

A Software Component for Model Output Evaluation

G. Bellocchi^{a,*}, M. Donatelli^{a,**,†}, M. Acutis^b, E. Habyarimana^a

^a Agriculture Research Council, 40128 Bologna, Italy;

* currently at European Commission Joint Research Centre, Institute for Health and Consumer Protection, Biotechnology and GMOs Unit, Ispra, Italy;

** currently seconded to European Commission Joint Research Centre, Institute for the Protection and Security of the Citizen, Agriculture Unit, Agri4cast Action, Ispra, Italy

^b Department of Crop Science, University of Milan, Italy

[†] Corresponding author, marcello.donatelli@irc.it

Abstract: As the role of biophysical models in ecological, biological and agronomic areas grows in importance, there is an associated increase in the need for suitable approaches to evaluate the adequacy of model outputs (model testing, often referred as to “validation”). Effective testing techniques are required to assess complex models under a variety of conditions, including a wide range of validation measures, possibly integrated into composite metrics. Both simple and composite metrics are being proposed by the scientific community, continuously broadening the pool of options for model evaluation. However, such new metrics are not available in commonly used statistical packages. At the same time, the large amount of data generally involved in model testing makes the operational use of new metrics a labour-consuming process, even more when composite metrics are meant to be used. An extensible and easily reusable library encapsulating such metrics would be an operational way to share the knowledge developed on model testing. The emergence of the component-oriented programming in model-based simulation has fostered debate on the reuse of models. There is a substantial consensus that component-based development is indeed an effective and affordable way of creating model applications, if components meet via their architecture a set of requirements which make them scalable, transparent, robust, easily reusable, and extensible. This paper illustrates the Windows .NET2 component IRENE (Integrated Resources for Evaluating Numerical Estimates) and a first prototype application using it, SOE (Simulation Output Evaluator), to present a concrete application matching the above requirements in the area of model testing.

Keywords: IRENE; Modelling; Model testing; SOE; Software component.

1. INTRODUCTION

The evaluation of model adequacy is an essential step of the modelling process [Jakeman et al., 2006] because it indicates the level of accuracy (how closely model-estimated values are to the actual values) of the model estimations. This is an important phase either to build confidence in a model or to allow selection of alternative models. The concept of validation is quite generally interpreted in terms of model suitability for a particular purpose, that means a model is valid and sound if it accomplishes what is expected from it [e.g. Sargent, 2001]. One of the principles of validating models dictates that complete testing is not possible [Balci, 1997], thus proving that a model is absolutely valid is an issue without solution. Exhaustive validation requires testing all possible model outputs under virtually all possible inputs (i.e., conditions). Combinations of feasible values of model inputs/outputs can generate a very large number of logical paths during execution. Due to

time and budgetary constraints, exhaustively testing the accuracy of so many logical paths is often unfeasible. Consequently, in model evaluation, the purpose is to increase confidence that the accuracy of the model meets the standards required for a particular application rather than establish that the model is absolutely correct in all circumstances. This suggests that the challenge for model evaluation is to ensure that, in addition to ensuring that minimal (application specific) standards are met, the testing should also increase the credibility of the model with the users while remaining cost effective. As a general rule, the more tests that are performed in which it cannot be proven that the model is incorrect, the more confidence in the model is increased. Yet the low priority typically given to validation in model project proposals and development plans indicates a tendency towards the minimum standards approach alone being adopted. Several validation methods are available, as reviewed in the international literature [e.g. Bellocchi, 2004; Tedeschi, 2006] but, typically, only a limited number of methods are used in modelling projects, often due to time and resource constraints. In general, limited testing may hinder modeller's ability to substantiate sufficient model accuracy.

The availability of appropriate software tools is necessary to assist model validation. The freeware, Microsoft COM-based tool IRENE_DLL (Integrated Resources for Evaluating Numerical Estimates_Dynamic Link Library, Fila et al., 2003), is a flexible tool providing extensive, integrated, statistical capabilities for use in model validation. IRENE_DLL-based spreadsheet applications were used into modelling projects to perform statistical evaluation of model outputs [Bechini et al., 2006; Incrocci et al., 2006; Diodato and Bellocchi, 2007a, b, c; Amaducci et al., 2008]. The DLL was also used to tailor a dedicated application for evaluation of pedotransfer functions [Fila et al., 2006], and coupled with the model for rice production WARM [Bellocchi et al., 2006]. Since IRENE_DLL was developed, the component-oriented paradigm has evolved, specifying new requirements in order to increase software quality, reusability, extensibility, and transparency for components providing solutions in the biophysical domain [Donatelli and Rizzoli, 2008]. Also, the COM paradigm has become *de facto* obsolete since the advent of the .NET platform of Windows which provides, among other features, much simpler provisions for installation and versioning.

In this paper, the role of computer-aided validation support is considered for its effectiveness to support modelling projects which are applicable across a broad range of fields. In particular, the features of the Windows .NET component IRENE (Integrated Resources for Evaluating Numerical Estimates) and one prototype application SOE (Simulation Output Evaluator) using it are presented, showing the advantages achievable via a component-oriented architecture in building a software component in the domain of model validation statistics.

2. STATISTICS FOR MODEL VALIDATION

The component implements a range of statistical measures and visual techniques that can be used to assess goodness-of-fit of a given model (when its outputs are assessed against actual data) and to compare the performance of a suite of models (Table 1). There are several papers that contain the equations of the statistics and methods used for model validation, hence the equations are not reported here (see online documentation).

Table 1. Model validation statistics implemented in IRENE.

Group of statistics	Type of statistics
Difference-based	Simple, absolute, squared indices; test statistics
Association-based	Regression parameters (from alternative fitting methods); test
Pattern	Pattern indices (range-based, F-based)
Aggregate	Multi-level composite indicators
Time mismatch	Time mismatch indices

2.1 Difference-based Statistics

Difference-based (or residual-based) statistics quantify the departure of the model outputs from the measurements, consisting of simple, absolute and squared statistics [Fox, 1981]. Mean bias, the mean difference between observed and model-estimated values, is likely to be the oldest statistic to assess model accuracy. More common is the mean square error, or equivalently its square root, the root mean square error (or derived statistics such as the relative root mean square error). Mean absolute error measures the mean absolute difference between observed and estimated values, and is also used as the mean absolute percent error. Willmott [1981] index of agreement and the modelling efficiency statistic [Nash and Sutcliffe, 1970], interpreted as the proportion of variation explained by the model, are to be used for a better interpretation of squared errors.

2.2 Association-based Statistics

Measures of statistical association such as correlation and regression coefficients [Addiscott and Whitmore, 1987] provide quantitative estimates of the statistical covariation between estimated and measured values. Linear regression parameters (intercept and slope) are calculated according to two methods: ordinary least squares and reduced major axis.

2.3 Pattern Statistics

The presence of patterns in the residuals versus independent variables (e.g., a model input or a variable not considered in the model) is quantified by computing pattern indices of two types: range-based and F-based [Donatelli et al., 2004a].

2.4 Probability Distributions

To identify model adequacy for stochastic models, both probability density and cumulative distribution functions from observed and estimated values are compared [Reynolds and Deaton, 1982].

2.5 Aggregation of Statistics

Multiple validation metrics can be combined according to the methodology originally developed by Bellocchi et al. [2002], based on an expert weighting expression (fuzzy-based rules) of the balance of importance of the individual metrics and their aggregation into modules (first-stage aggregation). The metrics which can be aggregated are the ones either made available within the component, and/or resulting from an extension of it. The modules (or modules and simple metrics) may be further aggregated into an overall indicator (second-stage aggregation). Several aggregation levels can be composed hierarchically. Figure 1 shows the tree-view of a two-level aggregation index.

2.6 Time Mismatch

Either simple or composite statistics can be used to assess the mismatch in time series comparison [Donatelli et al., 2002].

3. SOFTWARE COMPONENT DESIGN

The software component IRENE (Integrated Resources for Evaluating Numerical Estimates) contains functions for the computation of simple and composite statistics for evaluation of model estimates. Each simple statistic is implemented as a class. Composite statistics are the result of an association of classes, obtained dynamically via an XML file. Each statistic computed via internal methods is fully described as description and units, as well as maximum, minimum and default values. The software architecture of this component further develops the one used in previously developed components [Carlini et

al., 2006; Donatelli et al., 2006a, b], and is fully described by Donatelli and Rizzoli [2008]. Transparency and ease of maintenance are granted, also providing functionalities such as the test of input data versus their definition prior to computing any simple or composite validation metric; same tests can be run on outputs. The component can be extended independently by third parties without requiring re-compilation. The component is freely available to scientists and institutions developing component-oriented models and applications in the agro-ecological field. The component is written in C# for .NET2, but extensions can be written in any .NET language. The component is deployable and reusable in any application developed using the Microsoft .NET framework. The IRENE software development kit includes sample projects which show how to use and extend the component. Code documentation is also provided and the online help file is available at: <http://www.apesimulator.it/help/utilities/irene>.

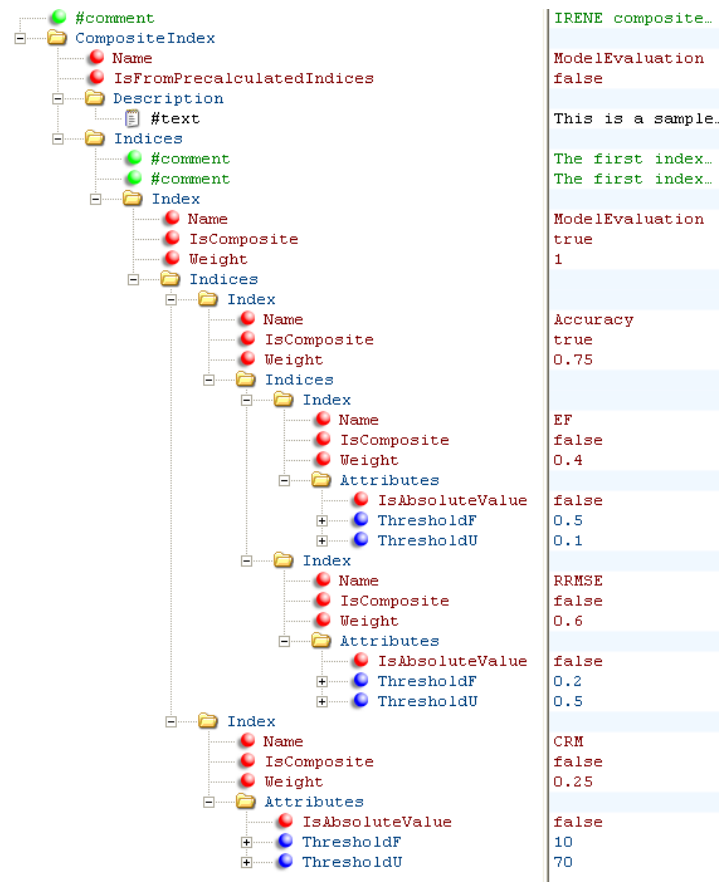


Figure 1. Tree view of a two-level composite indicator (see 3.1).

3.1 Composite Indicators

Composite indicators can be composed in hierarchical structure (including any hierarchical level), using statistics computed internally to IRENE, made available from extensions of IRENE, or provided as inputs. In the latter case, the component runs only as a fuzzy-based aggregator for construction of aggregated indicators. The structure of composite indicators can be saved as XML file for reuse. As simple example (XML code in Figure 2), RRMSE (relative root mean square error) and EF (modelling efficiency) are aggregated into an indicator (named “Accuracy”), and the latter is further combined to CRM (coefficient of residual mass) to give a second-level indicator (namely, “ModelEvaluation”).

```

<!--IRENE composite indices CRA.IRENE.dll -->
<CompositeIndex Name="ModelEvaluation" IsFromPrecalculatedIndices="false">
  <Description>This is a sample two-levels composite
  index</Description>
  <Indices>
    <!--The first index node has always weight = 1 -->
    <!--The first index node always IsComposite = true -->
    <Index Name="ModelEvaluation" IsComposite="true" Weight="1">
      <Indices>
        <Index Name="Accuracy" IsComposite="true" Weight="0.75">
          <Indices>
            <Index Name="EF" IsComposite="false" Weight="0.4">
              <Attributes IsAbsoluteValue="false">
                <ThresholdF>0.5</ThresholdF>
                <ThresholdU>0.1</ThresholdU>
              </Attributes>
            </Index>
            <Index Name="RRMSE" IsComposite="false"
              Weight="0.6">
              <Attributes IsAbsoluteValue="false">
                <ThresholdF>0.2</ThresholdF>
                <ThresholdU>0.5</ThresholdU>
              </Attributes>
            </Index>
          </Indices>
        </Index>
        <Index Name="CRM" IsComposite="false" Weight="0.25">
          <Attributes IsAbsoluteValue="false">
            <ThresholdF>10</ThresholdF>
            <ThresholdU>70</ThresholdU>
          </Attributes>
        </Index>
      </Indices>
    </Index>
  </Indices>
</CompositeIndex>

```

Figure 2. XML code of a two-level composite index (see tree view in Figure 1).

4. SIMULATION OUTPUT EVALUATOR

Simulation Output Evaluator (SOE) is a data analysis tool which makes use of the component IRENE, designed to provide easy access to model evaluation techniques and graphical views. It is illustrated here with the purpose of providing one proof-of-concept of a client-side application of IRENE. For each analysis, two datasets (which can be observed and model-estimated data points) must be loaded (in XML-format with schema, MS Excel worksheets, or any more compact binary format, the latter via a reusable component also made available, CRA.Core.IO.dll). When the second dataset is loaded, combo boxes of tables and variables are populated with the data from the supporting format that the user can select for analysis. Another variable, to be used as covariate in the pattern analysis, can also be loaded. A number of analyses can be performed and displayed in various screen tabs, providing graphical data views as well as the indices computed via IRENE. One of the tabs contains a summary of the analyses done which can be saved as XML file; the graphics can also be saved as .JPG-format files. A tree-structure graphic reflecting the hierarchical composition of statistics into aggregated indicators and their impact is also available. The current version of SOE is a first prototype, currently being refactored. Its help file is available online at <http://www.apesimulator.it/help/tools/soe>.

5. DISCUSSION

Software tools specifically created for model validation provide effective computer-aided support. In particular, they can significantly reduce the testing time and effort. A key step in this direction is the coupling between model components and validation techniques, the latter also implemented into component-based software [e.g. Bellocchi and Confalonieri, 2006]. To meet the substantial model quality challenges, it is necessary to improve the tools and technologies currently available, and their cost-benefit characterizations. The

emergence of new technologies in simulation modelling has, in fact, fostered debate on the reuse of models. A large number of existing agricultural and ecological models have been implemented as software that cannot be well maintained or reused, except by their authors, and therefore cannot be easily transported to applications developed by third parties. Therefore the focus needs to be on the use of design patterns that encourage usability, reusability and cross-language compatibility, thus facilitating model development, integration, documentation and maintenance [Donatelli et al., 2004b]. The component-oriented paradigm is the leading methodology in developing systems in a variety of domains, including agro-ecological modelling [e.g. Argent, 2005]. Component-oriented development has emerged steadily as a paradigm that focuses on explicit and semantically rich interfaces. The substantial consensus that emerged within the scientific community around component-based development as effective and affordable way of creating model applications relies upon the possibility offered by the component architecture to meet a set of requirements to make components scalable, transparent, robust, easily reusable, and extensible. The distribution of validated model components can substantially decrease model validation effort when reused, providing functionalities to improve the process of model verification. The component IRENE makes available a number of metrics in a discrete software unit, hence allowing operational access to the computation of a variety of model evaluation metrics. Its extensibility, also available independently for third parties, allows easy maintenance and further development.

6. CONCLUSIONS

The goals of IRENE development are to extend access to model validation statistics to multiple users, and provide architecture to ensure reuse and extension of coded statistics. IRENE attempts to overcome some of the technical challenges that to date have limited the development of reusable validation capabilities. Via its documentation and the metadata associated with each variable, IRENE is also a way to share data processing knowledge in an extensible way.

ACKNOWLEDGEMENTS

This publication has been funded by the SEAMLESS integrated project, EU 6th Framework Programme for Research, Technological Development and Demonstration, Priority 1.1.6.3. Global Change and Ecosystems (European Commission, DG Research, contract no. 010036-2).

The authors gratefully acknowledge Ing. A. Sorce for the design and implementation of the composite indices modules within the IRENE component.

REFERENCES

- Addiscott, T.M., and A.P. Whitmore, Computer simulation of changes in soil mineral nitrogen and crop nitrogen during autumn, winter and spring, *Journal of Agricultural Science (Cambridge)*, 109, 141-157, 1987.
- Amaducci, S., M. Colauzzi, G. Bellocchi, and G. Venturi, Modelling post-emergent hemp phenology (*Cannabis sativa* L.): Theory and evaluation, *European Journal of Agronomy*, 28, 90-102, 2008.
- Argent, R.M., A case study of environmental modelling and simulation using transplantable components. *Environmental Modelling & Software*, 20, 1514-1523, 2005.
- Balci, O., Principles of simulation model validation, verification, and testing, *Transactions of S.C.S.I.*, 14, 3-12, 1997.
- Bechini, L., S. Bocchi, T. Maggiore, and R. Confalonieri, Parameterization of a crop growth and development simulation model at sub-model components level. An example for winter wheat (*Triticum aestivum* L.), *Environmental Modelling & Software*, 21, 1042-1054, 2006.
- Bellocchi, G., Appendix A: Numerical indices and test statistics for model evaluation. In: Pachepsky, Y., and W. Rawls (Eds.), *Development of pedotransfer functions in soil hydrology*, Elsevier, Amsterdam, 394-400, 2004.

- Bellocchi, G., M. Acutis, G. Fila, and M. Donatelli, An indicator of solar radiation model performance based on a fuzzy expert system, *Agronomy Journal*, 94, 1222-1233, 2002.
- Bellocchi, G., R. Confalonieri, and M. Donatelli, Crop modelling and validation: integration of IRENE_DLL in the WARM environment, *Italian Journal of Agrometeorology*, 11, 35-39, 2006.
- Carlini, L., G. Bellocchi, and M. Donatelli, Rain, a software component to generate synthetic precipitation data, *Agronomy Journal*, 98, 1312-1317, 2006.
- Diodato, N., and G. Bellocchi, Estimating monthly (R)USLE climate input in a Mediterranean region using limited data, *Journal of Hydrology*, 345, 224-236, 2007a.
- Diodato, N., and G. Bellocchi, Modelling reference evapotranspiration over complex terrains from minimum climatological data, *Water Resources Research*, 43, doi:10.1029/2006WR005405, 2007b.
- Diodato, N., and G. Bellocchi, Modelling solar radiation over complex terrains using monthly climatological data, *Agricultural and Forest Meteorology*, 144, 111-126, 2007c.
- Donatelli, M., M. Acutis, G. Bellocchi, and G. Fila, New indices to quantify patterns of residuals produced by model estimates, *Agronomy Journal*, 96, 631-645, 2004a.
- Donatelli, M., M. Acutis, G. Fila, and G. Bellocchi, A method to quantify time mismatch of model estimates. In: Villalobos, F.J., and L. Testi (Eds.), *Proceedings of the 7th European Society for Agronomy Congress*, July 15-18, Cordoba, Spain, 269-270, 2002.
- Donatelli, M., G. Bellocchi, and L. Carlini, Sharing knowledge via software components: models on reference evapotranspiration, *European Journal of Agronomy*, 24, 186-192, 2006a.
- Donatelli, M., L. Carlini, and G. Bellocchi, A software component for estimating solar radiation, *Environmental Modelling & Software*, 21, 411-416, 2006b.
- Donatelli, M., A. Omicini, G. Fila, and C. Monti, Targeting reusability and replaceability of simulation models for agricultural systems. In: Jacobsen, S.E., C.R. Jensen, and J.R. Porter (Eds.), *Proceedings of the 8th European Society for Agronomy Congress*, July 11-15, Copenhagen, Denmark, 237-238, 2004b.
- Donatelli, M., and A. Rizzoli, A design for framework-independent model components of biophysical systems. *These proceedings*, 2008.
- Fila, G., G. Bellocchi, M. Donatelli, and M. Acutis, IRENE_DLL: A class library for evaluating numerical estimates, *Agronomy Journal*, 95, 1330-1333, 2003.
- Fila, G., M. Donatelli, and G. Bellocchi, PTFIndicator: an IRENE_DLL-based application to evaluate estimates from pedotransfer functions by integrated indices, *Environmental Modelling & Software*, 21, 107-110, 2006.
- Fox, D.G., Judging air quality model performance: a summary of the AMS workshop on dispersion models performance, *Bulletin of the American Meteorological Society*, 62, 599-609, 1981.
- Incrocci, L., G. Fila, G. Bellocchi, A. Pardossi, C.A. Campiotti, and R. Balducchi, Soil-less indoor-grown lettuce (*Lactuca sativa* L.): approaching the modelling task, *Environmental Modelling & Software*, 21, 121-126, 2006.
- Jakeman, A.J., R.A. Letcher, and J.P. Norton, Ten iterative steps in development and evaluation of environmental model, *Environmental Modelling & Software*, 21, 602-614, 2006.
- Nash J.E., and J.V. Sutcliffe, River flow forecasting through conceptual models, Part I - A discussion of principles, *Journal of Hydrology*, 10, 282-290, 1970.
- Reynolds, J.M.R., and M.L. Deaton, Comparisons of some tests for validation of stochastic simulation models, *Communications of Statistical Simulation and Computation*, 11, 769-799, 1982.
- Sargent, R.G., Verification, validation and accreditation of simulation models. In: Peters, B.A., J.S. Smith, D.J. Medeiros, and W.W. Rohrer (Eds.), *Proceedings of the 2001 Winter Simulation Conference*, Arlington, VA, USA, December 10-13, 106-114, 2001.
- Tedeschi, L.O., Assessment of the adequacy of mathematical models, *Agricultural Systems*, 89, 225-247, 2006.
- Willmott, C.J., On the validation of models, *Physical Geography*, 2, 184-194, 1981.