, N., Cluckie, I., Dangerfield, S., Ewen, J., Frogbrook, Z., Geris, J., Henshaw, A., Jackson, B.M., Wheater, H. (2013) N. McIntyre, C. Thorne (Eds.) *Land use management effects on flood flows and sediments - guidance on prediction* (C719). London, UK: CIRIA.

Jackson, B., Pagella, T., Sinclair, F., Orellana, B.M., Henshaw, A., Reynolds, B, Eycott, A. (2013) Polyscape: A GIS mapping framework providing efficient and spatially explicit landscape-scale valuation of multiple ecosystem services. *Landscape and Urban Planning,* 112: 74–88.

Jackson, B.M., Wheater, H.S., McIntyre, N.R., Chell, J., Francis, O.J., Frogbrook, Z., Solloway, I. (2008) The impact of upland land management on flooding: insights from a multiscale experimental and modelling programme. *Journal of Flood Risk Management,* 1(2): 71–80.

Jackson, B.M., Wheater, H.S., Wade, A.J., Butterfield, D., Mathias, S.A., Ireson, A.M., Whitehead, P.G. (2007). *Ecological Modelling,* 209(1): 41–52.

Jackson, B.M., Wheater, H.S., Mathias, S.A., McIntyre, N., Butler, A. (2006) A simple model of variable residence time flow and nutrient transport in the chalk. *Journal of Hydrology,* 330(1-2): 221–234.

Appendix M MyLand

Intended Purpose

Problem types

MyLand is a web-based strategic land use planning tool designed to assist land owners to improve the long-term profitability and sustainability of their land management. It is intended to simulate production, economic and environmental attributes of a range of land uses over the long term (i.e., one or more forestry rotations).

Key intended end users are land managers (e.g., farmers) and their advisors (e.g., farm consultants) and Regional Councils. It serves as an education tool in showing the potential integration of forestry production and cashflows within long term farm management. It does not predict or 'optimise' land use change – it simulates the impacts of user-specified land use scenarios on farm inputs and outputs.

Models

Spatial and temporal resolution:

- Planning horizon – typically long enough to cover one or more forestry rotations (e.g., 30+ years).
- Temporal resolution – annual time steps.
- Spatial scale – typically farm-scale, potentially up to catchment scale.
- Spatial resolution – sub-paddock modelling units defined by the user.

Land uses included are currently plantation forestry and sheep farming. Forestry production is based on productivity surfaces for five species that drive internal lookup tables derived from detailed *Forecaster* modelling (the growth and yield models incorporated within Forecaster are not themselves implemented within MyLand). Forestry cost models for harvesting, road construction and log transport are implemented, while other costs are represented by regime and species defaults. Grazing production is based on user inputs derived from models like FARMAX – MyLand provides the framework for keeping track of the area, inputs and outputs over time.

Environmental attributes are currently limited to N, P and greenhouse gases (CO_2, CH_4, N_2O). These are tracked based on user-defined values for grazing (e.g., from OVERSEER) and default values for forestry. If a starting date for joining the ETS is entered the system sells or buys carbon depending on the carbon balance.

Economic analyses are undertaken in four ways – with no environmental costs, with carbon, with nitrates, and with carbon plus nitrates. Outputs include annual cashflows, net present value, land expectation value, internal rate of return, whole property earnings before interest and taxation.

Other land uses could be modelled on the same basis (i.e., annual time steps with user-defined attributes that can be expressed as inputs or outputs on a per hectare basis through time). MyLand does not model interactions between modelling units or modify outputs according to their spatial arrangement.

Data

- High resolution aerial photography (Terralink).
- Public domain GIS layers, e.g., Altitude, Temperature, Slope, Aspect, Land Use Capability (LUC), Land Environments New Zealand (LENZ), Land Cover Database (LCDB2).
- User-defined polygons and other attribute data.

The user draws polygons onto underlying aerial photography imagery to define management units. The intersection of these management units with the underlying GIS layers sets parameters for the models or attribute default values. Other inputs can be entered manually (e.g., from FARMAX outputs).

Institutional Support

MyLand was initially funded by MPI as a joint project between Scion and Agresearch. Further development has been funded by Scion but is currently on hold as the project champion has left Scion and funding for further work has not been sought.

Technical aspects of integration

Model coupling

All models used by MyLand have been coded into the main software solution. Relevant outputs from OVERSEER and FARMAX can be manually entered into the user interface. To introduce new models or external connectivity would require modification of the code base.

Data import and export

SQL Server 2008R2 is used to store all non-spatial data as well as user-defined geometric shapes delineating blocks, roads and skids (log processing sites). All non-static data is entered directly though the web interface. Pre-defined spatial layers are stored in ArcGIS Server 10. Access to the data is only available through the supplied web interface for on-screen viewing or printing of pre-defined reports. There is limited exporting of some cash flow and production outputs.

Visualisation

All data is available for viewing through the web site including a number of geospatial layers.

User interface

MyLand is a web-based solution that uses the now deprecated Microsoft Silverlight to provide a rich internet application. It uses Latitude Geographics GeoCortex Viewer for Silverlight 1.9 to display web maps.

Metadata, scenario management, provenance, auditability

MyLand uses standard versioning when updates are delivered. Reports include the MyLand version used to generate them. While results are cached for improved performance, whenever a new version is uploaded the results are rerun when next retrieved to ensure they are up-to-date.

MyLand is based around comparison of scenarios. Each user has their own login and set of scenarios that they create and manage. The sharing of scenarios between users is not supported.

Underlying language and computing requirements

MyLand is written in C# using the Microsoft Silverlight 4 framework. It is hosted on an instance of Microsoft's web server (IIS 7.5). MyLand uses a third-party commercial package (Latitude Geographics GeoCortex Viewer for Silverlight 1.9) to display web maps.

Because of the use of Silverlight the website can only be displayed on web browsers that support NPAPI plug-ins. Most current browsers no longer support this plug-in model so MyLand is only known to work in Internet Explorer.

Uncertainty methods and calibration

Uncertainty and calibration is not explicitly addressed.

Management of IP and proprietary material

MyLand is not open source. Access to source code would need to be arranged with Scion. MyLand requires a login to be created for each user.

Silverlight will not be supported by Microsoft beyond 2020.

Quality assurance

There are regression tests that can be run before releasing a new version but because of its limited use no formal testing procedures have been put in place. Comparisons with Excel implementations of the models have generally been used to check the validity of results.

Experience with Use of the Framework

Does it meet the intended purpose?

MyLand has been well-received by many stakeholders but has not gained wide use. Its focus has changed in an attempt to garner additional support with a variety of stakeholders but it has not yet been used consistently by individuals. Its main use to date has been for internal research purposes and specific contracts, as well as extension/education.

Barriers to use

The main reason that MyLand has not gained wider acceptance is probably the lack of a strong incentive for landowners to plan changes to their current farm management. Without the push from agriculture entering the ETS or nutrient caps (until recently), farmers have been reluctant to invest their time and money in this type of modelling. Users must redefine the spatial areas, integrate data with their existing systems and tailor models to their particular property as much as possible. It also requires a change in thinking – strategic planning over a 30-year time horizon is very different to addressing shorter-term management issues confronting farmers. The representation of agriculture relies on OVERSEER and FARMAX outputs and is relatively simplistic – MyLand itself does not predict responses to farm system changes.

IP constraints have not been a key barrier to use.

Flexibility, extensibility and adaptability

Modifications require detailed knowledge of the MyLand codebase. Additions to the model that are compatible with the existing model behaviour can be done. Additional outputs can be stored in the database against existing entities (e.g., Block or Scenario) but updated reports would need to be manually constructed to make these visible. Introduction of models with a different temporal time-step or different spatial description would be cumbersome.

Ease of use in particular applications

Entry of data and run time is not difficult for smaller farms but manual GIS entry for larger estates can be time-consuming. Providing accurate values for inputs can be difficult (although defaults are provided). MyLand sits between the detailed Whole Farm Plan approach (e.g., Horizons) and "ready reckoner" calculators.

Suitability for the needs of the OLW programme

MyLand does not provide a suitable open-source, open data framework for the OLW programme to build on. However, the data and models used by MyLand are still relevant, as are the attributes reported and the problems addressed.

MyLand provides a useful example of a tool intended for use by land managers themselves rather than researchers, and the issues faced by developers trying to reach these end users. Regional Council advisors have commented that it – or something like it – is exactly what is required to communicate with landowners.

Reference

West, G.G., Turner, J.A. (2013) MyLand: a web-based and meta-model decision support system framework for spatial and temporal evaluation of integrated land use. *Scandinavian Journal of Forest Research*. http://www.tandfonline.com/doi/abs/10.1080/02827581.2013.866690

Appendix N NZFARM

Intended Purpose

Problem types

New Zealand Forest and Agriculture Regional Model (NZFARM) is an economic land use model developed by Landcare Research (Daigneault et al. 2012; 2014; 2016). Its primary use is to provide decision-makers with information on the economic impacts of environmental policy (including options to improve water quality), agribusiness or farm policy, and/or climate change. It can be used to assess how changes in climate, technology, commodity supply or demand, resource constraints, or farm, resource, or environmental policy could affect a host of economic or environmental performance indicators that are important to decisions-makers and rural landowners. NZFARM has been used to assess agro-environmental policies and resource management options for government ministries (Daigneault et al. 2012), regional councils (Daigneault et al. 2013), and collective landowners (Awatere et al. 2015).

Models

NZFARM is a comparative-static, non-linear, partial equilibrium mathematical programming model of New Zealand land use capable of operating at the national, regional, catchment and sub-catchment scale. The model tracks multiple parameters including changes in land use, land management, agricultural production, nutrient losses, sediment and GHG emissions/C sequestration. The model can assess a range of policy options including, but not limited to, catchment-level cap and trade programme, imposition of nutrient leaching constraints at the enterprise-level, allocation options, taxes/subsidies, and good management practice requirements. The model is parameterised such that responses to policy are assumed to be a medium- to long-term response where landowners make changes over a 5 to 10-year period[28]. There are three key components (see Figure 1) to the model (***Economic, environment, and land management***) and these are discussed in more detail below:

Economic component. The core component of the model is economic with the objective function maximising rural income while accounting for the environmental impacts of land use and land use changes. Production activities in each region of NZ-FARM are characterised by fixed and variable input costs, output price and other relevant forms of payments such as environmental payments. Production and land use are endogenously determined in a nested framework such that landowners simultaneously decide on the optimal mix of land use for their fixed area, given their land use classification (LUC; if appropriate) and soil type. This then allows landowners to allocate their land between various enterprises that will yield them the maximum net return for their land use.

[28] The static analysis compares two different equilibrium states (before and after a change in some underlying exogenous parameter), in which the outcome of these two states is annualized. The annual outcome of the aftershock state is assumed to be in the steady state situation (achieved after 5-10 years). Although the model does not study the motion towards equilibrium (the path of the above mentioned 5-10 years period), in some cases, five-year time steps for model runs, however, have been used for some analysis to simulate a dynamic transition pathway.

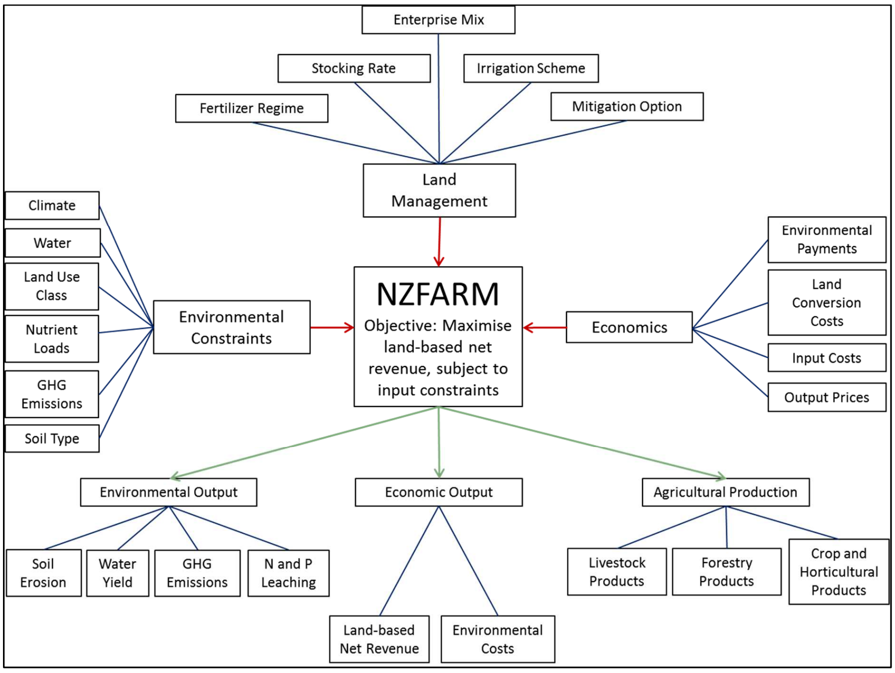

Figure 1: NZFARM components.

NZFARM can account for all types of rural production activities. To date the following activities or enterprises have been included in the modelling:

- Pastoral uses (sheep, beef, dairy and deer).
- Horticultural uses (e.g., kiwifruit, grapes).
- Arable uses (e.g., maize, various arable rotations).
- Forestry.

Other land uses can be included as-long-as profitability and environmental impacts are available.

Environmental component. In addition to estimating economic output from agricultural and forestry sectors, NZFARM also has the ability to track environmental outputs. Currently the model has been used to track:

- Nitrogen (N) and Phosphorus (P) leaching rates for pastoral farming were obtained from the most recent version of OVERSEER while N and P leaching rates for all other enterprises were constructed using SPASMO or other literature.

- Forest productivity and carbon sequestration were derived from the CenW model.
- Greenhouse gas (GHG) emissions for all other enterprises were derived using the Intergovernmental Panel on Climate Change's (IIPCC) Good Practice Guidance (2000) and match the categories in the latest New Zealand GHG Inventory (manure management, agricultural soils, etc.).
- Water yield is based on WATYIELD (Ausseil et al. 2013).
- Sediment losses are based on SedNet and NZEEM models.
- *E. Coli* has also been included for some analyses and based on CLUES/SPARROW modelling.

Land management component. Simulating endogenous land management is an integral part of the model, which can differentiate between 'business as usual' (BAU) farm practices and less-typical options that can change levels of agricultural output, nutrient leaching, sediment loss, and GHG emissions, among other things. Key land management options include changing fertiliser regimes and stocking rates, adding an irrigation system or implementing mitigation technologies such as the installation of a dairy feed pad, fencing streams, constructing wetlands or specified packages of management practices. Again, additional management practices can be included provided it is possible to estimate the environmental impacts and profitability.

Data

NZFARM has already been parameterised in detail for several New Zealand catchments (e.g., Manawatu, Selwyn, Hinds, Hurunui-Waiau, Whangarei Harbour, Ruamāhanga and Kaipara Harbour). In addition, a more aggregated version of the economic land use model has been parameterised for all of New Zealand using representative farm data. The full range of model variables and typical data sources for NZFARM are listed in Table 1, however, the range of variables and data can be modified based on the scope of the work. Note that if data on additional land uses/management practices and/or environmental outputs exist, they can easily be incorporated into the existing model framework. Technically, input data are based on polygons (i.e., farms) which are then converted to XLS format to be used in the model.

Institutional Support

The initial funding for NZFARM was through FRST research funding with subsequent MPI "Sustainable Land Management and Climate Change Programme (SLMACC)" funding used to expand and enhance model capability. Landcare Research has led the development of NZFARM with advice and support also being provided by the USDA Economic Research Service (USDA-ERS). The NZFARM modelling structure is based on the USDA-ERS REAP model. On-going support and development of NZFARM has been provided through internal funding from Landcare Research (SSIF funding), other MBIE research programmes and commercial funding sources. The commercial and MBIE funding sources have contributed to the model extensions required to analyse specific questions/issues with these extensions being embedded into the modelling framework. There is on-going maintenance of the model and where applicable some of the data sources that underpin the modelling. This support is expected to continue into the future given the commercial use of the model.

Table 1. Data Sources for NZFARM's modelling of a specific catchment.

Variable	Data requirement	Availability	Comments
Geographic area	GIS data identifying the catchment or other relevant area	Catchment and sub-catchments based on REC	Can use alternative boundaries if available/desired
Land use and enterprise mix	GIS data file(s) of current land use with the catchment Key enterprises (e.g., dairy).	A national land use map was estimated based on AgriBase and LCDB4 (2012/2013)	Land use map should be verified by project partners and/or stakeholders
Climate	Temperature and precipitation	Historical data is available. Future climate projections for all of New Zealand is now available	Required for assessimg impacts on primary productivity so need to link with pasture/livestock, crop, and forestry models
Soil type	Soil maps (Fundamental Soil Layer) used to divide area into dominant soil types	S-map (partial coverage only) and the NZ Land Resource Inventory (NZLRI) are available	Used for estimating impacts on nutrient losses
Stocking rates	Based on animal productivity model (eg FARMAX) estimates or carrying capacity map	Average land carrying capacity from NZLRI as well as more detailed 'stocking budgets' for various dairy and sheep and beef systems have been estimated	
Input costs	Stock purchases, electricity and fuel use, fertiliser, labour, supplementary feed, grazing fees, etc.	Obtained using a mix of: pers. comm. with farm consultants and regional experts, MPI farm monitoring report, Lincoln Financial Budget Manual	If appropriate, additional information can be sourced or verifed by project partners and/or stakeholders.
Product outputs	Milk solids, dairy calves, lambs, mutton, beef, venison, grains, fruits, vegetables, timber, etc.	Yields are available at the farm scale. Data comes from farm consultants and regional experts, MPI farm monitoring report, Lincoln Financial Budget Manual	If appropriate, additional information can be sourced or verifed by project partners and/or stakeholders.
Commodity Prices	Same as outputs, but in $/kg or $/m3	Obtained from MPI and other sources	
Environmental indicators	GHG Emissions Forest carbon sequestration Nitrogen and phosphorous loss Water yield Sediment loss *E.Coli*	GHG Emissions estimated using same methodology as MfE's Annual NZ Inventory calculations Forest sequestration based on CenW Leaching rates derived using the OVERSEER and/or SPASMO model	Can be updated with farm or catchment-specific data

Variable	Data requirement	Availability	Comments
		Water yield estimated using methods WATYIELD (see Ausseil et al 2013) Sediment loss based on SedNet or NZEEM models *E.Coli* based on CLUES/SPARROW model	

Technical aspects of integration

Model coupling

Spatial layers of model output are generated and integrated at the farm scale using up- or down-scale techniques. There is no feedback between model components (see Figure 1). The outputs of all the component models are integrated within NZFARM. Adding further components involves modification of the spatial input dataset and then inclusion into the core model code.

Data import and export

NZFARM uses ESRI file geodatabases for the spatial input data and associated tables, which in turn is converted to XLS files for use within the model. All data is stored locally on the user's machine.

Visualisation

Model outputs (micro-economic and environmental data) are mainly presented quantitatively in XLS files (as regional aggregated totals and average per hectares). However, an algorithm (i.e., GUI coded in PERL) has been developed in the national model that automatically generates 2-D maps.

User interface

NZFARM operates at a range of scales, with the key scales being national, regional, catchment and sub-catchment. The model uses the interface in the proprietary software, General Algebraic Modelling System (GAMS), to run NZFARM. The results can be output into an XLS format.

There is one application where a customised user interface has been developed. This interface was developed to undertake climate and nutrient analysis at the national-scale (0.5-minute grid for New Zealand). The interface works alongside the GAMS software. From the interface the user can define the level of GHG, nitrogen and phosphorous tax rates, and milk, beef, lamb, wool and timber prices. The user can also specify what output they would like – revenue, area, environmental parameters, animals and/or production. The interface is able to produce estimates in the form of graphs, figures, GIS-based maps, and compiled reports. This GUI was written in PERL.

Metadata, scenario management, provenance, auditability

NZFARM uses comparative analysis. This approach is implemented using the GAMS loop procedure to create cross scenario comparison tables. The basic structure of a comparative analysis is outlined in Figure 2. The first three boxes (i.e., baseline calibration) reflect preparatory steps that would be done in a conventional GAMS program where one sets up the initial data and the model then solves. This first stage will generate the baseline results. In the second stage (i.e., scenario analysis), the comparative model analysis begins, in which, the scenarios are identified and the scenario data defined. After that, the base scenario values are preserved so that the parameter values could be changed during the scenario run. Finally, in the third stage (i.e., results), the output of the scenario is compared to the baseline.

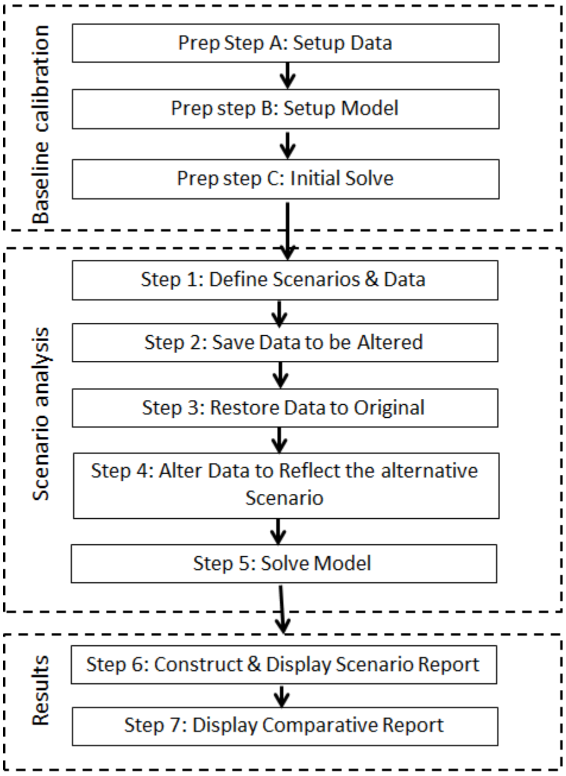

Figure 2. The basic structure of a comparative analysis within NZFARM.

Regarding auditing, NZFARM have been peer reviewed in the literature for several case studies. In addition, the model outputs have been validated with stakeholders and it was ensured that the model is responding logically to historical changes in output prices and land-use policy (Daigneault et al. 2012; 2013; 2016).

Underlying language and computing requirements

The model is written and maintained in GAMS – a high level modeling system for mathematical optimization. GAMS contains an integrated development environment (IDE) and is connected to a group of third-party optimization solvers. NZFARM optimization is based on the CONOPT Solver. The model can operate on MS Windows, Mac OS X, Linux, and DOS systems.

Uncertainty methods and calibration

Calibration
NZFARM's use of positive mathematical programming (PMP) and constant elasticity of transformation (CET) functions allows the modelled land-use area to closely match the initial GIS-derived land-use areas. In addition, this calibration framework addresses problems of overspecialisation and corner solutions. We find that this method results in only minor differences between observed and modelled baseline land use at the enterprise level (e.g., three per cent for the Manawatu catchment (Daigneault et al. 2012); two per cent for the Hurunui–Waiau catchments (Daigneault et al. 2012); and less than one per cent for the Hinds catchment (Daigneault et al. 2013)).

There has been some criticism of the calibration process of NZFARM (Doole and Marsh 2013). Many of the Doole and Marsh concerns related to their perceived degree of manipulation required for NZFARM's baseline to reflect an observed baseline. However, as mentioned above, NZFARM's use of PMP and CET functions allows the modelled land-use area to closely match the initial GIS-derived land-use areas. In all analyses to date there have only been minor differences between observed and modelled baseline land use at the enterprise level (all less than 3 percent). Most other key model outputs are based on fixed coefficients from the enterprise area and again there are small relative differences between the observed and calibrated values. Therefore, the initial land-use areas and farm-level financial budgets used as key inputs for NZFARM calibration do provide an accurate representation of catchment economic conditions and generate a model baseline similar to observed land uses. This level of precision demonstrates that baseline calibration is minimal when sufficient effort is taken to obtain robust input data.

Uncertainty
Sensitivity analysis is typically used to determine the impact of those more uncertain parameters or data sets. The GAMS loop procedure is used to streamline the sensitivity analysis. A previous study has approached uncertainty with respect to input data and parameters. That was achieved by using alternative datasets and parameter assumptions, and through a scenario analysis. The analysis helped in understanding the role of uncertainty on the economic land use model, NZFARM (Fernandez et al., 2014). The study found that NZFARM responds consistently, at every scenario, to the policy shock regardless of the input data introduced. In addition, the study found that uncertainty around net revenue and the adoption of mitigation practices demand greater attention because of the implications on land allocation and production. Uncertainty in environmental outputs (e.g., N leaching figures) introduces variability on land conversion that from a wider perspective appear to be negligible if compared to total land areas.

NZFARM is currently being complemented with probabilistic Monte-Carlo simulation techniques to assess the implications of bio-economic uncertainty. Uncertainty is being modelled through probability distribution functions of productivity and prices of various crops and tree species as a post-solution procedure taking NZFARM's optimal land allocation as an input. The project's aim is "to enhance the future prosperity of Māori by incorporating potential climate change impacts into land investment decisions and providing holistic approaches for managing climate-sensitive catchments" and is funded by the Deep South National Science Challenge. The uncertainty analysis is being undertaken by Scion.

Management of IP and proprietary material

GAMS and ArcGIS are proprietary software. NZFARM is written in GAMS with the IP for the model held by Landcare Research. To use NZFARM a GAMS licence is required and access to the NZFARM code, which would require an arrangement with Landcare Research for the code's usage.

Quality assurance

There are three areas where quality assurance is considered:

1. Solvers: The GAMS corporation has a process for managing the quality assurance of their modelling platform, including the solvers that are used to solve models.

2. Modelling code and theoretical underpinning: For NZFARM a calibration process is used during the baseline development phase. The resulting NZFARM baseline is then compared to the current state to determine how well the modelled baseline matches the actual baseline. The methodological underpinning is based on mathematical programming approaches which are well-debated in the literature. Similarly, analyses based on NZFARM and also the REAP model on which NZFARM is based have been peer reviewed in the literature.

3. Input data: The data used within NZFARM comes from many sources, mostly published data sources which have their own review processes. Potential errors in input data or related coding are typically identified during the calibration process as noted in 2).

To date there has been core master model from which a number of different analyses have been done. There has not been a need to generate a new version of that core model but some modifications have been made to streamline the code. This may change in the future if methodological changes to the core model are required or are made. Therefore, the key to quality assurance of NZFARM lies in the calibration process used to establish the baseline and then via feedback from stakeholders for the results.

Experience with Use of the System

Does it meet the intended purpose?

NZFARM has shown usefulness in illustrating the trade-off between economic and environment impacts of environmentally-focused policies. While the model to date has only been used to assess the impact of environmental policy, it can also be used to assess other types of policies, e.g., agricultural policies. In addition, a more aggregated version of the economic land use model has been parameterised for all of New Zealand using representative farm data.

Barriers to use

NZFARM is closed-source with code managed by Landcare Research, so that other parties cannot modify or extend the models or interface themselves. NZFARM input layers, such as land-use, water yield, and soil erosion, are being updated by Landcare Research team. The range of the input data can be modified based on the scope of the work. NZFARM relies on GAMS, which might present a barrier for smaller consultants or community groups. Agreement with Landcare Research is required for use of NZFARM in consulting projects, although this has not been a constraint.

Flexibility, extensibility and adaptability

It is relatively easy to incorporate new components into the existing model framework, if data on additional land uses/land management and/or environmental outputs exist. This new data is integrated as a spatial layer to NZFARM dataset. For instance, *E. coli* data was integrated with NZFARM to estimate the cost-effective of sediment and *E. coli* mitigation practices in Whangarei catchment. Recently, NZFARM was successfully linked to an agent-based decision-making framework (ARLUNZ) to estimate the impacts of climate change policy on land use. ARLUNZ is written in Version 5.0.5 of NetLogo (Wilensky, 1999) using the GIS, String and Shell extensions. Python 2.7 is used to facilitate a loose coupling (Brandmeyer & Karimi, 2000) between ARLUNZ and a modified version of NZFARM that provides economic information within ARLUNZ. The modified version used within ARLUNZ has been refined to produce an economically optimised result for each farm rather than an optimised landscape for an entire catchment.

Ease of use in particular applications

Through numerous different analyses in different catchments it has been shown that it is relatively easy to set up and run the core model. However, if additional components are requested to be incorporated to the model, resources, in terms of time and skills, might be needed to integrate and validate the new component. The model processing time is quite fast, although it might take a bit more time if it runs at the national level (around 20-30 minutes).

GAMS, through some of its solvers, provides limited parallel-computing capabilities that could fully utilise the new multi-core architecture offered by most computers these days. Hence, depending on the amount of variables, parameters, constraints, and type of problem (defining the solver to be used) the model could be subject to long run times. Such shortcoming constraints the spatial and temporal detail that can be included in the model. However, this is currently not a problem for NZFARM as it was coded efficiently leading to a short execution time.

Suitability for the needs of the OLW programme

The main objective of the programme is to develop “a nationally recognised modelling platform, populated with models and drawing on national datasets, to be used for assessment of environmental, production and economic implications of land use and land use change”. Hence, NZFARM has the potential to be highly suitable to the programme’s objectives. The main advantages offered are its flexibility to incorporate data from various models and to be used in various contexts. From an economic point of view, NZFARM will be able to find the optimal (i.e., most profitable or least costly) combinations of land uses and mitigation strategies to reduce environmental impacts, at the block and catchment levels, which is among the Challenge’s main objectives.

NZFARM has some limitations in its applicability to the objectives of the programme.

- To use NZFARM a GAMS licence is required and access to the NZFARM code, which would require an arrangement with Landcare Research for the code’s usage.
- Only steady-state predictions are provided, whereas the programme calls for dynamic simulations. Five-year time steps for model runs, however, have been used for some analysis to simulate a dynamic transition pathway.

- Contrary to other similar models (e.g., LAM), the fact that the NZFARM's structure is not freely available entails all of the usual disadvantages offered by closed-source models: they cannot be modified, expanded or improved by any modeller with the right sets of skills. However, access can be granted under mutual agreement. Among the advantages of such arrangement is that any changes would be controlled and known by Landcare.
- The fact that the NZFARM includes the time dimension simplistically is also a limitation that could be improved by including a time-dependent path for some of its variables to make the model of a dynamic nature. Such extension would be useful for land uses for which the time dimension is of critical importance, e. g. forest productivity affecting cash flow, erosion and leaching phenomena.
- It is common practice in mathematical programming models (i.e., the broad category to which NZFARM belongs) to lump space as much as possible to make the model tractable and to achieve faster solution times. This could become a barrier when the spatial correlations are critical to the problem in hand, e. g. clustering land uses to achieve higher economies of scale or modelling land uses that reduce nutrient leaching (a riparian forest between a dairy paddock and a stream).
- Uncertainty is considered only through a scenario approach rather than a probabilistic one. However, Monte Carlo simulation techniques have been used with NZFARM to include bio-economic uncertainty.

References

Ausseil, A.G., Dymond, J.R., Kirschbaum, M.U.F., Andrew, R.M., Parfitt, R.L. (2013) Assessment of multiple ecosystem services in New Zealand at the catchment scale. *Environ. Modell. Softw.*, 43: 37–48.

Awatere, S., Daigneault, A., Hainsworth, S., Fenemor, A., Tahi, M. (2015) Land-Use Options for Mākirikiri Aggregated Trust lands under a kaitiakitanga framework. *Landcare Research Contract Report,* LC2135 for Ministry for Primary Industries.

Brandmeyer, J.E., Karimi, H.A. (2000) Coupling methodologies for environmental models. *Environmental Modelling & Software*: 15(5): 479–88. doi: http://dx.doi.org/10.1016/S1364-8152(00)00027-X.

Daigneault, A., Greenhalgh, S., Samarasinghe, O. (2014) A response to methodological limitations in the evaluation of policies to reduce nitrate leaching from New Zealand agriculture. *Aust. J. Agr. Resource. Ec.*: 58(2): 281–290.

Daigneault, A., McDonald, H., Elliott, S., Howard-Williams, C., Greenhalgh, S., Guysev, M., Kerr, S., Lennox, J., Lilburne, L., Morgenstern, U., Norton, N., Quinn, J., Rutherford, K., Snelder, T., Wilcock, B. (2012) Evaluation of the impact of different policy options for managing to water quality limits, Final Report – Main Report and Appendices. *MPI Technical Paper,* No: 2012/46.

Daigneault, A., Samarasinghe, O., Lilburne, L. (2013) Modelling Economic Impacts of Nutrient Allocation Policies in Canterbury: Hinds Catchment: Final Report. *Landcare Research Contract Report,* LC1490 for Ministry for the Environment.

Daigneault, A., Greenhalgh, S., Samarasinghe, O. (2016) Economic Impacts of Multiple Agro-Environmental Policies on New Zealand Land Use*. Environmental and Resource Economics*: 1–23.

Doole, G., Marsh, D. (2013) Methodological limitations in the evaluation of policies to reduce nitrate leaching from New Zealand agriculture*. Australian Journal of Agricultural and Resource Economics*, 58: 78–89.

Fernandez, M., Samarasinghe, O., Daigneault, A. (2014) Uncertainty Analysis of NZFARM: Selwyn Case Study Approach. *Uncertainty Pipeline Capability Fund Report.*

Wilensky, U., NetLogo (1999) http://ccl.northwestern.edu/netlogo. *Centre for Connected Learning and Computer-Based Modeling. North-western* University, Evanston, IL.

Appendix O OVERSEER – Linkages with other models

Intended Purpose

OVERSEER® Nutrient Budget model (OVERSEER) is a decision support system (DST) farm model that has been developed in New Zealand as an industry standard for recommending nutrient inputs and estimating nutrient losses to water (Wheeler *et al*., 2014; www.OVERSEER.org.nz). The DST computes the long-term (multiyear) nutrient budget of many farming enterprises including pastoral, horticultural, arable and vegetable farming (Rutherford et al. 2008; Cichota *et al*., 2010). It models the complex and dynamic bio-physical processes occurring on a farm to estimate nutrient losses through the air, soil and across the land. OVERSEER assumes near equilibrium farm systems and so losses that occur as the system changes (e.g., transition from dryland to irrigated farming or from cropping to pasture) are not captured. Both an online version, which allows data storage, and a standalone version are available. OVERSEER has an API that let you make calls to the internal functions of another program.

OVERSEER is jointly owned by the Ministry for Primary Industries (MPI), AgResearch Limited and the Fertiliser Association of New Zealand (FANZ). OVERSEER Limited was established in 2016 and is licenced to deliver OVERSEER. The DST is a product of many years of scientific research, and is updated regularly (Shepherd and Wheeler, 2012), with the latest version (6.2.3) released on 7 November 2016.

OVERSEER is increasingly being used by regulatory authorities (e.g., Regional Councils) as a tool to enforce limits on nutrients losses and maintain or improve catchment water quality (e.g., Bay of Plenty Regional Council; Environment Canterbury; Otago Regional Council, 2014; Waikato Regional Council).

Currently, OVERSEER is free of charge, but any connection of OVERSEER into another software application requires a licence from OVERSEER Limited and attracts a licence fee.

Spatially explicit soils data is directly available in the OVERSEER model, by selecting the link to S-map (digital soil spatial information system for NZ; www.S-map.landcareresearch.co.nz).

Various institutions have built tools for running multiple OVERSEER files based on a base scenario and variation in certain parameters (pers com. Mike Rollo, AgResearch; Hemda Levy, Dairy NZ), with compilation of the output data into a spreadsheet. One of these tools also allowed detailed sensitivity analyses, based on the OVERSEER text files used by version 5. With the current change to xml format used by web versions, this tool is not working anymore, but could be modified.

OVERSEER linkages with other models are based on loose-coupling. Input and output files between different models, that are run independently, are exchanged either via manual data transfer or lookup tables. OVERSEER has been used in conjunction with various models to:

- To estimate nutrient concentrations in receiving waterways (surface water), OVERSEER has been linked with **CLUES** (see Appendix J). Examples include modelling studies from Elliott *et al*. (2008), Harris *et al*. (2009) and Parshotam *et al*. (2013).

- To spatially allocate N & P loss on a farm, outputs from OVERSEER have been linked with GIS data within the **MitAgator** (Critical Source Area Modelling tool). Examples include modelling studies by McDowell *et al.* (2015) and Risk *et al.* (2015).
- To assess the economic and environmental impact of farm management and farm management changes (including land use change), OVERSEER has been "linked" with **FARMAX** via manual transfer of data/information. Examples include studies by Wedderburn *et al*. (2013), Vogeler *et al*. (2014) and Watkins *et al.* (2015).

Further examples on the linkage of OVERSEER with other models include the LUCI framework, MyLand and NZFARM, all of which are described in separate reviews within this report.

OVERSEER was set up as a web service and coupled into OMS3 in an exploratory exercise in the Interoperable Freshwater Models project (Elliott et al., 2014), altering an example input (rainfall) through modification of the XML input file and reading the results.

CLUES has been applied in several locations in NZ to assess the impact of land use change on water quality and socioeconomic factors at a regional or national scale. For assessing nutrient losses from the pastoral sector CLUES uses a version of OVERSEER with restricted inputs and settings. To calculate the leaching from each spatial unit the model needs to be run repeatedly. Model outputs are stored in a simple csv-based record structure. Further detail of the model is provided in a separate summary as part of the current "Interoperable Models Review".

FARMAX is a whole-farm decision support model that uses monthly or 10-daily estimates of pasture growth, farm and herd information to determine the production and economic outcomes of managerial decisions (Bryant *et al*., 2010; White *et al*., 2010). Farmax operates a commercial business, licensing FARMAX applications to farmers, consultants, research institutions and industry bodies. There are two different versions with the FARMAX® Pro model designed for sheep and beef systems and FARMAX® Dairy Pro for dairy systems (www.FARMAX.co.nz). The model is a Windows application. Currently there is no automatic linkage between FARMAX and OVERSEER (Version 6), thus any linkage requires manual transfer of data. This manual transfer is not only time consuming but also prone to errors. It also limits the number of farm system scenarios that can be analysed.

While an automated transfer of data from FARMAX to OVERSEER is technically possible, some issues regarding the scientific alignment of the models need attention (as for example the metabolic models used). Also, FARMAX would need to be modified to include the farm block concept as used in OVERSEER, and OVERSEER requires a larger amount of farm input data. Technically, linkages between the two models would require a standardisation of a common file format that allows import of data. While OVERSEER uses an xml file format for its data, FARMAX currently uses a proprietary binary file format. The possibility of directly importing files from OVERSEER into FARMAX and vice versa via a web service call would also be valuable.

MitAgator™ is a farm scale geographic information system (GIS) based decision support tool (DST) which has been developed by Ballance Agri-Nutrients Limited (Risk et al. 2015). The model combines OVERSEER® inputs / outputs with GIS data layers to allocate N & P loss spatially across a farm. It estimates the risk of nitrogen, phosphorus, sediments and faecal indicator bacteria (E. coli losses) and identifies critical areas within a farm landscape to help landowners make more informed decisions about where losses occur the most appropriate mitigation options. Inputs required for MitAgator tool include both OVERSEER files and spatial information/map packages (e.g., Digital elevation model (DEM), geo-referenced farm map, soil map, and aerial photo).

Output of the model is a risk map of annual losses, broken into 20% quantiles for each contaminant (McDowell et al. 2015). The map identifies areas of higher nutrient loss for targeting the most suitable and cost effective mitigations. Mitigation strategies are specified by the user as either a single mitigation or several mitigations from a list, or a set target based on a percentage decrease desired (% less loss) or cost. The optimal solutions are found by an automated linear optimization routine, using the open source lpsolve (Berkelaar et al, 2004). Within the program, compatible combinations of mitigation strategies are added to a linear programming formulation involving binary variables and special ordered sets of type One (SOS1; http://lpsolve.sourceforge.net/5.5/LPBasics.htm). The combination of strategies that constitutes the optimal solution is then found by lpsolve using a branch and bound solution strategy, with carefully chosen branch and bound parameters to ensure sufficient solution speed. A new output is then provided with a new map of estimated losses, including histograms for load decreases and estimates of the upper and lower range of costs and efficiencies. The resolution of output data from MitAgator™ is a reflection of the resolution of input data (soil and elevation data). Currently, elevation data is available nationally at a 15-m resolution, but finer spatial data input is possible.

Suitability for the needs of the OLW programme

Given that OVERSEER is increasingly being used as a regulatory tool within New Zealand, the model should be part of the Interoperability Framework that will be developed. OVERSEER is already a component of many frameworks used within New Zealand (e.g., CLUES, LUCI, MyLand and NZFARM), with data transfer either manually or via lookup tables.

Coupling with other models has also been done using bespoke coupling, where models are coupled via codes or approaches developed for specific applications. The advantage of bespoke coupling is the highly flexibility, but it is very time-consuming, as coupling solutions must be individually developed and tested for the application. More details about bespoke coupling, and examples of its use within NZ is provided in Appendix C.

Full coupling of OVERSEER within a system by which individual models exchange input and output files and define their parameters through a joint calibration/inversion procedure is probably not feasible, partly due to licence issues. Other constraints such as temporal scale of the models, with OVERSEER mainly providing annual outputs, might also hinder full coupling.

References

Berkelaar, M., Eikland, K., Notebaert, P. (2004) lpsolve: Open source (Mix-Integer) Linear Programming System. Available at: http://lpsolve.sourceforge.net/5.5/ Accessed 18 December, 2014.

Bay of Plenty Regional Council (2016) "Plan change 10: Lake Rotorua Nutrient Management." Retrieved from: http://www.boprc.govt.nz/knowledge-centre/plans/regional-water-and-land-plan/lake-rotorua-nutrient-management-proposed-plan-change-10.

Bryant, J.R., Ogle, G., Marshall, P.R., Glassey, C.B., Lancaster, J.A.S., Garcia, S.C., Holmes, C.W. (2010) Description and evaluation of the FARMAX Dairy Pro decision support model. New Zealand. *Journal of Agricultural Research*, 53: 13–28.

Cichota, R., Brown, H., Snow, V.O., Wheeler, D.M., Hedderley, D., Zyskowski, R, Thomas, S. (2010) A nitrogen balance model for environmental accountability in cropping systems. *New Zealand Journal of Crop and Horticultural Science,* 38: 189–207.

Environment Canterbury. Farm Portal. Available at: https://farmportal.ecan.govt.nz

Elliott, S., McBride, G., Shankar, U., Semadeni-Davies, A., Quinn, J., Wheeler, D., Wedderburn, L., Small, B., Hewitt, A., Gibb, R., Parfitt, R., Clothier, B., Green, S., Harris, S., Rys, G. (2008) CLUES Spatial DSS: From Farm-Scale Leaching Models to Regional Decision Support. *International Congress on Environmental Modelling and Software Society* (iEMSs).

Elliott, S., Turek, G., Snow, V., Rutledge, D., Ritchie, A., Herzig, A., Snow, V. (2014) Framework for Interoperable Freshwater Models: Testing and Recommendations. Prepared for Ministry of Business, Innovation and Employment. NIW*A Client Report* No. HAM2014-013, National Institute of Water and Atmospheric Research.

Harris, S., Elliott, S., McBride, G., Shankar, U., Quinn, J., Wheeler, D., Wedderburn, L., Hewitt, A., Gibb, R., Parfitt, R, Clothier, B. (2009) Integrated assessment of the environmental, economic and social impacts of land use change using a GIS format–the CLUES model. In: *New Zealand Agricultural and Resource Economics Society Conference* August (pp. 27–28).

Wedderburn, M.E., de Oca, O.M., Dieguez, F. (2013) Developing frameworks to assess impacts of multiple drivers of change on grassland system. In: *Proceedings 22nd International Grassland Congress*, Australia.

Alexander, R.B., Smith, R.A., Schwarz, G.E. (2004) Estimates of diffuse pollution sources in surface waters of the United States using a spatially referenced watershed model. *Water Science and Technology*, 49(3): 1–10.

Elliott, A.H., Semadeni-Davies, A.F., Shankar, U., Zeldis, J.R., Wheeler, D.M., Plew, D.R., Rys, G.J., Harris, S.R. (2016) A national-scale GIS-based system for modelling impacts of land use on water quality. *Environmental Modelling & Software*, 86: 131–144. http://dx.doi.org/10.1016/j.envsoft.2016.09.011

Gillibrand, P.A., Inall, M.E., Portilla, E., Tett, P. (2013) A box model of the seasonal exchange and mixing in Regions of Restricted Exchange: Application to two contrasting Scottish inlets. *Environmental Modelling & Software*, 43: 144–159. 10.1016/j.envsoft.2013.02.008

Harris, S., Elliott, S., McBride, G., Shankar, U., Semandi-Davies, A., Quinn, J., Wheeler, D., Wedderburn, L., Hewitt, A., Gibb, R., Parfitt, R., Clothier, B., Green, S., Cacho, O., Dake, C., Rys, G. (2009) Integrated assessment of the environmental, economic and social impacts of land use change using a GIS format - the CLUES model. *New Zealand Agricultural and Resource Economics Conference*, August 2009.

Hughes, A., Semadeni-Davies, A., Tanner, C. (2013) Nutrient and sediment attenuation potential of wetlands in Southland and South Otago dairying areas. *National Institute of Water and Atmosperic Research.*

Luketina, D. (1998) Simple tidal prism models revisited. *Estuarine Coastal and Shelf Science*, 46(1): 77-84. DOI 10.1006/ecss.1997.0235

McDowell, R.W., Lucci, G.M., Peyroux, G., Yoswara, H., Cox, N., Brown, M., Wheeler, D., Watkins, N., Smith, C., Monaghan, R., Muirhead, R., Stafford, A., Risk, J., Old, A. MitAgator™: A tool to estimate and mitigate the loss of contaminants from land to water. *Moving farm systems to improved attenuation.* Fertilizer and Lime Research Centre, Massey University, Palmerston North, New Zealand. http://flrc.massey.ac.nz/publications.html

Parliamentary Commissioner for the Environment (2014) *Water quality in New Zealand: Land use and nutrient pollution*, Wellington, New Zealand. http://www.pce.parliament.nz/publications/all-publications/water-quality-in-new-zealand-land-use-and-nutrient-pollution

Risk, J., Old, A.B., Peyroux, G.R., Brown, M., Yoswara, H., Wheeler, D.M., Lucci, G.M., McDowell, R.W. (2015). MitAgator™ - in action solutions for managing nitrogen, phosphorus, sediment and e. Coli loss. *Moving farm systems to improved attenuation.* Fertilizer and Lime Research Centre, Massey University, Palmerston North, New Zealand. .http://flrc.massey.ac.nz/publications.html

Roberts, A., Watkins, N. (2014) One nutrient budget to rule them all–the OVERSEER® best practice data input standards. *Nutrient Management for the Farm, Catchment and Community, Occasional Report,* No. 27. Fertilizer and Lime Research Centre, Massey University Palmerston North, New Zealand.

Rosen, M.R., Reeves, R.R., Green, S., Clothier, B., Ironside, N. (2004) Prediction of groundwater nitrate contamination after closure of an unlined sheep feedlot. *Vadose Zone Journal*, 3(3): 990–1006.

Semadeni-Davies, A., Elliott, S. (2011) Application of CLUES to the Mataura Catchment. Impacts of land use and farm mitigation practices on nutrients. Prepared for Environment Southland. *NIWA Client Report*: HAM2011-018, National Insitute of Water and Atmospheric Research.

Semadeni-Davies, A., Elliott, S., McBride, G., Shankar, U. (2009) Using CLUES to identify impact catchments - Waikato River Catchment Pilot Study. *NIWA Client Report* HAM2009-101, National Institute of Water and Atmospheric Research.

Woods, R., Elliott, S., Shankar, U., Bidwell, V., Harris, S., Wheeler, D., Clothier, B., Green, S., Hewitt, A., Gibb, R., Parfitt, R. (2006) The CLUES Project: Predicting the Effects of Land-use on Water Quality – Stage II. *NIWA Client Report* HAM2006-096, National Institute of Water and Atmospheric Research.

Appendix P WISE (Waikato Integrated Scenario Explorer) – an application of the RIKS Geonamica framework

Intended Purpose

Problem types

WISE originated as part of a 4 year "Creating Futures" project (2006-2010) funded by the New Zealand Foundation for Research, Science and Technology. The project brought together an interdisciplinary team consisting of a regional council and social, environmental and economic researchers. The project aimed to develop new methods and tools to support integrated, long-term planning by 1) developing processes to evaluate, deliberate, and choose regional futures through scenario analysis and multi-criteria deliberation frameworks, and 2) developing an integrated spatial decision support system (ISDSS), dubbed the Waikato Integrated Scenario Explorer or WISE, to support the evaluation and deliberation processes. Together those tools were designed to help councils identify links and explore trade-offs between economic, environmental and social/cultural outcomes and the cumulative effects of many decisions over space and time. WISE was designed to allows users to explore "what-if" questions related to complex issues of sustainable development that require integrated planning and decision-making, e.g., for regional development and community outcome processes.

WISE is an integrated framework "designed to help examine weakly structured or unstructured problems characterised by many actors, many possibilities, and high uncertainty". WISE was originally developed to support integrated, long-term policy development and spatial planning processes, enabling Waikato Regional Council to holistically explore regional futures by integrating climate change, demographics, economics, hydrology, land use changes, terrestrial biodiversity and water quality. This objective continues, and specifically as of 2016 is framed as being development to "support evaluation and deliberation of different policy and planning strategies and scenarios in the context of long-term integrated planning as required by the Local Government Act and the Resource Management Act" (Rutledge et al., 2016), with potential to use it to explore options under non-statutory planning processes also noted.

Models

The current version of WISE (version 1.4) contains models operating across four spatial scales with various degrees of coupling within and between these four scales: 1) NZ with global climate and economic consideration, 2) the Waikato region, 3) cities and districts (e.g., Hamilton, Waipa), and 4) local scale operations (100 m x 100m grid cells). It outputs results on an annual time step, from 2013 to 2064. Many of the internal calculations are carried out at more detailed temporal resolution, or account in some other way for sub-annual variation (e.g., the hydrology model).

At NZ/global scale, a range of climate scenarios and economic drivers are explored, with the assumption that these influence or "drive" events within the Waikato region but that the region has insignificant influence back to the national and global climate and economy.

At regional scale, an economic futures model considers about 50 industries, household demand, and external economic drivers, which in turn generate demand for land informing land use change predictions. In addition to modelling economic activity, employment, energy use and energy-related

CO_2 emissions, and solid waste generation are considered by the futures model. A hydrological simulation model produces annual surface runoff and summer flow yields considering land use as well as climate inputs, and water quality is estimated via an adaptation of SPARROW to NZ (Elliott et al. 2005). Via this adapted model, WISE estimates annual nitrogen and phosphorus loads for each individual reach defined within the regional network.

At district scale, there are two models: a zoning tool and demographics tool. The zoning tool is effectively a constraint on the other models that include simulations of future land change outcome possibilities, allowing users to input various zones, plan, rules, designations, overlays etc. as desired. The user can specify a probability of change being permitted ranging from 0 (land use always prohibited) to 1 (land use always permitted). Zoning rules can change in time. The demographics model simulates fertility, mortality, net migration among districts, and net migration to/from each district to outside the region. Fertility, mortality and migration can be adjusted for different age-sex cohorts and migration can be adjusted for individual districts.

At local scale (100 x 100 m grid cells) two further models explore land use change and terrestrial biodiversity. Land use change is modelled over time based on demand for housing as generated by the district scale demographic model described above, and economic drivers as generated by the regional economic futures model. This is accomplished via a dynamic, spatially explicit cellular automata model considering 25 classes of land use. For each grid cell at each time step, the model calculates the potential for a cell to transition to every land use based on four factors: accessibility, local influence, suitability and zoning restrictions. The biodiversity model considers threats to ecosystems by considering current vs future projected proportions of native land cover. Current land cover data is compared to the future conditions expected given input from the land use change model output, further constrained by protection information from the NZ Protected Areas Network. The amount of native land cover expected for any scenario is estimated via LENZ data on average percentage of native cover associated with each land cover classification.

WISE is implemented in the Geonamica® framework (see discussion below). Some of the WISE models are specifically tailored for NZ and/or the Waikato specifically, while others use or built upon existing Geonamica model components. More generally, Geonamica includes as standard three model components simulating human processes: (1) water management, (2) land-use, and (3) crop choice and profit. The environment sub-system consists of a climate and weather model, a hydrology and soil model and a vegetation model. Simulations are performed on grid level with different temporal resolution.

Data

WISE contains a variety of component models, so there are a wide range of time/space scales and data conversion calculations taking place within the framework. GIS vector and raster information are used as are excel spreadsheets with tabular data; the internal coupled calculations are in main discrete time and space exchanges of information. A range of externally developed NZ models are embedded with the framework and these generally are hard-wired to call a range of data formats as per their individual requirements; specifics are addressed in the model detail in the WISE v1.4 technical manual.

Institutional Support

The initial development of WISE was carried out under the 'Creating Futures' project funded by the then Foundation for Research Science and Technology (FRST) under contract ENVW0601 to Waikato Regional Council. Waikato Regional Council provided additional funding and administrative support for the project. Landcare Research funded early development of this integrated spatial decision support system as part of the Sustainable Futures Waikato Capability Fund project. Since this contract concluded, Waikato Regional Council have taken custody of WISE. It has continued to promote the use and development of WISE in Waikato, and continues a program of development and update to keep the WISE model up-to-date and targeted to user needs. As part of its long-term Council Community Plan 2015–2025, Waikato Regional Council has committed funds for the maintenance, enhancement and upgrading of WISE over the next 10 years.

As noted above, WISE is built using Geonamica, a closed source, object-oriented, commercial application framework for constructing integrated models of the land system owned by the Research Institute for Knowledge Systems (RIKS), which operates out of the Netherlands. Geonamica has been designed for developing decision support frameworks that feature integrated dynamic models as core elements. It contains a variety of ready-to-use software building blocks required for the development of models, analysis tools and user interfaces. Geonamica is a key part of the RIKS business model, underpinning most of their consultancy services as well as their main other software product, the Metronamica land-use model. The current website for RIKS has only been minimally updated since 2013/2014 but the company is still active; a RIKS consultant and researcher is visiting Hamilton to discuss further collaboration towards the end of June (2017).

Technical aspects of integration

Model coupling

The level of integration is WISE is high with numerous feedback mechanisms, as is standard in Geonamica-based frameworks. It supports spatial scale-hierarchies consisting of macro-levels (regional level) and spatial grids of any desired resolution, and more generally any spatial and a-spatial models that calculate in discrete time steps. The temporal scale is adjustable and different scales for different models are handled by the application framework. Model components exchange data directly in a "type-safe, read-only manner", to ensure that a model can only alter its own variables. Mutual links between models are supported and automatically incorporated in this calculation order.

Data import and export

WISE holds all the data needed to run baseline simulations within its installation file. Data for other scenarios or analyses needs to be installed separately. There is some external GIS pre-processing required for creation of scenarios; steps are well documented in the technical manual (Appendix C of that manual). Input spatial data layers are generally stored as IDRISI files, but other ESRI and MapInfo formats are supported. Many parameters or model coefficients and user created scenarios are stored in xml-formatted files.

Key output data are graphs, maps and time-series data such as macro-economic data, land use projections and information on expected population cohorts into the future. Output formats are excel files, image files (e.g., animated gifs allowing visualisation of changing spatial patterns over time), and GIS, specifically ESRI ArcGIS and MapInfo compatible raster and vector formats, Excel spreadsheets, animated GIFs (showing changes over time for many variables)

WISE can export data for further analysis by other programs such as GIS programs (ArcGIS, MapInfo) or RIKS' Map Comparison Kit (available for free download from the RIKS website). Note that a link to the Map Comparison Kit is built in directly to WISE for ease of spatial analysis.

Geonamica is not well documented; but RIKS states it uses a range of open standards, data structures and file formats intended to allow for "maximum compatibility with outside tools and easy integration with other software". A set of software tools is available to pre and post process model inputs and manage model data over a network or the internet.

User interface

WISE is primarily accessed through a GUI that allows users to view and modify inputs such as model parameters and policy context and to define scenarios. A variety of visualisation tools are available to view maps, data tables, time series output, etc. As per the model coupling approach in WISE, this builds and tailors Geonamica's existing structure, here a library of user interface components, including map display and editing tools, list and table views and two-dimensional graph editing components. A link to a further RIKS product, the RIKS Map Comparison kit is built directly into WISE to allow spatial analysis from this interface.

Metadata, scenario management, provenance, auditability

The WISE technical manual outlines WISE overall and provides detail on each module including its input, outputs, underlying mathematical structure and links to other modules. Scenarios are managed using a RIKS tool for management, dissemination and joint development of scenarios on the basis of the Geonamica xml structure. Tools to assist comparison and parameter auditing are also available. WISE follows a versioning numbering system. As Geonamica is closed source, as are some of the component models in the WISE framework, the quality of software implementation is not fully auditable; however, developers have strong track records and quality peer reviewed publications.

Underlying language and computing requirements

WISE requires Microsoft Windows and Excel and runs on a PC. The Geonamica® framework used for WISE is based in the C++ programming language.

Uncertainty methods and calibration

WISE and Geonamica do not have calibration facilities, relying on pre-calibrated models.
Currently uncertainty exploration is limited to investigation of the impact of uncertainty in land use change. WISE 1.3 and upwards provides the land use change model with the functionality to perform a Monte Carlo analysis.

Management of IP and proprietary material

Waikato Regional Council are the custodians of WISE. They possess an ongoing license from RIKS to use the underlying Geonamica software which is closed source and proprietary. Therefore, any negotiations regarding IP or ongoing use around WISE are carried out by Waikato Regional Council and RIKS together.

Quality assurance

WISE is well documented, version controlled, and tests are done before version releases. Waikato Regional Council are active in updating the model as bugs are reported or potential inconsistencies noted. It is a highly complex and still evolving model.

As well as being proprietary/closed source, technical details of Geonamica are not well documented. It is However, delivered by a respected research institution who have been developing and applying the software around the world for over 20 years.

Does it meet the intended purpose?

WISE (version 1.0) was originally developed under objective 2 of the Creating Futures project to support and facilitate a range of integrated, long-term policy development and planning processes undertaken by Waikato Regional Council, for both statutory and non-statutory spatial planning purposes. Environment Waikato continues to support development of WISE and has used the evolving model (current version is 1.4) to project future population, households, works force and employment under a range of future climate and economic driver scenarios, assess urban sprawl impacts under different zoning rules, and examine implication of promoting “carbon farming”. Other applications include assessing urban sprawl impacts under different zoning in Future Proof area, and identifying implications of promoting ‘Carbon Farming’ on future rural land use. It has also informed development of later integrated spatial planning and scenario exploration in the Auckland and Wellington regions.

Barriers to use

A limited time trial version is available for download from the Creating Futures site.

http://www.creatingfutures.org.nz/wise/download-wise/

As noted earlier, an integral component of WISE is the commercial Geonamica software which is closed source and proprietary. Ongoing use of WISE therefore needs to be negotiated with both Waikato Regional Council and RIKS and the cost of a license to Geonamica is not clear (agreed on a case by case basis).

Flexibility, extensibility and adaptability

As model components in the underlying Geonamica program are specified in an open XML-based structure, models can be easily swapped in and out without need for changes to the software framework. Executables, dynamic link libraries, web services, OpenMI compliant interfaces or source code. This creates a wrapper for the model compliant with the interface of a Geonamica model component. It is therefore very easy to update or otherwise modify individual component models within the framework with minimal external and/or commercial assistance required.

In theory, it is also easy to modify linkages between component models, i.e., augment the conceptual and software framework to bring in further dependencies, predicted variables, drivers of change, etc. However, in practice due to Geonamica being a commercial closed source product, any such modification would require time from RIKS personnel, with associated costs.

Ease of use in particular applications

Relative to its complexity, WISE is fairly easy to use. The FIFM project suggested it would be a matter of days for most users to get up to speed on its basic application, although much longer to understand all the potential nuances of a complex integrated system, and this review agrees on that timeframe. It is well documented. Some GIS skill is required. There is some significant pre-processing of GIS and other data required to develop scenarios, but these steps again are well documented (see for example WISE v1.4 technical manual, Appendix C).

Suitability for the needs of the OLW programme

WISE does not appear suitable for the specific needs of the OLW programme, but does offer some interesting examples of good practice and potentially appealing ways forward for development of a suitable interoperable platform. There are a number of NZ-specific models within it relevant to OLW and this project in particular, as well as approaches to learn from in coupling and recognising key international as well as national drivers and their impact at farm or other small scales.

It meets the requirements of scalability, and taking information on Geonamica at face value, also provides what appears to be a highly flexible system for model coupling and interoperability. However, this would require significant time input from RIKS with associated costs, and most importantly as the framework is proprietary and closed source would place an ongoing reliance on an external party maintaining their software into the future.

Web Resources

http://www.creatingfutures.org.nz/wise/what-is-wise/
https://teamwork.niwa.co.nz/display/IFM/Waikato+Integrated+Scenario+Explorer+(WISE)
[exploration of suitability of WISE for earlier NZ interoperability project].
http://www.riks.nl/products/Geonamica

References

Rutledge, D.T., Cameron, M., Briggs, C., Elliott, S., Fenton, T., Hurkens, J., McDonald, G., McBride, G., Phyn, D., Poot, J., Price, R., Schmidt, J., van Delden, H., Tait, A., Urich, P., Vanhout, Woods, R. (2016) WISE – Waikato Integrated Scenario Explorer, Technical Specifications Version 1.4. *Waikato Regional Council Technical Report 2016/16* (Hamilton, June 2016).

Aljoufie, M., Zuidgeest, M., Brussel, M., Van Vliet, J., Van Maarseveen, M. A cellular automata based land use and transport interaction model applied to Jeddah, Saudi Arabia. *Landscape and Urban Planning,* Volume 112: 89–99.

van Vliet, J., Hurkens, J., White R., van Delden, H. An activity based cellular automaton model to simulate land-use dynamics. *Environment and Planning B,* volume 39: 198–212

Appendix Q Emerging technology for building and managing models

Overview

Extensible modelling efforts are likely to involve building, modifying and hosting software code that runs existing models. In this document, we describe a number of software engineering approaches that are gaining traction, either in terms of delivery of modelling services, or in terms of technologies that help develop systems that need to run models, for example particular software platforms that sets of models might run on. These include:

- Collections of models.
- Software container technology (*e.g.*, Docker, etc.).
- Unikernels.
- Serverless computing (*e.g.*, Amazon Web Services' Lambda product).
- Virtual Laboratories (*e.g.*, Australia's NeCTAR offerings).

While we focus on code (i.e., computational) aspects, it is key that management of the datasets that a model runs over be coordinated with the code that drives the model itself. This is because the different technologies for managing the code and the data cannot always be applied independently.

Potential to integrate models

Although it will become important to discuss how amenable models are to being integrated and/or extended, this is not a topic that can be addressed without considering the specific code of each model that needs to be integrated. This is not the focus of this document, which instead discusses particular technologies that can be applied to software more widely, but are nonetheless expected to be relevant for extensible modelling work.

Nonetheless, we note that there are some relevant survey papers that are beginning to appear on topics relevant to model interoperation, such as "An overview of the model integration process: From pre-integration assessment to testing" (Belete *et al.*, 2017). This paper covers a number of very useful points, and provides a good overview of different phases of integration and design possibilities. Nonetheless, from a software engineering perspective, the paper raises a number of red flags: some of its figures make possible integration approaches look far simpler than the reality of dealing with large sets of heterogeneous software components. While it is very relevant to promote adoption of standards, as the paper encourages, it is important to appreciate that in software systems there are many nearly-equivalent standards that continue to compete for almost no obvious reason. Software standards are cheap to create and maintain, but can be hard to position into a dominant position.

Curated collections of models

Description of the technology or system

Collections of models involve a team, possibly with members from many different organisations, collecting the software for a set of models, with the aim of making the process of building these models more convenient.

The models may already be built and hosted by that team. In such cases, it is common for a web interface to be provided to allow researchers to interact with the hosted software. Aspects of this are discussed elsewhere in this document under the heading of virtual laboratories.

What problems does it solve?

The software that embodies a given model will always need some sort of environment in which to operate. This includes the need to set the model's parameters, and a requirement for mechanisms that provide initial input data, and to monitor output produced by the model (see Belete et al., 2017 for an overview).

Collecting sets of models under the control of one team can provide an economy of scale in terms of building the models. The environment in which they run is homogeneous by virtue of the models belonging to the collection. However, it is important to note that there are many levels at which homogeneity may be achieved (or not), for example, a collection of models might allow convenient mechanisms to get to run the models, but it might not necessarily imply that the input data formats are compatible, i.e., each model in the collection may still have a degree of heterogeneity that means that data most be modified before it can be fed into different models in the collection.

Advantages over other approaches

The alternative is to have to source and build the relevant models in the collection independently. For models contained within well documented software projects (e.g., hosted on GitHub, and with demonstrators provided via software building systems discussed elsewhere in this document, such as Docker or Vagrant), the effort to build a model's software should not face technical barriers other than the need to deploy the appropriate operating system version and any other software dependencies. Nonetheless, there is still an overhead to needing to seek out separate software sources for models.

Applicability to our problem/goals (OLW objectives)

It is hard to say in abstract how effectively model collections will assist the extensibility and interoperability of modelling. It is possible that some existing collections of models will contain a subset of models that are specifically relevant and desired for use within the OLW work packages.

Even so, while running models in the collection may be well supported, it is not necessarily the case that combining those models or extending their functionally will be straightforward. The situation will, of course, be specific to the model(s), collection(s), and goals of the extension in question.

Examples of use from the literature

One example system that we have been referred to is the open source Cloud Services Integration Platform29 from Colorado State. To quote their website:

"The Cloud Services Integration Platform is a SoA implementation to offer a Model-as-a-Service framework, Application Programming Interface, deployment infrastructure, and service implementations for environmental modeling. CSIP leverages JAX-RS and OMS3, currently offering 200+ different model and data services."

So in this case, it is clear that homogeneity is desired in both the execution environment and the interface to it.

To elaborate on the terminology used in the above website extract:

- JAX-RS is a Java API for the development of RESTful Web Services. To unpack that further, REST is an abbreviation for *representational state transfer*. REST is a mechanism to facilitate software interactions using the communication protocols that support the World Wide Web. A number of the key design and implementation principles that made the web work so well for web browser use are transferred to these other machine-driven software interactions.
- OMS3 has been covered in Appendix H. Briefly, to quote NIWA's web resources:[30] "OMS3 is a modelling framework for component-based model and simulation development on multiple platforms. It is intended mainly for simulations of effects of agricultural systems on water quantity and quality."

Maturity, status, support

Each given model collection will have different project ecosystems. Naturally, the better funded and/or coordinated projects are likely to provide more useful platforms for collaboration and support. Particularly with a need to extend models, the OLW use will most probably require seeing how the collection works internally, as opposed to just using its published interface. This need to look within the model collection rapidly leads to a need for good system documentation and software engineering practices from those who developed the model collection.

Source code management systems have evolved significantly in recent years: it is now possible to manage effectively projects' code, their documentation and their project management through systems such as GitHub, Bitbucket and GitLab. Of more use in terms of model reuse are tools for automating software compilation. The aforementioned platforms provide, or provide access to, continuous integration systems. Virtualisation systems such as Vagrant, and containers (these terms and technologies are discussed below), greatly assist a project being able to share a working demonstrator, without having to directly support all interested clients' operating systems. Note that running a built model in a full virtual machine is likely to be too resource inefficient for production use, and thus a phase of transformation to specific operating platforms will be needed, However, this is usually made more straightforward when a "known good" reference virtual machine is available. Containers are designed to be able to be used in production, as discussed below.

[29] https://alm.engr.colostate.edu/cb/project/csip
[30] https://teamwork.niwa.co.nz/display/IFM/OMS3+Object+Modeling+System+v3

Containers

Description of the technology or system

Software containers (Merkel, 2014) are a form of virtualisation technology. There are many types of virtualisation, but the overriding principle of virtualisation is that a software system is being run within another software system. For example, the computers that old arcade games ran on can comfortably be simulated by modern computers at interactive speeds. As far as the arcade game software can tell, it is running on its original hardware, when this is not the case.

Full machine virtualisation is a common offering in Infrastructure as a Service (IaaS) clouds such as Amazon EC2. Users of EC2 typically create a disk image that represents, say, a Linux server. Amazon's virtualisation runs that Linux (virtual) server so that the server does not realise that it is actually running on a physical machine that is hosting many virtual servers simultaneously.

While full machine virtualisation is ideal for being able to reproduce software systems, it is also heavyweight: the virtual servers run a complete boot sequence when they are turned "on", which takes the same order of magnitude of time that the boot sequence would take on a physical machine.

Container technology changes the location at which virtualisation occurs. Instead of "booting" an operating system within every virtual machine, the containers all share one instance of the operating system. So for example, rather than three Linux virtual machines requiring the time and memory cost to start three copies of Linux, the container host will boot Linux once, and each container will just run its own processes, storage, and virtualised hardware interaction (*e.g.*, computer network activity).

Just as for full machine virtualisation, containers are an ideal way to package software for sharing, as all of the software dependencies (*e.g.*, libraries, *etc.*) can be pre-packaged into the container. However, containers can "boot" extremely quickly, e.g., within half a second or less. This means that they can be used for production systems. Indeed, Google have contributed significantly to the Linux kernel source code so that it supports containers. Google have indicated that they run all systems that interact with user requests within containers, for security isolation, and to assist with resource management and accounting—figures that they reported a few years ago indicated that they start and stop more than two billion containers each week.

What problems does it solve?

Containers allow software to be developed on one operating system environment and run very easily in another, different operating system environment. Despite providing this high degree of manageability, they can be very resource efficient, and thus usable on production systems.

Beyond the changes within the operating system that make containers work—the operating system kernel needs to be extended to understand how to manage multiple independent copies of parts of itself—container technology has typically been accompanied by good management tools. These tools simplify the process of starting, stopping, monitoring, building and sharing containers.

Advantages over other approaches

Containers evolve the virtual machine concept to allow efficient resource usage while still permitting instances of software to be isolated from each other. Containers are far quicker to start, smaller to store and easier to manage than full virtual machines.

Applicability to our problem/goals (OLW objectives)

Containers are a convenient way to package up and distribute complex software. If the OLW objectives require building a software system that many users can download and run on their own machines, containers are likely to be an effective way to do so.

Note that containers are a general software technology, and have no specific relationship with modelling code. Thus, many different software architectures could be developed for extensible modelling that use containers. It might be that the whole modelling framework is packaged up into a single container, so that stakeholders can download and run this container to access the whole modelling framework. An alternative would be to run each separate model in its own container: container technology provides convenient mechanisms to configure dedicated virtual network links between two containers in order for them to exchange data (doing so more efficiently than were they accessing each other over a real network link).

A system that runs separate models, which need to interact, in separate containers would need a mechanism to interconnect them. If their application programming interfaces are already compatible, then the virtual networks mentioned previously may be a sufficient mechanism. If translation is required between the models' data and/or parameter representations is required, an approach that can work effectively is to create yet another container that performs this coordination. The container technology itself is orthogonal to whether the modelling code provides a sufficiently powerful application programming interface to allow this interconnection to operate efficiently. To reach model code that is within a container, it will be necessary to use simultaneously both the model's API, and some form of container interface, such as a virtual network.

Examples of use from the literature

A mention of container use, Docker in particular, in the context of hydrologic models is made in Hut et al. (2017), emphasising points raised regarding the need for reproducibility in Hutton et al., 2016. Docker has often been used as a tool for rapidly and reliably reproducing software artefacts by programmers. It is important to appreciate that container technology is not a panacea: while it can collect chunks of software system very effectively, interoperation between software systems is still going to require careful design, planning, implementation and testing.

Maturity, status, support

Container technology has been used within Google for many years. The widespread popularity and use of containers was largely stimulated by the Docker software ecosystem. Docker Inc. is the company that primarily manages the Docker software, although it is open source. Docker also provides an online repository—the Docker Hub—of Docker containers: open source software can be published straight to the Docker Hub. The Docker Hub also provides structured documentation to allow users to find out configuration options that might be necessary for starting particular Docker containers to satisfy common requirements.

Docker specifically, and containers more generally, are now very widely used in production systems globally. So despite Docker only being a few years old, it can be considered a mature, stable system with strong community (and commercial) support available.

Unikernels

Description of the technology or system

Unikernels are operating systems that are custom built to run a single application. They have no capability to run any other application without being rebuilt. They are also sometimes referred to as Library Operating Systems, as this was the term used for them when the notion was first explored. The concept of Unikernels has been around for decades, but they have become practical recently. This is because (1) the interaction needs of many modern APIs is a simple set of web technologies, for example to build a RESTful interface; and (2) modern operating systems and programing languages allow for the necessary modularisation to omit unneeded OS components, which is necessary to affect the first point.

What problems does it solve (*e.g.,* overcoming current hurdles)

Unikernels solve the problem that both containers and full-machine virtualisation typically retain a vast proportion of operating system code that is not needed for a particular application to run. Unikernels provide specialisation rather than generalisation.

Advantages over other approaches

Unikernels start up even more quickly than containers (usually within the order of milliseconds), and thus can be used in even more production contexts than containers can. Because Unikernels omit features that they do not need from the underlying operating system, they are typically extremely small and resource efficient. Moreover, security holes that might exist in a whole operating system are highly likely to have been omitted from the Unikernel. The challenge is for Unikernel systems to provide an effective way to rapidly rebuild the Unikernels when software changes.

Applicability to our problem/goals (OLW objectives)

If containers are useful to OLW, then Unikernels are going to be worth considering too. They are likely to be the most efficient form of environment where clients can build software that uses reasonably general-purpose software code and dependencies, and yet will run efficiently when compiled into a Unikernel. Note that interconnecting Unikernels will usually require a similar approach to that of interconnecting containers: the Unikernel will have both an API layer for the model, and a more general layer such as a virtual network interface.

Examples of use from the literature

No explicit example of use of Unikernels in OLW-relevant literature has been identified yet.

Maturity, status, support

Unikernels are not yet a mature technology. Due to the technology being young, there are still limits in their applicability and use patterns that are not actually a function of their potential applicability in future.

Much recent Unikernel work has been centred on the University of Cambridge and spin-off companies from it. One such start-up company, Unikernel Systems, was bought in early 2016 by

Docker Inc. Thus, it seems likely that Docker containers may soon also reap some of the benefits of Unikernels, while maintaining the convenient development tools provided by the Docker ecosystem.

Serverless computing

Description of the technology or system

Serverless computing platforms allow software to be deployed that processes data without having to consider what computing infrastructure is required to support that processing. The notion of serverless computing is being promoted strongly recently in cloud computing, although it does have some connections to distributed stream processing systems too.

Usually the software that operates on a server-less computing platform will involve dataflow programming, rather than control flow programming. An example (albeit limited) of dataflow programming is how formulae re-compute their values in spreadsheets such as Microsoft Excel, Apple Numbers and Google Sheets. The re-computation of values within these systems is triggered by changes in the data within the system, as opposed to procedural code instructions.

What problems does it solve (*e.g.,* overcoming current hurdles)?

Serverless computing solves the problem of researchers needing to focus on system details related to the operation of a computer that distract them from more important programming connected to their research.

Advantages over other approaches

Serverless computing is the logical extreme of the spectrum that includes containers and unikernels: it depends on software components that should have almost instantaneous start up times. The difficulty is that a sufficiently useful set of functions has to be provided to the serverless computing components that users submit. This has typically been offered within the production environment of a commercial cloud provider rather than, say, as an open source distribution.

Applicability to our problem/goals (OLW objectives)

It is possible that some models may fit well as serverless computing components. This would allow them to recompute their output triggered on changes to model input, which might itself be piped in from a previous stage in the processing topology.

Examples of use from the literature

Serverless computing is too young (in its current form) for there to have been many uses of it for modelling work. However, one related set of technologies that may be useful when deploying researchers' code into a serverless computing environment is web-based programming interfaces. These allow the hosting of interactive high-level programming language engines within the host infrastructure, be it a serverless computing framework, a virtual laboratory system (as discussed below), etc. A specific example is the Jupyter notebook system, which was developed from the IPython system (Pérez and Granger, 2007), which facilitates programming in Python programming language (Van Rossum and Drake, 2003) through a live, web-based interface.

Maturity, status, support

As mentioned, commercial cloud providers offer serverless computing platforms, notably Amazon Web Services' Lambda system. Google and Microsoft have both scrambled to release serverless computing platforms on their public clouds also.

As an alternative, IBM has released as open source the Open Whisk system. It is possible that this might be a useful platform for OLW work, but some preliminary evaluation would be required to confirm this.

In general, serverless computing is a young technology. It is not clear yet whether it provides a sufficient amount of community support for use within extensible modelling projects. One of the main challenges on this front is that the serverless platforms need to rely on systems accessed over the network rather than, say, their own files. The object storage etc., requires additional components to be installed into any given system before the serverless platform can actually process data usefully.

Virtual Laboratories (particularly within NZ)

Virtual laboratories are not necessarily the most important emerging technology for the OLW goals, However, they are a technology accompanied by specific platforms that the National Science Challenges can use (i.e., the New Zealand eScience Infrastructure—NeSI), coupled with recent developments regionally (i.e., by Australian government organisations such as CSIRO, that are not entirely unlike NZ's CRIs). Being able to rely on other organisations to develop services may increase leverage to develop features that would be nice to have but are not critical in order to deliver extensible modelling, such as data visualisation. It is for this reason that there is so much content under the "Virtual Laboratories" section.

Also, while high performance computing (HPC or cloud) may not appear to be relevant for the OLW projects, it is important to realise that the target platform is not the only consideration, but also the software development tools and methodology that such a target provides, that may be applicable in other contexts than that particular target. That is, employing ways of writing efficient software for HPC platforms may not only be of benefit when operating on an HPC target environment. This logic applies to targets other than HPC, such as public/private/hybrid clouds, containers, heterogeneous computing with GPGPUs and FPGAs, unikernels, etc.

Description of the technology or system

The term "virtual laboratory" has been defined in various different ways. In this document, we use the term in the sense of the Australian National eResearch Collaboration Tools and Resources project's (Nectar) virtual laboratories.31 This is also the sense in which it is being used in the New Zealand eScience Infrastructure's (NeSI) plans regarding the platform refresh of their high-performance computing (HPC) facilities that is currently in progress.

Virtual laboratory systems can be considered as a type of hybrid between typical HPC infrastructure, and cloud computing. As noted above, the OLW contexts may not need HPC per se, but consider also the architecture that can be used on a dedicated non-HPC system that still impacts the necessary

[31] https://nectar.org.au/labs-and-tools/

software engineering/interoperability efforts. (With typical levels of computing resources available today, an HPC architecture can run on a single laptop. Of course, whether or not that is actually useful is a different question.)

The primary HPC aspect of most virtual laboratory systems is that large-scale computation and storage is available to support the virtual laboratory. This processing is likely to involve scheduled, queued batch processing, as this is the way in which existing HPC systems typically work. Results from the processing will be stored on large, shared storage infrastructure. However, the user interface, which is most often provided through a web interface, is not a batch process; its persistent presence on the web is more similar to the infrastructure that cloud computing supports. Virtual laboratory systems are often implemented by adding to the existing HPC computing cluster the ability to host persistent virtual machines that have access to the HPC cluster's high-capacity, high-scale distributed communication and storage infrastructure.

What problems does it solve (*e.g.,* overcoming current hurdles)?

The goal of virtual laboratory systems is to ensure that appropriately developed model code can be configured to use the virtual laboratory interface for multi-user collaboration, while avoiding the need to repeatedly provision computational and storage resources to support the model's operation.

The models that might fit within a virtual laboratory (NeSI or otherwise) need to have components with well-defined APIs, but otherwise the hosting environment will not constrain the types of models that can potentially be deployed—see above regarding the same concerns for the "collections of models" section. However, for models to be most easily managed, and to run most efficiently will impose more constraints on how they are built. As discussed elsewhere in this document, the interaction with stored data for input and output may have a significant effect on the efficiency. Models that are implemented using software stacks that are directly supported by the virtual laboratory will avoid the inefficiency of needing to run the models in some sort of "boxed" form. One way that this can be achieved is by hosting a programming language environment on the virtual laboratory servers, for example hosting the sorts of interactive Jupyter notebooks discussed above in the "serverless computing" section. Hosting Jupyter notebooks on the virtual laboratory servers allow a convenient programmable interface that also has high-speed access to the complete underlying datasets. Jupyter notebooks are provided as a query interface to the Australian Geoscience Data Cube32, although it is not specifically for modelling, in this case.

Data storage, import and export

Virtual Laboratories will not directly constrain the types of files used, and typically provide use of a large-scale, high-performance, distributed storage infrastructure such as GPFS (Schuck and Haskin, 2002), Luster (Braam, 2004), Ceph (Weil et al., 2006). In general, many software systems are moving to the use of object stores (e.g., Amazon S3, and other components that support the S3 REST API interface, or interfaces similar to it.) since that suits the software's requirements better than a traditional operating system filesystem.

Older modelling software is likely to need to be optimised for use of a network filesystem or an object store. If necessary, deployment on the cluster can provide local, filesystem storage on each

[32] http://www.datacube.org.au/

computational node. However, the cluster network filesystem is itself designed for very high speed access, so use and management of local filesystem storage ideally will be unnecessary.

As an example of the practice typical of HPC facilities, NeSI allocates storage within a “project” that will have an amount of persistent storage on the network filesystem connected to it. Note that the NeSI platform refresh is intending to create a new management layer that can group related projects — let’s call it a “programme”. A virtual laboratory framework would exist at the programme level, and would have resources allocated at that level. Specific users’ investigations, or particular service implementations might well be defined as projects within such a programme.

Advantages over other approaches

Just as for the “collections of models” section, virtual laboratory systems increase the separation of concerns within the software system: the platform itself can be managed independently of the code it runs or the users who use it. However, virtual laboratories are likely to further increase the number of types of stakeholder compared to the “collections of models” approach, since the service is normally hosted, allowing a separate system administration team from the programmers and users of models.

Visualisation

Virtual laboratory systems are likely to need to split the code performing analyses, and the systems that are presenting the web-based output of use of the system. Virtual laboratories aim to do visualisation as close to the computational work as possible, but will usually separate these functions in the software implementation.

The need to support visualisation is well understood by NeSI, and is a priority of the current NeSI platform refresh. In the past, NeSI included access to a visualisation cluster based at the University of Canterbury. This system provided a remote desktop display over the high-speed REANNZ network. In the platform refresh it looks like the replacement visualisation facilities are going to operate within the main computing cluster, rather than as a separate facility.

User interface

The user interface of modelling frameworks will need to be able to work within the virtual laboratory system’s cluster. Typically, these computational computing nodes will have no graphical display, and will support network-based command line interfaces only. This command-line focus has is seen as a barrier to use of HPC facilities, spawning initiatives such as Software Carpentry to try to address the skills gap regarding researchers’ use of command-line tools.

The virtual laboratory user-facing view is most likely to need to be web driven. There are projects that can facilitate building REST APIs33 rapidly that then can themselves allow command-line software to be made accessible using web technology. The work to build a complete web interface is more arduous. The likely cost of any of these sorts of software engineering must be considered carefully.

[33] http://swagger.io

Applicability to our problem/goals (OLW objectives)

The virtual laboratory concept, if it can be made to work with the specific modelling software that is required to interact, will be extremely suitable to the needs of the OLW programme. In particular, it factors out significant volumes of hardware and software management tasks, leaving researchers to focus on research questions being answered by querying analysis-ready data. It should also facilitate the deployment of appropriately engineered models into the framework easily.

The architecture of virtual laboratory infrastructure may impose constraints on the forms of data handling that will work most efficiently. For example, modelling frameworks that manage their own storage within a filesystem may or may not work well, depending on the match to the filesystem facilities provided by the platform. Modelling frameworks that expect to rely on a geospatial database or some other separate software component to manage their data will need to consider how to effect their requirements, such as (1) to ensure that there is sufficient data reduction leading to the final results to be able to run the storage component on the persistent VMs that serve the virtual laboratory user interface, or (2) to deploy instances of the storage component within the computing cluster as a batch job, alongside the batch processing of the modelling computation, i.e., instances of the model and instances of the storage component supporting the model can be deployed in parallel within the nodes in a computing cluster. This second approach would require the final output from the model to be transferred off the cluster nodes before the batch job finishes.

Does it meet the intended purpose?

NeSI's incorporation of virtual laboratory support within their platform refresh is a strong indication that virtual laboratories have been found elsewhere to boost the accessibility and usefulness of HPC infrastructure. As discussed above, the Australian NeCTAR framework has developed support for more than ten virtual laboratory systems since it began operation in 2011. Without examining its impact on policy-level decision making, it appears to have met many of its technical goals of collocating the significant volumes of data and computational needs, alongside providing a convenient user interface for researchers.

The notion of virtual laboratories has matured effectively in the bioinformatics space. This has been partly due to the large reduction in data volume from source data to results of queries and analysis: it thus is much more effective for users to query a remote system that does computing near the data than for them to download the source data sets and run computations themselves. The NeCTAR Genomics Virtual Laboratory (GVL) states that it has achieved "taking the IT out of bioinformatics". Early indications are that NeSI plans to develop a similar virtual laboratory in the genomics space.

More closely related to the OLW programme is the NeCTAR Virtual Geophysics Laboratory (VGL) being led by CSIRO. One of the features that it highlights is its "run your science" facility. This sort of facility provides a means to upload and run model code on the virtual laboratory's high performance computing systems.

Barriers to use

The virtual laboratory concept itself is agnostic to intellectual property considerations, However, the likely need to build the software on the target computing platforms will often lead to open source software being favoured. This is true of both the application software and the underlying operating system software. Having said that, Microsoft's recent focus on cloud computing is leading to more flexible operating system licensing models, for example, the support for virtualisation within Microsoft Windows systems without incurring license costs.34

Software licence management is generally a pain-point within outsourced computing infrastructures, and thus is a concern inherited by virtual laboratory platforms—at least for those setting up the virtual laboratory platforms in the first place. Since the computing components within HPC and cloud computing are reused for different jobs it is not straightforward to identify particular machines that should contain static licences. On the other hand, managing 'floating' licensing typically requires connectivity between the outsourced computing nodes and a licence server. This is often not ideal regarding the network security within the cluster system. However, use of floating licenses has been shown to work for deploying MATLAB code on the NeSI Pan cluster.

One of the largest barriers to use for virtual laboratory frameworks is likely to be need for software reengineering. Many stand-alone applications are not immediately suited to deployment on the platforms that run virtual laboratories. Beyond getting the software to run in that environment, the primary challenge is the efficient operation of the software: well implemented API use must be made for these platforms to perform optimally.

Also, the split between computing systems that perform computation, and the virtual machines that host the persistent user interface may lead to responsiveness problems: while the underlying computing facilities will be powerful enough to execute large models, it will not be possible to always initiate back-end computing with the low latency required for interactive user interfaces, depending on the design of the virtual laboratory framework.

Flexibility, extensibility and adaptability

In terms of extensibility of models, it would be instructive to compare the capabilities of the Australian Geoscience Data Cube with the virtual laboratory connected to the AWARE EU project (Granell et al., 2009). API functionality in general, and support for geospatial applications in particular, have evolved significantly in recent years. Standards published by the Open Geospatial Consortium (OGC) are likely to be used in the majority of projects, including these two.

In general, virtual laboratories being built over complex software stacks (such as those used in HPC facilities) is both an advantage and a disadvantage in terms of adding or modifying components. Advantages include the ability to gain high performance, scalable systems, for example using storage and networking within an HPC facility. The abstraction of application functions away from implementation mechanisms is usually well defined. For example, systems developed using the Message Passing Interface (Gropp, Lusk and Skjellum, 1994) are likely to be able to run efficiently on a variety of different actual HPC platforms and configurations (even on high-core-count single machines). However, MPI is almost certainly not the form of intercommunication than an existing piece of modelling software is likely to use if it was not designed from the beginning to run on HPC infrastructure. Skimming over the modelling literature relevant to OLW, a number of models appear to be distributed as (Microsoft Windows) DLLs, which may make use of the software difficult in a cluster, unless those who developed the models had the foresight and/or freedom to maintain a compatibility layer (e.g., running Windows code on Linux using WINE).

[34] https://www.microsoft.com/en-us/licensing/product-licensing/innovations-for-the-cloud.aspx

In an ideal case, the modelling code being deployed uses support libraries that themselves can be built to run efficiently on whatever computing infrastructure is provided. This is particularly true of efficiently using CPUs with very large numbers of cores, and of heterogeneous computing over general purpose graphics processing units (GPGPUs), and other forms of emerging hardware accelerator devices such as field programmable gate arrays (FPGAs). For example, OpenCL (Stone, Gohara and Shi, 2010) attempts to provide a programming abstraction that can run efficiently on all the aforementioned types of heterogeneous computing infrastructure.

Ease of use in particular applications

The virtual laboratory concept is highly likely to be coupled with the emerging push for "analysis ready data". As the term suggests, data alignment and cleaning is done once and interested users are expected to be able to interact with the system without needing to become technically expert in its underlying engineering. It is extremely likely that compelling datasets are made immediately available to users, and the effort of combining information from multiple datasets can be expended once by the technical teams supporting the infrastructure. However, this also highlights that the scale of computing is likely to require a carefully managed facility such as those run by NeSI, including application specialists and other programmers who have experience in optimising software for the HPC infrastructure that they manage.

By using shared data, and avoiding redundant data cleaning activity, virtual laboratories are likely to offer significant economy of scale to researchers.

Model coupling

A virtual laboratory approach does not directly influence the coupling of models implemented within it. Any given model or integrated set of models should be able to be engineered to run on an HPC cluster. A key concern, though, is how efficiently the system works. In terms of NeSI, the computational cluster's nodes are interconnected by both InfiniBand and Ethernet networks, which are will quite often be used for tight and loose coupling, respectively. Reengineering a model that was not written to run on an HPC cluster is likely to be a reasonably expensive process both in terms of time and effort.

Irrespective of efficiency, just regarding operation, the virtual laboratory framework will not limit addition of new model components if the modelling system itself supports this extensibility.

Metadata, scenario management, provenance, auditability

There are no specific technical restrictions on how modelling can support metadata and management of sets of experiments. In terms of commercially sensitive data, however, shared computing environments will not typically provide service level agreement (SLA) guarantees regarding security, and the NeSI HPC facilities are no exception. For resource efficiency, the sharing of infrastructure is the foremost consideration compared to security isolation. The situation for commercial cloud computing is less clear-cut, with an increasing number of organisations, for example Revera35 in New Zealand, providing SLA guarantees that do cover security of commercially sensitive data.

Underlying language and computing requirements

[35] https://www.revera.co.nz

There are no specific constraints on the type of code that the modelling framework uses, provided that the software environment required to build the model's code is supported on the shared computing environment, so that the software may be compiled and run there.

The operating system of the majority of shared computing facilities is Linux, and this includes the target NeSI clusters. Nonetheless, Microsoft Windows systems are available within the commercial cloud, and within other virtualisation systems that the OLW host organisations will be able to provide.

Uncertainty methods and calibration

The virtual laboratory system itself will be agnostic as to whether or not calibration and uncertainty are directly supported in the model or not. Modelling in a virtual laboratory is likely to facilitate access to larger volumes of data storage than stand-alone infrastructure, which may have an impact on the support for tracking uncertainty in models, should this increase the storage requirements of intermediate modelling results.

Management of IP and proprietary material

It is typically preferred that non-commercial software be used within virtual laboratories, since this simplifies the process of building, running and optimising the software. Given the popularity of working with models in proprietary systems such as MATLAB, though, clusters are likely to have experience with how to share licenses from users (or their institutions) into the computing resources of the cluster. This has been done for MATLAB successfully on the NeSI Pan cluster.36 The openness of the code otherwise entirely depends on what system is deployed.

Update cycles of code are likely to be best handled through a version management system, and the server nodes of a large-scale model deployment can be given a deployment key: this means that the computational nodes can access repository data efficiently (e.g., to synchronise a set of new files). This type of file management is usually able to be wrapped around existing modelling software, or to be incorporated within it directly, at a fairly low software engineering cost.

Quality assurance

The testing for quality assurance of the modelling within a virtual laboratory system does not have a direct bearing on the virtual laboratory infrastructure. The need to compile the modelling software for the virtual laboratory may give useful insight into the model's software dependencies, such as the libraries that is uses. However, it is likely that the core modelling code itself may remain opaque throughout the software building process.

Examples of use from the literature

Many existing virtual laboratory systems, such as the WebMARVL web-based marine virtual laboratory (Oke et al., 2016) and others hosted on the Australian NeCTAR platform appear to expect that modelling code has already been installed on the computers behind the virtual laboratory's user interface. The container technologies discussed above may be relevant for deployment of software to platforms such as those offered by NeCTAR, not least in that they can provide security isolation of users' software from the underlying virtual laboratory framework. Certainly, NeCTAR already offers Australian researchers a number of different facilities for hosting virtual servers to be used for

[36] https://support.nesi.org.nz/hc/en-gb/articles/212639047-MATLAB

research. (The design of WebMARVL does not make it immediately clear that containers would be a good match—they may still be too heavyweight.)

Maturity, status, support

The New Zealand eScience Infrastructure (NeSI) platform refresh[37] that is due to complete at the end of 2017 includes the goal of introducing a virtual laboratory facility. Modelling appropriate to the OLW programme is a potential application of a NeSI virtual laboratory, assuming that modelling code can technically and legally run on NeSI computing facilities, and that the data can be stored within the NeSI storage infrastructure. The NeSI infrastructure itself is readily accessible to New Zealand Universities and CRIs.

The virtual laboratory framework will usually be supported by the owner of the physical resources. This will include managing the user-facing VM support (e.g., through the web), the HPC cluster filesystem and the HPC cluster computing resources. This holds true in the case of the plans for the NeSI platform refresh, although given that that service has not yet been launched, the specifics of the final result cannot be assumed. NeSI always provides technical support from low-level system management concerns right up to high-level scientific programming concerns: it is a key priority of NeSI to do so, i.e., to provide expert advice rather than just access to technology.

The virtual laboratory environments themselves are likely to be managed mostly by the consortia that cause them to be set up in the first place, in collaboration with those that manage the HPC infrastructure used.

References

Get chew F., Belete, Alexey Voinov, Gerard, F. Laniak. (2017) An overview of the model integration process: From pre-integration assessment to testing. *Environmental Modelling and Software* 87: 49–63. doi:10.1016/j.envsoft.2016.10.013

P. J. Braam. (2004) The Lustre storage architecture. http://www.lustre.org/documentation.html, Cluster File Systems, Inc.

Carlos Granell, Laura Díaz, and Michael Gould. (2010) Service-oriented applications for environmental models: Reusable geospatial services. *Environ. Model. Softw.* 25, 2 (February 2010), 182–198. doi:10.1016/j.envsoft.2009.08.005

William Gropp, Ewing Lusk, and Anthony Skjellum. (1994) *Using MPI: Portable Parallel Programming with the Message-Passing Interface*. MIT Press, Cambridge, MA, USA.

Hut, R. W., van de Giesen, N. C., Drost., N. (2017) Comment on “Most computational hydrology is not reproducible, so is it really science?” by Christopher Hutton et al.: Let hydrologists learn the latest computer science by working with Research Software Engineers (RSEs) and not reinvent the waterwheel ourselves. *Water Resource. Res.*, 53, doi:10.1002/2017WR020665

[37] https://www.nesi.org.nz/services/high-performance-computing/platforms/national-platforms-framework

Hutton, C., T. Wagener, J. Freer, D. Han, C. Duffy, Arheimer B. (2016) Most computational hydrology is not reproducible, so is it really science?, *Water Resource. Res.*, 52, 7548–7555, doi:10.1002/2016WR019285

Dirk Merkel. (2014) Docker: lightweight Linux containers for consistent development and deployment. Linux J. 2014, 239.

Nativi, S., Mazzetti, P., Geller, G.N. (2013) Environmental model access and interoperability: The GEO Model Web initiative. *Environmental Modelling & Software*, 39: 214–228. http://dx.doi.org/10.1016/j.envsoft.2012.03.007

Oke, P. R., Proctor, R., Rosebrock, U., Brinkman, R., Cahill, M. L., Coghlan, I., Divakaran, P., Freeman, J., Pattiaratchi, C., Roughan, M., Sandery, P. A., Schaeffer, A., Wijeratne, S. (2016) The Marine Virtual Laboratory (version 2.1): enabling efficient ocean model configuration. *Geoscientific Model Development*, 9(9), pp 3297–3307. doi:10.5194/gmd-9-3297-2016

Pérez, F., Granger, B. E. (2007) IPython: a system for interactive scientific computing. *Computing in Science & Engineering*, *9*(3).

M.B.J. Purss, A. Lewis, A. Ip, L. Lymburner, S. Oliver, *et al.* (2015) The Australian geoscience data cube. S. Morain (Ed.), *Manual for remote sensing* (4th ed.) American Society for Photogrammetry and Remote Sensing (ASPRS).

Van Rossum, G., Drake, F. L. (2003) *Python language reference manual*. Network Theory Ltd.

Frank Schmuck and Roger Haskin. (2002) GPFS: A Shared-Disk File System for Large Computing Clusters. In *Proceedings of the 1st USENIX Conference on File and Storage Technologies* (FAST '02). USENIX Association, Berkeley, CA, USA.

Stone, J. E., Gohara, D., Shi, G. (2010) OpenCL: A parallel programming standard for heterogeneous computing systems. *Computing in science & engineering*, *12*(3), pp 66–73.

Sage A. Weil, Scott A. Brandt, Ethan L. Miller, Darrell D. E. Long, and Carlos Maltzahn. (2006). Ceph: a scalable, high-performance distributed file system. In *Proceedings of the 7th symposium on Operating systems design and implementation*. USENIX Association, Berkeley, CA, USA, pp 307–320.

Appendix R Internet of Things

Management of land and water is a complex paradigm which has been the focus of decision-makers for decades. Different dimensions are normally considered in the management of land and water. These dimensions include environmental, social, cultural and economic aspects. Some of the complexity of the decisions makings can be attributed to the fact of the underlying processes governing these dimensions are not very well understood and they have high degrees of spatial and temporal uncertainty. The high degrees of spatial and temporal uncertainty would inevitably translate to uncertainty in the spatial sand the temporal decisions being transmitted to end-users.

Although the management land and water is a complex paradigm, at a higher conceptual level the management decision framework is very simple. The framework involves the following steps:

1. Gathering data from the real environment and storage.
2. Data processing, visualization, modelling and knowledge discovery.
3. Decision making and transmission of decisions to end-users.

Although the main focus of this report is the interoperability of the substantive environmental models, it is paramount to holistically examine interoperability in the context of interactions with data collection and transmission of results to end-users within the land and water management decision framework. In recent years, considerable advances have been made in the developments devices that can gather environmental data where the gathered data can be accessed remotely in real-time via a web interface. Likewise, models can be executed via interface and their results can also be viewed via web interface. Thus, the interoperability of the substantive environmental models cannot be viewed in isolation without regard to data gathering mechanisms and communication tools of models results to end-users.

An emerging technology which encapsulates at the conceptual level the steps involved the management decision framework for land and water is the Internet of Things (IoT). There are many definitions for IoT. For example, The Oxford dictionary defines the IoT as "*The interconnection via the Internet of computing devices embedded in everyday objects, enabling them to send and receive data*". The International Telecommunication Union (ITU) defined as "*a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies"* (ITU, 2012).

In a research project report published by the European Research Cluster (IERC), IoT is defined as "*A dynamic global network infrastructure with self-configuring capabilities based on standard and interoperable communication protocols where physical and virtual 'things' have identities, physical attributes and virtual personalities and use intelligent interfaces, and are seamlessly integrated into the information network" (IERC, 2014).* This IoT definition is further elaborated in Figure (1). This above definition of IoT shows the interoperability is a subset the IoT system.

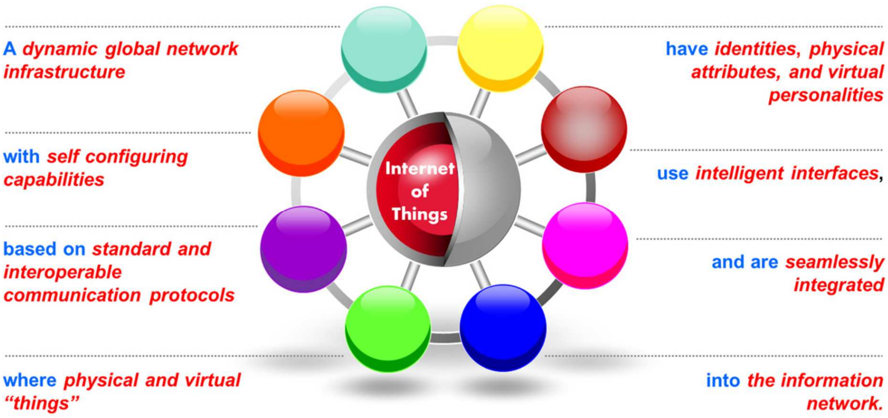

Figure (1): IoT definition (http://www.internet-of-things-research.eu/about_iot.htm)

The impetus to the development of IoT is the high demand for resource use and its efficient utilization (Vijai and Sivakumar, 2016). Organizations around the world are keen to maximize the efficiency of resource use of scare resources such as water. Accordingly, concepts of SMART cities and smart water management have emerged and implemented in many countries around the world. SMART city IoT applications include smart environment, smart governance, smart people, transportation and emission, energy, public health, smart mobility, water and waste management and smart economy. Figure (2) shows the various pillars and smart city applications of IoT. This concept can also be adapted to yield SMART land and water management.

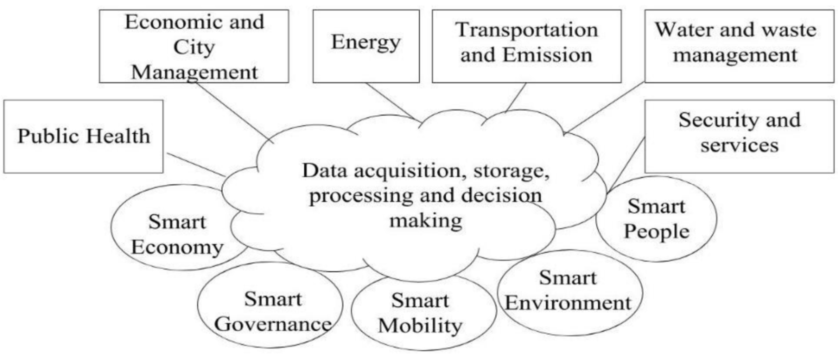

Figure (2): pillars and smart city applications of IoT after (Vijai and Sivakumar, 2016).

Commented [AS1]: Reference not supplied

The concept is currently gaining attention in New Zealand. There is an on-going project in Christchurch called "Sensing City". The Christchurch rebuild provides opportunities for "*the* *integration of a network of digital sensors into the physical infrastructure of the Christchurch CBD as it is rebuilt, generating data sets with multiple uses and benefits*" (Ref).

Vijai and Sivakumar (2016) provide the following standard stages to be adopted in the IoT system setup:

- "The sensor that collects the data from the environment (including the identification address of the sensor).
- An application that is used to collect and analyse this data to infer knowledge from it.
- Decision making and transmitting of information to the necessary hubs. Actuators and big data analytics is used for the same."

Vijai and Sivakumar (2016) noted that "*the main requisite for an IoT system is the solution should be usable for everyone and not just an expert*". The above generic steps used in the IoT are broadly similar to those associated with the land and water management decision framework. Thus, the IoT system can be used within the land and water management decision framework.

In the context of IoT system, traditional and artificial intelligence data driven models play a considerable role in knowledge discovery particularly in the case of complex systems where the underlying physical processes are not very well understood. In the context of land and water management, data-driven models can be used for water demand forecasting, water quality monitoring and pattern recognition.

The IoT technology is an emerging field and it is rapidly growing. It has been noted by that in 2011 "*the number of Internet-connected devices surpassed the number of human beings on the planet in 2011, and by 2020, Internet-connected devices are expected to number between 26 billion and 50 billion*" [reference not provided]. The future evolution of Internet-connected devices is shown in Figure (3).

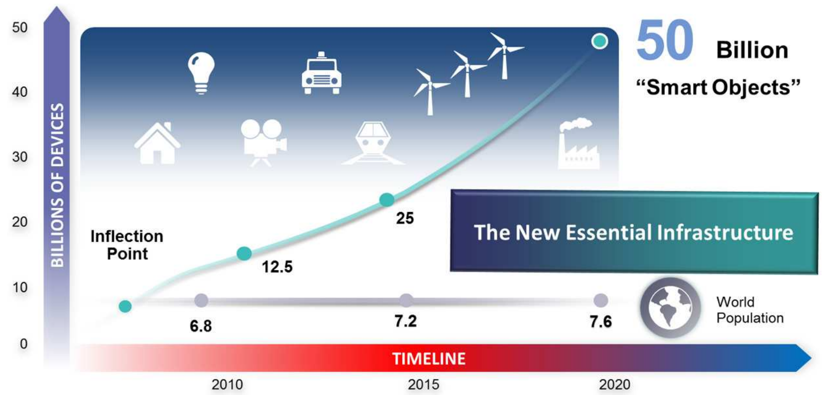

Figure (3): future Evolution of Internet-connected devices source

James (2014) discussed the concept of connected farm. It an IoT application in farm management developed in order to improve agricultural productivity. Figure (4) shows a schematic diagram of the connected farm which has the following main components:

- Precipitation monitoring: Precipitation is measured using high tech devices. The measured precipitation data is accessed through the internet by the farmhouse computer. The data is analysed and subsequently to support the farm management.
- Irrigation Monitoring: Irrigation systems are connected and are controlled remotely via the internet. Computer simulations model are then used to determine the irrigation requirements. Once the irrigation water requirements are determined information is transmitted to the irrigation systems to control the amount of water being released.
- Field activity: Various devices are used to gather information about the different farm activities such as fertilizing and seeding. The gather information is the transmitted and viewed in real-time through a dash board.
- Office to Field to field data exchange: This provides the farmers with the capability to transmit data from the office to devices placed in the field to change guidance lines and drainage designs.
- Monitored and connected tractors: Monitoring of tractors provides the farmers with the ability to get real-time information about fuel usage, batter voltage, and movement. Connected tractors provide an auto-pilot system using pre-sets condition to maximise farm profitability.
- Connected Cows which provide the framers the opportunity to monitor and track cows in real-time.

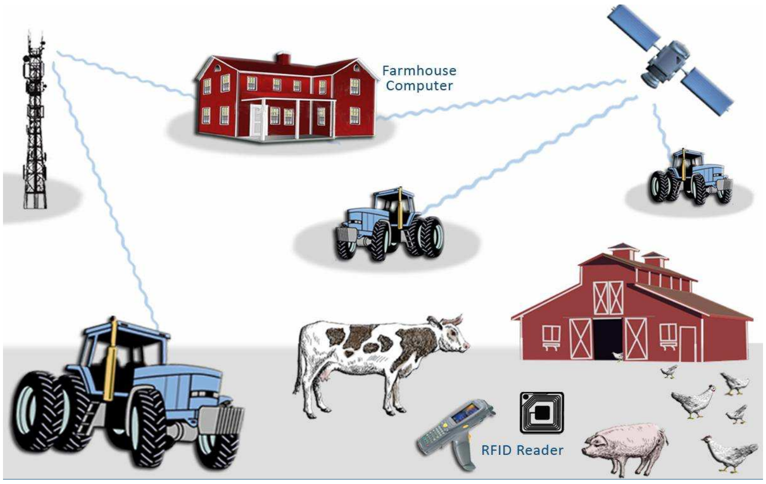

Figure (4): Connect farm after (James, 2014).

Hyde and Spargo (2014) discussed the use of IoT in the context of Resource Management Act in New Zealand. The noted that "There is rapidly advancing ability to remotely view, monitor, analyse and interact with our environment via the web. This strengthening of the information base, tools to interpret this information, and general access for the majority to the web poses both opportunities and challenges". Furthermore, they have listed are which requires attention by the RMA professional. These areas include:

- "seek to understand and monitor the existing environment
- source our tools and experts (e.g., deal with the emergence of global scale providers of „big data‟
- cloud based tools, and scope for more use of global gurus „beaming in‟)
- develop and hypothesize future states (scenarios), and assess and model these
- engage and communicate with communities and seek their values, and
- interact with stakeholders across the spectrum of RMA processes from „visioning‟ exercises, to hearing processes, to practical implementation, monitoring and enforcement".

References

Hyde, T. and Spargo, G. (2014) Smarter, easier, faster – innovative technologies to manage water, land use and adaption. NZPI Conference 2-5 April 2014 https://www.planning.org.nz/Attachment?Action=Download&Attachment_id=3111

James, R. (2014) The Internet of Things: A study in Hype, Reality, Disruption, and Growth. Raymond James US Research, Technology & Communications, Industry Report.

Vijai, P., Sivakumar, P.B. (2016) Design of IoT Systems and Analytics in the Context of Smart City Initiatives in India. *Procedia Computer Science*, 92: 583-588. http://dx.doi.org/10.1016/j.procs.2016.07.386

Appendix S List of model variables

See table on the next page.

Objective Class	Objective	Variable to predict	Time and spatial resolution and scale	Critical data requirements
Production and efficiency, resource use	pasture production	growth rates for different grasses	In growing season, and change between growing seasons.	climate, rainfall, soil type, landform (e.g., slope); Primary land-parcel data is available to use with this. This should let us drill down to individual farms in terms of spatial resolution.
	pasture production	pasture growth	Inter-annual variability	As with other production aspects
	Farm production	wool production (kg/ha/yr)	paddock level	
	Farm production	meat production (kg/ha/yr)	paddock level	
	Farm production	milk solids (kg/ha/yr)	paddock level	
	Farm production	crop (various species) production (t/ha/yr)	paddock level	
	Farm production	timber production (t/ha/yr)	stand	
	Farm production	honey production (kg/ha/yr)	paddock level	
	Resource use	fertiliser requirement (N,P) use (kg/yr)	annual at farm level	
	Resource use	water use (m3)	annual / seasonal at farm level	
	Resource use	GHG emission (t CO2 eq./yr)	farm level	
	Resource use	electricity consumption (kWh/yr)	farm level	
	Forest production	Total standing volume (TSV) per ha by age class (m3/ha)		Both land cover database version 4 and primary parcel data has the extent of relatively current production forestry parcels. The only real difficulty lies in getting the age of the stands from stakeholders.

Objective Class	Objective	Variable to predict	Time and spatial resolution and scale	Critical data requirements
Production and efficiency, resource use	Forest production	Total standing recoverable volume (TRV) per ha by age class and by log grades (m3/ha)		Both land cover database version 4 and primary parcel data has the extent of relatively current production forestry parcels. The only real difficulty lies in getting the age of the stands from stakeholders.
	Forest production	Total stand carbon by IPCC pool per ha by age class (tCO2/ha)		Ditto
	Forest production	Total stem biomass by age class odt/ha		Ditto
	Forest production	Annual harvest area (ha/year), volume (m3/ha), biomass (t/ha)		
	Forest production	Product quantity (units/ha/year by age class)		Both landcover database version 4 and primary parcel data has the extent of relatively current production forestry parcels. The only real difficulty lies in getting the age of the stands from stakeholders.
	Irrigation water supply, streams	Flow available for abstraction, and associated reliability measures. Demand for determining reliability.	Daily (and various aggregations) stream flow, groundwater levels	Hydrological model inputs, abstractions database, flow database, storage parameters, distribution system, groundwater model
	Irrigation water supply	Available volume of supply	Mean annual, variability.	Hydrological model inputs, abstractions database
	Irrigation demand	Demand for water	Mean annual, seasonal variability.	Crops, irrigation system, efficiencies, schedules

Objective Class	Objective	Variable to predict	Time and spatial resolution and scale	Critical data requirements
Environmental	Swimmability	*E. coli*/100 ml	95%ile all times and flows or during Bathing season (Nov-April?)/excluding floods (flows exceeded 10% of time)?	Data collected by councils. Perhaps possible to derive river-specific data
	Wadeability	*E. coli*/100 ml	Annual median all times and flows	Data collected by councils. Perhaps possible to derive river-specific data
	Swimmability and life-supporting capacity	Water clarity as black disc visibility	Annual median and during Bathing season (Nov-April?)/ excluding floods (flows exceeded 10% of time)?	Regress turbidity against water flow (relates to sediment run-off)
	Provide life-supporting capacity in streams	Nitrate-N (toxicity)	Annual median and annual 95%ile	
	Provide life-supporting capacity in streams	Ammonia-N (toxicity)	Annual median and annual 95%ile	Ph-dependent variable. Only toxic when Ph is high.
	Provide life-supporting capacity in streams	Lake TP and TN	Annual medians	N & P Budgets
	Provide life-supporting capacity in streams	Phytoplankton Chl a NOF bands for seasonally stratified and polymictic lakes	Need annual medians	Derive from wq simulations would be one way to go.
	Provide life-supporting capacity in streams	Planktonic cyanobacteria biovolume equivalent for the combined total of all cyanobacteria OR of total cyanobacteria	Lakes and lake fed rivers, 80%ile	Derive from wq simulations would be one way to go.
	Provide life-supporting capacity in streams	low flow (m3/s)	Mean annual low flow or similar (catchment / reach)	groundwater level and location and timing of groundwater-surface water interaction

Objective Class	Objective	Variable to predict	Time and spatial resolution and scale	Critical data requirements
Environmental	Provide life-supporting capacity in streams	flow habitat for fish etc.	Optimum or minimum? annual low flow or similar (catchment / reach)	IFIM? Optimal flow conditions or minimum requirements? Need to strike a balance.
	Provide potable drinking water	groundwater level	seasonal estimates of mean groundwater level	
	Provide potable drinking water	groundwater quality	long-term estimates of the median concentrations	
		P run-off	paddock level/farm level	GIS Data layers: land-use proxies to generate the necessary data.
	land productivity	sediment yield (t/ha/yr)	catchment	GIS Data layers: land-use proxies to generate the necessary data.
	water quality	sediment load (t/yr)	catchment, reach	GIS Data layers: land-use proxies to generate the necessary data.

Appendix T Optimisation

Introduction

In this section, we discuss aspects of spatial optimization, uncertainty, and parameter estimation.

The "Ideal" (spatial) optimization model has the following attributes:

- Spatial – Besides the most common/default features, adjacency (or compactness) constraints to cluster common land uses would need to be included in an ideal model. These constraints facilitate discovery of advantageous contiguous placements of common land uses and/or management options spatially, so that solutions with large contiguous blocks (as in amalgamation of several polygons) are preferentially obtained, for economies of scale.
- Dynamic – this is key for many reasons, e.g., to include forestry as a land use, not only due to the time dimension of cash flows but also the time dependencies of biological phenomena such as contaminant generation or, more importantly, carbon sequestration (not relevant in this project though). The erosion rates from forests increase when harvested, otherwise they are very low when the trees are growing. Economic land-use optimization models in NZ have treated forestry as an annual enterprise disregarding the time dimension. Similar arguments apply to farming where system and seasonal management changes have a large input on biophysical outcomes as well as economic impacts.
- Uncertainty – This should be considered in three different fronts:
    - Biogeophysical modelling uncertainty. This includes (among many others): uncertainty in spatial and temporal farm and forest productivity estimates, water flow, contaminant generation, attenuation mechanisms and ecosystem service uncertainty. There are a wide variety of methods available to help characterize and/or reduce uncertainty in biophysical models; often divided into methods to characterize or constrain 1) input uncertainty (with methods further split into those dealing with parametric and other input uncertainties such as driving rainfall, spatial soils data, etc.), 2) structural (model) uncertainty and 3) output uncertainty, which again can be split into estimates of predictive model uncertainty vs estimates of constraining data /observational uncertainty which may then be used to re-examine aspects of input or structural uncertainty.
    - Uncertainty around the prices of products obtained from productive land (e.g., kg of milk solids, cubic meters of timber, kg of red meat, etc.). Such prices are highly volatile (e.g., milk price in the last decade) and we cannot keep relying on average prices as has been done in NZ so far.
    - Decision-making under uncertainty component, which in economics is dealt with "empirical utility" functions that are not only functions of profits but also of risk aversion and other types of preferences (remember that decisions are not always taken based on profits). To date, most of the economic land-use optimization models take average prices disregarding such volatility.
    - Correlation of uncertain variables through the use of copulas and multivariate probability distribution functions. Now the availability of copulas to correlate a

whole range of non-normal distributions are available in Python, R, C++, etc. We cannot keep assuming that uncertain variables are independent and normally distributed.

- Multi-criteria – This should also be considered in two different fronts:
    - Biological or ecological optimization: Maximizing ecosystem services or minimizing contaminant generation, which may or may not be subject to specific constraints (e.g., maximize productivity while remaining under set N and P targets).
    - Economic optimization: The criteria should be the maximization of a decision-maker's utility function, which is a more comprehensive measure including profits and other preferences such as risk aversion. So far, this type of optimization has been profit-based disregarding decision-making under uncertainty.
- Conclusion: Considering all of the previous "ideal" components to include in spatial optimization model, the best candidates would be spatial models that rely on heuristic principles (e.g., genetic or evolutionary algorithms) as their optimization component. (Rabotyagov et al., 2016; Chikumbo et al., 2014; Lehmann et al., 2013; Musshoff et al., 2009). Most of these rely on guided searches and therefore the incorporation of a range of "objective functions". There are various colours and flavours of these algorithms and most of them are available as open source in Python, R, C++, C#, etc. Others are available at a cost for Matlab.

What is currently happening in NZ

Economics

- Optimisation principles: Currently in NZ economic aspects of land-use change optimisation are mostly dealt by mathematical programming approaches – linear and non-linear programming. The former assume that product prices are constant and are treated as an exogenous parameter into the model (e.g., Graeme Doole's model also known as LAM). The latter, known as "partial equilibrium" models, assume that product prices vary, are endogenous to the model and are conditioned to supply and demand functions (e.g., NZFARM). Such assumptions are useful when the region/catchment of interest is important/big enough to set market prices (as opposed to being a price taker).
- Platform: Both types of models share the same GAMS programming platform (non-open source), which provides access to a whole range of solvers at a specific price.
- Space: Such models currently treat space in a very simplistic manner by classifying it into a small set of classes, to make the models tractable and solvable.
- Uncertainty: is treated using a "what if" scenario approach rather than using probabilistic approaches. The models are driven under the main assumption that profits are maximised. Hence, they assume that all of the landowners in the block, catchment or region maximise for average profits disregarding uncertainty and other types of preferences.

- Dynamic: Although both models are static, Landcare is working on a prototype to make NZFARM dynamic. To include the dynamic component, such land-use models such consider the dynamic principles of forest management models (e.g., Woodstock).

Spatial Optimisation

- Optimisation of resource allocation across the landscape (e.g., land-use, fertiliser, pesticides, water, management practices (e.g., mitigation measures).
- Spatial optimisation (based on linear programming) to minimise impact on the environment, i.e., maximisation of ecosystem services.
- Soil erosion mitigation optimisation.
- NZ model: LUMASS.

Parameter Optimisation and constraint of input/model uncertainty.

- Monte Carlo methods interrogating impacts of *a priori* specified parametric and/or other uncertainties on predictive uncertainty are very common, in theory if the *a priori* ranges are perfectly specified and enough statistical samples can be generated this method is "perfect"; neither of those conditions generally hold. Markov Chain Monte Carlo (MCMC) or a limited number of other global search routines can "perfectly" update the a priori to true ranges if measures of "good fit"/objective functions constraining model realisations can be specified. This also generally does not hold, but MCMC does also more efficiently search the parametric space, with some coverage everywhere (given enough run time), but more complete/efficient coverage where predictive accuracy is higher.
- Related to the above point, "true" global exploration of uncertainty and parametric optimization are often very expensive and a range of heuristic methods are available to reduce computational burden; generally relying on appropriate specification of "measures of good fit".
- These may be deterministic or stochastic, both are common; the extent to which they search for local minima versus attempting to provide some cover over the full parametric space is generally a function of the computational burden involved.
- LUCI includes classic Monte Carlo, MCMC (heuristically updating posterior parameter distribution) and a range of measures of fit (e.g., Nash Sutcliff/least squares, entropy/distributional shape measures, and fairly standard implementations of with options to view parameter interactions with objective functions graphical representations to aid exploration of parametric uncertainty and interactions which should be transferable to a number of other models without the interoperability framework.

Technical aspects

- Economic models and spatial optimisation model applications utilise mathematical optimisation for solving revenue / ecosystem services maximisation problem.
- Open source libraries are available for implementation of optimisation applications, However, for spatial application a framework is required to translates a spatial problem into a form the library understands.

- LUMASS provides a very flexible implementation of a spatial optimisation framework:
    - Arbitrary resource allocation to fixed spatial units
        - Land-use
        - Management practices
        - Water
        - Fertilizer
        - Pesticides
        - Etc.
    - Multi-objective optimisation.
    - Allocation constraints.
    - Performance constraints (min. requirements/max. tolerable).
    - Arbitrary zoning (representation of agricultural. Enterprises).
    - Flexible objective (i.e., max/min any criterion).
    - Framework for uncertainty assessments (with regards to constraints as well as with regard to input data (e.g., nitrate leaching values)).
    - Seems an 'easy' option to include economic modelling (based on linear programming) into the framework.
    - Also, provides the above mentioned non-economic options for spatial optimisation out-of-the-box.

Appendix U Risk table from the project proposal

See table on following pages.

Risk Category	Descriptors	Consequence	Mitigation	Responsibility
Complexity and usability	Representing a range of attributes (hydrologic, production, economic) in an interlinked system may result in complex and complicated model assemblies. Difficult to use the model and visualise results.	Models not useful for meeting Challenge goals. Only a few expert users able to operate the system.	Clear definition of scope and model selection. Inclusion of simplified models in the model suite. Include case studies and evaluation. Engage with stakeholders throughout the project. Framework to include visualisation aspects.	Governance Group, with support of Technical Group
Ownership	Models and the framework may not be accessible, usable and re-distributable.	Models and the framework are unavailable for use or are too expensive.	Require framework software is compliant with the Open Source Definition[1]. Promote open availability and open source of model components. Availability to be considered in selection of initial model set. Establish and maintain model metadata and repositories.	Governance Group
Model and framework continuity	Modelling framework and the implemented models are not maintained, no capability and resources made available for this.	Models stop being used. Loss of support and trust by model users.	Strong governance structure established. Funds sought from key model users. Key technical developers and managers of the code are identified. Use international framework and standards.	Governance Group
Documentation	Models and the framework are not documented, including underlying assumptions and peer review.	Lack of confidence and trust by user community.	Require contributed models and the framework to be documented.	Technical Group

Risk Category	Descriptors	Consequence	Mitigation	Responsibility
Data	Data is not available, is unsuitable, or is not delivered according to standards.	Complex data required means model only able to be used in well-resourced catchments.	Conceptual model design recognises data availability as an important consideration. Promotion of standards-driven data. Support from other Challenge and external data initiatives. Promotion of web access standards.	Governance Group, with support of Technical Group
Uncertainty	In the modelling framework, how is it analysed and are the acceptable error bands for calibration?	Poor use of data, inability to distinguish accuracy and precision.	Framework includes reported uncertainty and objective function tools	Technical Group
Time constraints	Models are not linked into the framework and to data within the available time.	Models not available to be used or integrated.	Select an initial core set of models for implementation done in first stage, with mid-implementation review, and sufficient resourcing. Use standards where appropriate and international framework software.	Governance Group
Recognised use	Framework has bespoke or 'in-house' models with little institutional support	Models not used or trusted.	Use peer-reviewed or tested models in the core model set, while enabling innovation through use of a flexible framework.	Governance Group
Cost	Insufficient financial resources for adapting models and data for the framework, trialling, and maintaining the framework and providing training and dissemination. Costs for applying the	Model is not delivered satisfactorily and is considered too expensive for use.	Ensure significant ongoing funding available. Design and evaluation steps to consider ease of use and cost of the system. Governance Group and the Challenge seek external leveraging funding.	Challenge Governance Group

Risk Category	Descriptors	Consequence	Mitigation	Responsibility
	framework in a given situation become high.			
Scalability and extensity	Framework too inflexible to enable variable model scales and resolutions, or adaptation as new component and model structures become available.	Framework has limited use, and loses support.	Use flexible framework software and data standards. Preferably have scaling tools available (e.g., tools to modify grid resolution.)	Technical Group

[1] https://opensource.org/osd-annotated.