

Progetto SIPEAA
Strumenti Informatici per la Pianificazione
Eco-compatibile delle Aziende Agrarie

MIPAF - Ministero delle Politiche Agricole e Forestali



Agriculture Research Council

Course:

**“Development of biophysical
modular models in .NET”**

Bologna, October 17-21, 2005



*Co-sponsored by the Italian
Society of Agronomy*

Istituto Sperimentale Colture Industriali
Via di Corticella, 133
40128 Bologna

<http://www.sipeaa.it/corso>
email: agronomy@isci.it

Course objectives

In the domain of agro-ecological modelling, many efforts have been devoted in the last decade to define software designs which improve the efficiency of the process of model development by taking advantage of new technologies.

The advent of component-based programming has enabled the development of scalable, robust, large-scale applications in a variety of domains, including agro-ecological modelling. The concept of developing modular systems for biophysical simulation has led to the development of several modelling frameworks (e.g. Simile, ModCom, IMA, TIME, OpenMI, SME, OMS), which make use of models as software components that can be linked together and composed according to the modeling purpose. However, little attention has been placed to the development of components which intrinsically promote re-usability, interchangeability, and extensibility of models. Components with such characteristics could find a different use in different clients and contexts, from end-user applications to web services through modelling frameworks; this would be interesting for cross-cutting communities of model developers.

While waiting for the availability of tools and open systems which will make the underlying software structure irrelevant to model developers, the need to deliver model applications ready for specific use still requires to take into account several non-trivial software-related aspects. Reusability chiefly depends on the capability of a component to fully encapsulate the solution to a common problem in terms of its underlying model: technology adopted and component architecture are then parts of the more general encapsulation problem. Rather than a mere issue of technology or component architecture, this one is a software engineering issue that commands a general approach meant to reduce component dependencies to the

minimum. As a result, a simulation system should adhere to the following principles:

- Explicit ontology. Both individual components and the whole system should be designed upon an explicitly-formulated ontology which should cover every aspect shared among components.
- Semantic-based components. Individual components are not constructed as collections of procedures or functions exhibiting some commonalities, but are rather designed to encapsulate all the knowledge and functionalities related to a specific field or sub-field.

The goal of this course is then twofold: 1) to provide to participants an overview of aspects related to the development of model components, 2) to show how to use components to build a simulation system using the modelling framework NModCom.

Target participants

The course is targeted to doctorate students, junior and senior researcher with a basic programming experience. A working experience with an OOP language will be beneficial.

Software and technology

The programming language used will be C#. The wide overlap between the C# and Java object models make this course interesting also for those who aim at developing applications in Java. Participants must come with a laptop with a network card, Windows XP, Visual studio 2003 (the academic version is ok), NUnit (<http://www.nunit.org>), NDoc (<http://ndoc.sourceforge.net>), and Tortoise (<http://www.tortoise cvs.org>) installed (the latter three are free software)

Programme

Monday 17

11:00 -12:30

Registration and setting of participants' laptops

12:30 – 14:00

Lunch

14:00 – 15:00 (A. Rizzoli)

Object oriented programming: key concepts

15:15 – 15:45 (M. Donatelli)

Visual Studio 2003: overview

15:45 – 16:00

Coffee break

16:00 – 16:45, 17:00 – 17:45 (M. Donatelli)

Key concepts in component- oriented programming; development of a sample component

Tuesday 18

8:45 -9:30, 9:45 - 10:30 (M. Donatelli)

The design pattern strategy

10:30 - 10:45

Coffee break

10:45 – 11:30, 11:45 - 12:30 (M. Donatelli)

Unit tests: definition and application

12:30 – 14:00

Lunch

14:00 – 14:45, 15:00 - 15:45 (M. Donatelli)

The design-by-contract approach: definition and application

15:45 – 16:00

Coffee break

16:00 – 16:45, 17:00 - 17:45 (M. Donatelli)

Development of a client for the component

Wednesday 19

8:45 – 9:30, 9:45 - 10:30 (M. Donatelli)

A component/sub-component architecture

10:30 - 10:45

Coffee break

10:45 – 11:30, 11:45 - 12:30 (M. Donatelli)

A dynamic component: "CLIMA"

12:30 – 14:00

Lunch

14:00 – 15:45 (M. Acutis)

A dynamic component, part 1: "SOILWCN"

15:45 – 16:00

Coffee break

16:00 – 17:45 (M. Acutis)

A dynamic component, part 2: "SOILWCN"

Thursday 20

8:45 – 9:30, 9:45 – 10:30 (F. Van Evert)

The modelling framework ModCom

10:30 - 10:45

Coffee break

10:45 – 11:30, 11:45 - 12:30 (F. Van Evert)

Developing a ModCom application, part 1

12:30 – 14:00

Lunch

14:00 – 14:45, 15:00 - 15:45 (F. Van Evert)

Developing a ModCom application, part 2

15:45 – 16:00

Coffee break

16:00 – 16:45, 17:00 - 17:45 (F. Van Evert)

Developing a ModCom application, part 3

Friday 21

8:45 -10:30

To be defined, based on previous days outcome

10:30 - 10:45

Coffee break

10:45 -12:30

To be defined, based on previous days outcome

Lecturers

Prof. Marco Acutis, DiProVe, UNIMI, Milano

Dr. Marcello Donatelli, CRA-ISCI, Bologna

Dr. Andrea Rizzoli, IDSIA, Manno-Lugano,

Dr. Frits Van Evert, PRI, Wageningen

Secretariat

Mrs. Claudia Maestrini

Tel +39 051 6316846, Fax +39 051 7456931

email: c.maestrini@isci.it