modextreme

agriculture facing extreme climatic events

European Project n° 613817

Workshop – November 3rd 2015

The BioMA platform and applications

Marcello Donatelli (CREA)

crea

November 3-4, 2015

Outline

- Do we need a modelling framework?
- What is BioMA?
- BioMA applications

Model development and reuse

- The demand of model tools to perform integrated evaluation of agro-ecological systems has further increased in the last decade.
- The major obstacle to develop such simulation systems has been the fragmented availability of modelling resources, partly due to technical bottlenecks.
- Extension of modelling resources by adding modules, and replacing or changing existing ones to accommodate new modules, has not been at reach except by full recoding.

The generic modelling problem

$$f(x) = a_0 + \sum_{n=1}^{\infty}\left(a_n \cos\frac{n\pi x}{L} + b_n \sin\frac{n\pi x}{L}\right)$$

Rainfall

Management

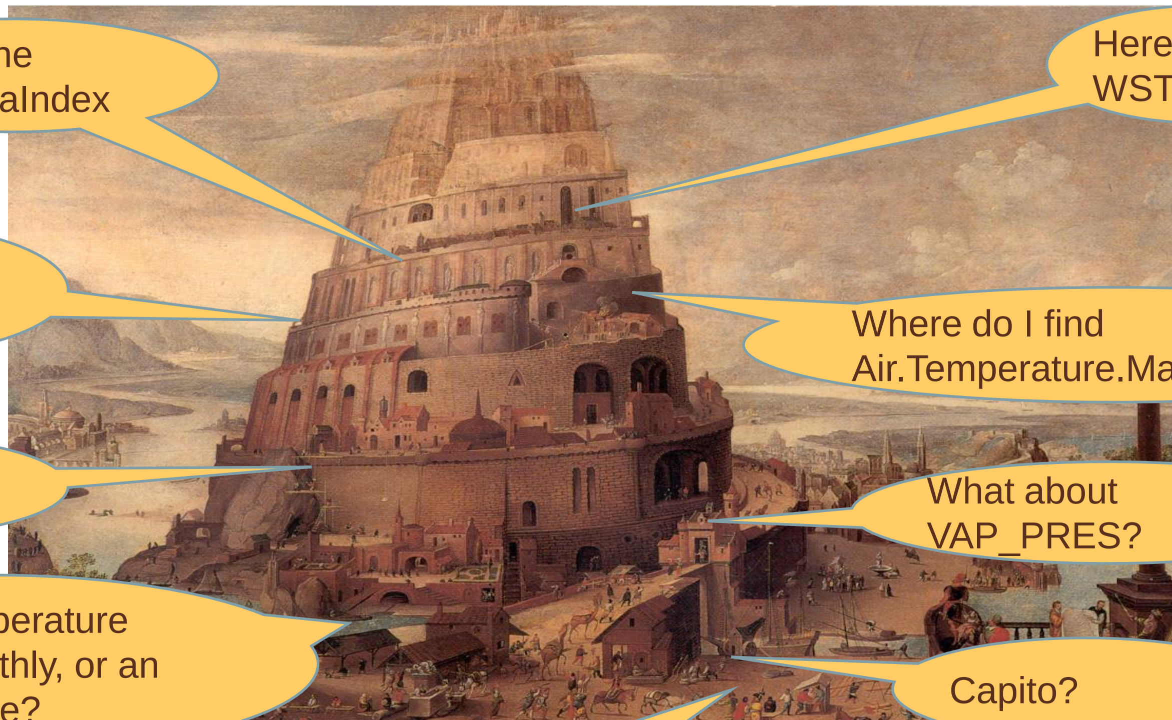
irrigation
tillage
fertilization
pesticides
herbicides
harvest/planting

Percolation

Leaching

- water infiltration
- water redistribution
- runoff
- evaporation
- percolation
- solutes transport
- salinization
- nitrogen fixation
- residues fate
- O.M. mineralization
- nitrogen transformations
- water erosion
- ammonia volatilization
- ammonium sorption

Agronomists, Soil scientists, Geographers, Meteorologists, …



Slide courtesy of I.Athanasiadis

Model frameworks

- Since many years model frameworks have represented a substantial step forward with respect to monolithic implementations of biophysical models.
- The separation of algorithms from data, the reusability of services such as I/O procedures, have brought a solid advantage in the development of simulation systems.
- However, the diffusion of such frameworks beyond the groups developing them, as model development environment, can be considered modest.
- The reusability of models has also proved to be modest; a model unit for a given framework is not used in other frameworks.

Model frameworks (cont.)

- Current implementations of modeling solution vary from monolithic Fortran code to C++, to fully object oriented memory managed languages (C#, Java).
- Even if the technology available would allow much more functionalities of what is being exploited in most cases, software framework have been constantly improved in terms of quality; still, the problem of model reuse has been largely unsolved.
- The focus on frameworks has made software architects at least partially overlooking on the requirements of reusability per se of model units.

Reuse of what?

- Model reuse is an overloaded term. Within biophysical modelling, a "model" can be:
  - A process based representation of either a single process or of a compartment;
  - A whole modelling solution allowing the simulation of the system of interest.
- The point in reuse however is not the granularity of the model unit; the key difference is if the discrete unit to be reused is used in workflows, or is composed to iterate over the execution of time-steps with other units.

Asynchronous or synchronous?

- If the reuse is of modelling solutions, adapters are built following the requirements of a specific software framework to enable links in workflows, or for ensemble runs.
  - The “model” units run asynchronously
  - Incompatibilities of binaries can be overcome
- In the case of iteration across model units over time the requirements for an effective reuse increase:
  - Key features in the architecture of model units are required
  - Binaries incompatibility across models units becomes an insurmountable obstacle in practical terms
- Model development requires being able to modify and link model units working synchronously at run time.

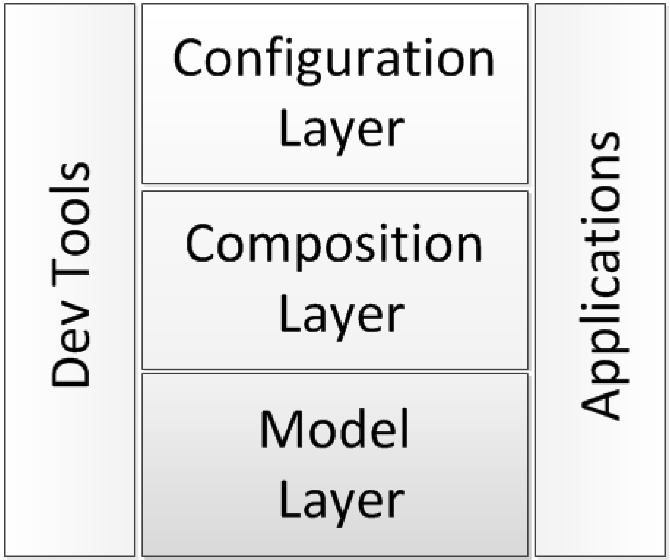
New requirements

- Also, some new high level requirements emerged for modelling frameworks:
  - To increase the transparency of the modelling solutions being built compared to legacy code available, for each of the modelling solutions being built;
  - To increase the traceability of performance of each modelling unit used in modelling solutions;
  - To involve teams without requiring them to commit to a whole infrastructure they would not own.
- To maximize both reusability and accessibility, we chose to develop a simulation system based on framework-independent components, both for model and for tool components.

The high level architecture of the software modelling environment

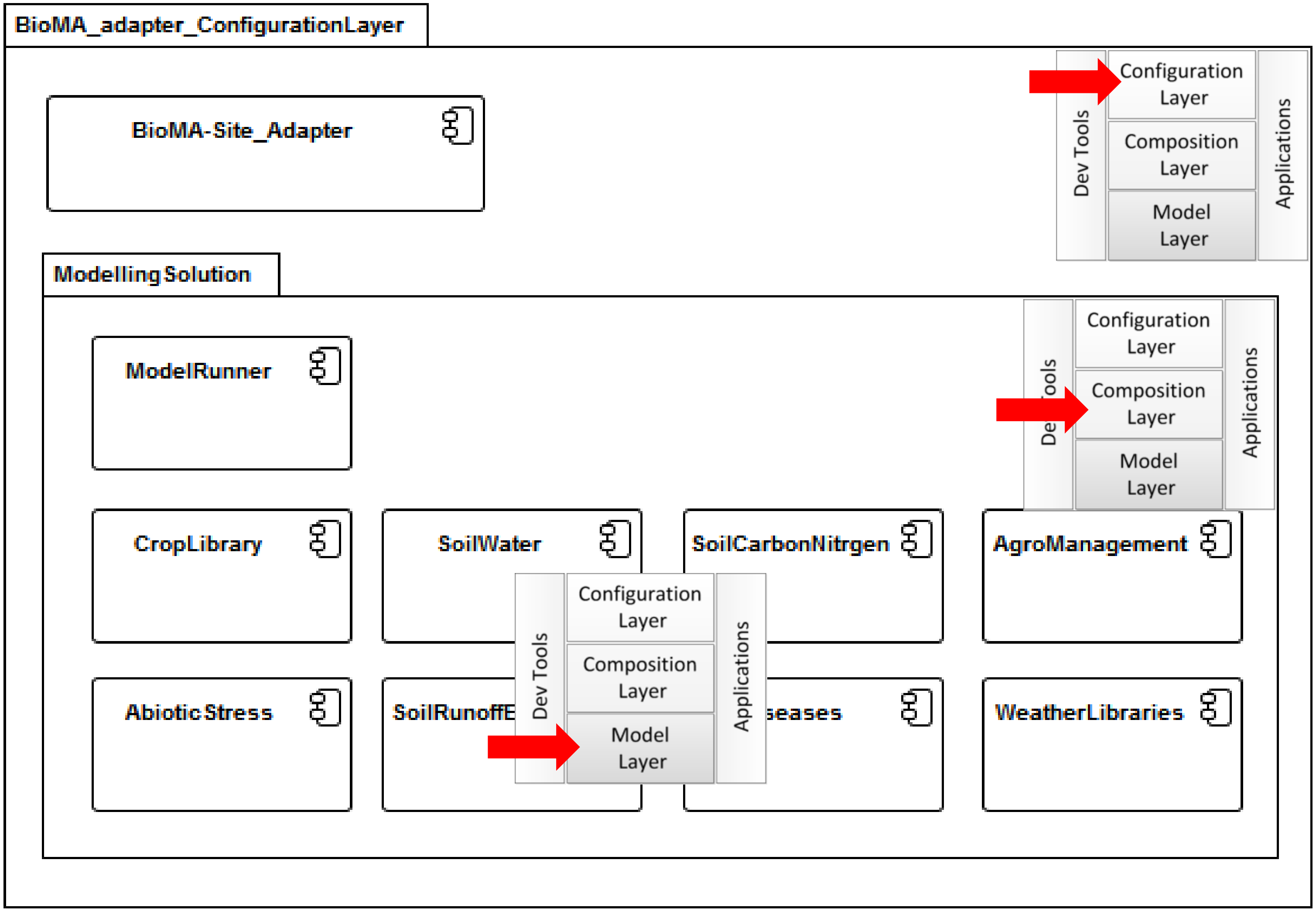
FROM MODELS TO VIEWERS

From models to viewers



- **Model Layer**: fine grained/composite **models** implemented in components
- **Composition Layer**: **modeling solutions** from model components
- **Configuration Layer**: **adapters** for advanced functionalities in controllers
- **Applications:** from console to advanced MVC implementations
- **DevTools**: code generators, UI components and applications

From models to applications



BioMA Model Libraries

BioMA

Weather

Weather variables (*AirTemperature, Evapotranspiration, LeafWetness, Precipitation, SolarRadiation, Wind*)
Weather generators (*ClimGen*)

Abiotic stress

Heat damage, cold shocks, lodging, water stress

Biotic stress

Generic air-borne diseases simulator (*Diseases, Magarey*)
Soil-borne diseases (*SBD*)
Corn borer (*MYMICS*)

Quality

Agricultural products (*AgroProQ*)

Crop / Plant

Generic crop simulators (*Wofost, CropSyst3*; in progress *AcquaCrop*, new *CropSyst*)
Rice (*WARM*)
Wheat (in progress *SiriusQ*)
Tree species: (*Hazelnut;* in progress *Grapes, Poplar*)
Sugarcane (*Canegro*)
Giant Reed (*Arungro*)
Generic pasture (*STICS-Pasture*)

Agro-chemicals

Agro-chemicals dynamics (*AgroChemicals*)

Soil

Soil water erosion runoff (*CN, Eurosem*)
Soil water (*cascading, cascading travel time, Richards*)
Soil surface and profile temperature
Soil carbon and nitrogen
Soil Pedotransfer functions (*SoilPAR*)

Agro-management

Rule-based models (*AgroManagement*)

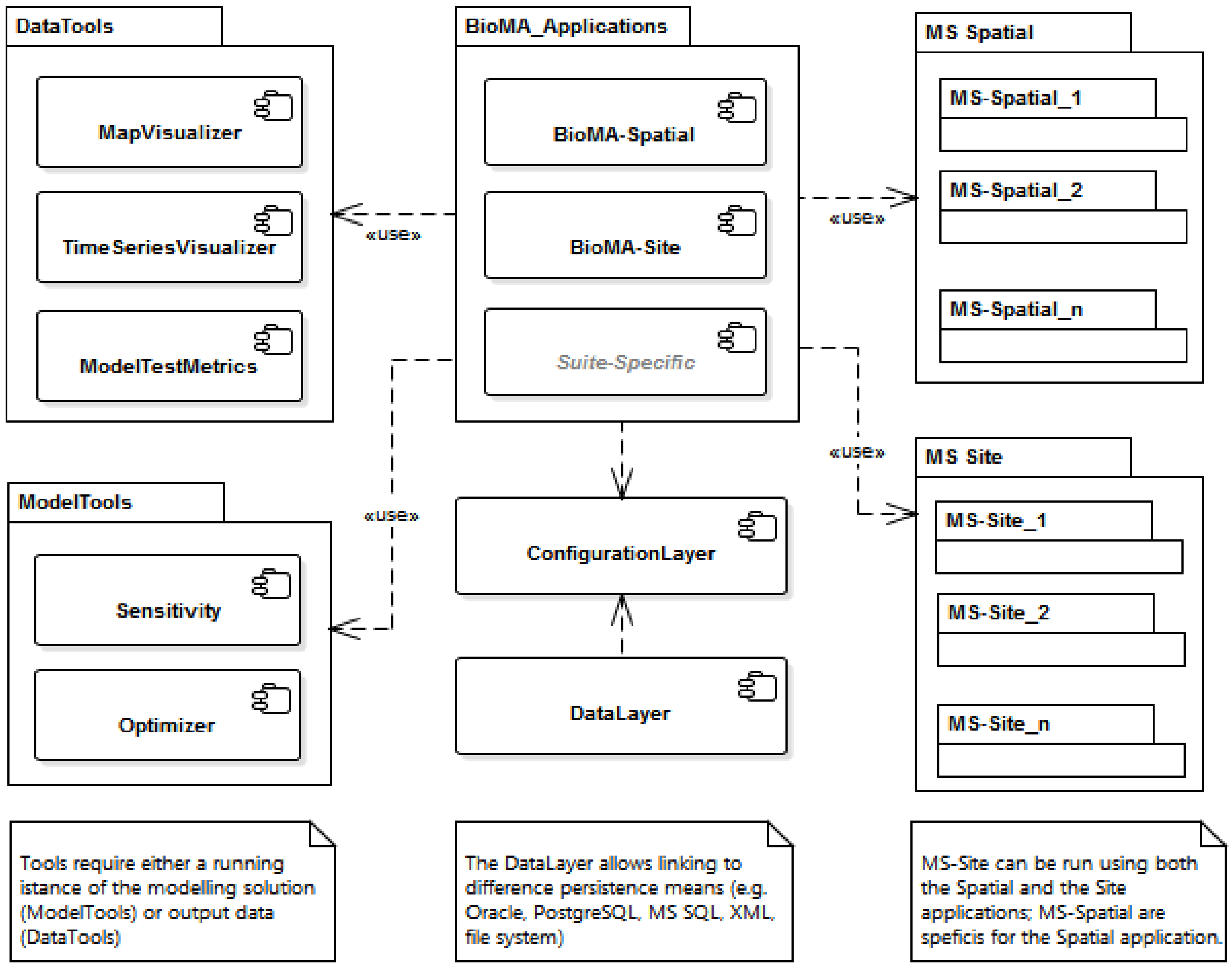
Impact models responding to *AgroManagement* events in crop/plant, soil, diseases, agro-chemicals models

Model components

- Model components built with the BioMA architecture:
  - Are reusable across frameworks;
  - Have a semantically explicit interface;
  - Can be extended both for data and models;
  - Include the definition of their own parameters;
  - Allow running pre- and post-conditions;
  - Have a scalable logging.

- They are a way to share knowledge while providing operational software units, to be used alone or via composition.

BioMA Deployments



The IPR model

- Working with a model framework requires investing resources, and it requires a medium-term perspective;
- No institution will do it on a code base of core components which are owned by someone else and which have code not accessible;
- BioMA will adopt a **LGPL version with open source access to core components**;
- A consortium is being developed.

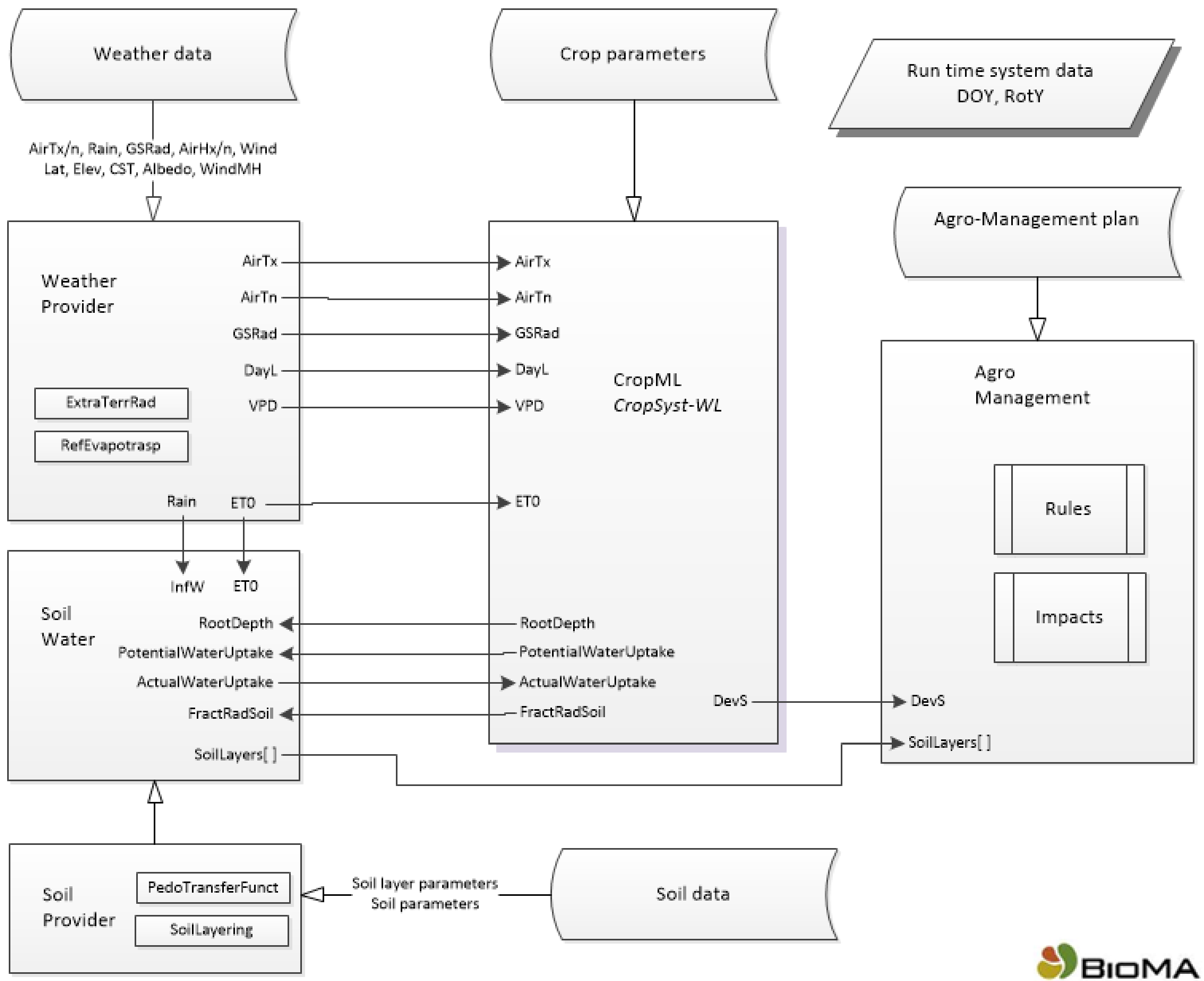
BioMA applications

- BioMA applications have been used for different research projects (https://en.wikipedia.org/wiki/BioMA):
  - weather datasets for biophysical simulation
  - CC impact on crop production and adaptation in Europe
  - soil pathogens under climate change
  - corn borers under climate change
  - modelling solutions comparison at sub-model level
  - impact of CC on crop production in Latin America
  - fungal infections under CC
  - estimate agro-meteorological variables
  - functions to estimate soil hydraulic properties
  - quality of agricultural products

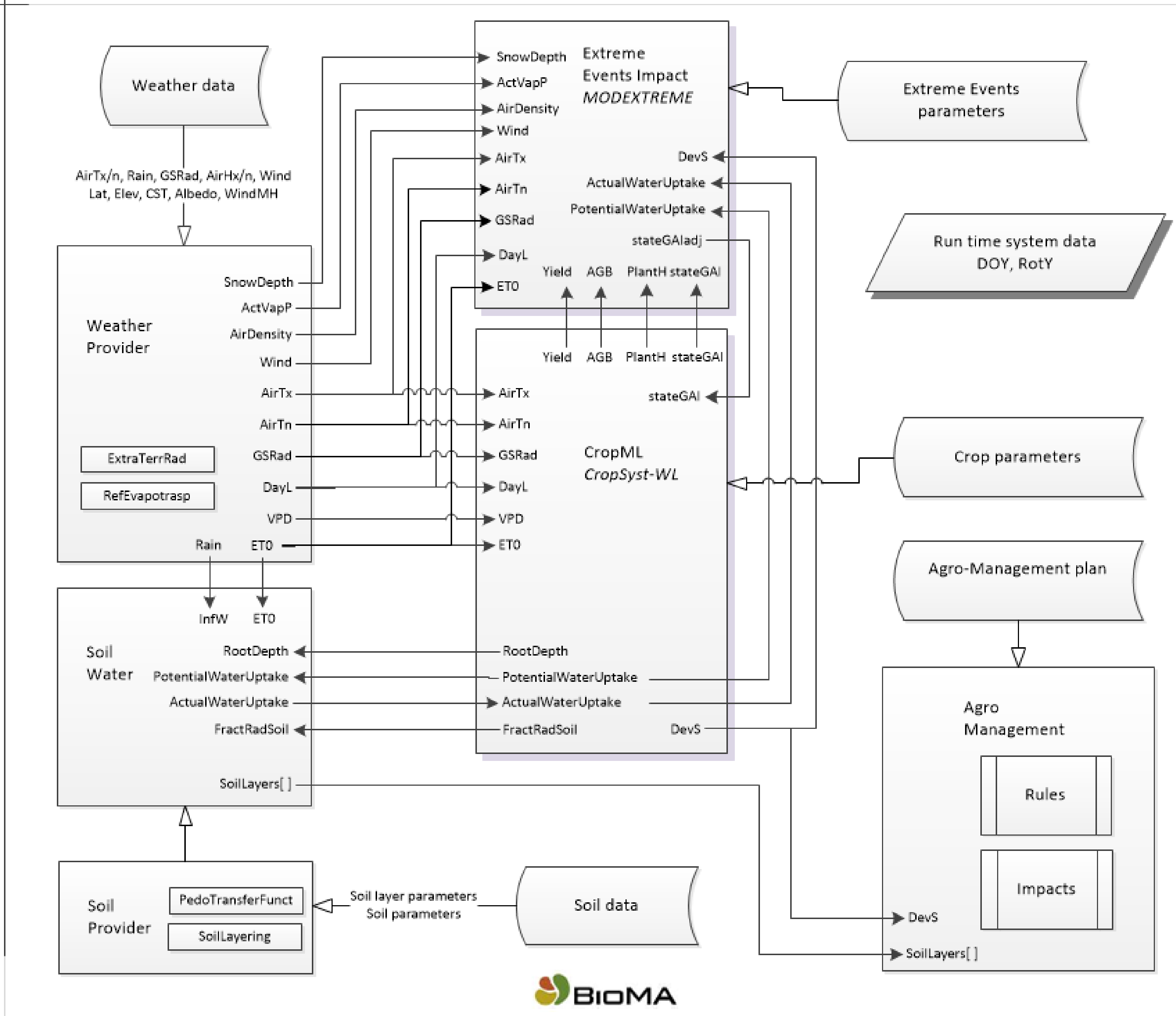
BioMA for MODEXTREME

- The need of extending simulation capabilities is key for the MODEXTREME analyses.
- Traditional modelling solutions need to be compared to the same approaches adding the models developed to better account for «extreme» events.
- Modelling solutions envisioned in the DoW are:
  - CropSyst and CropSyst+ModExt.ExtremeEvents;
  - Wofost and Wofost+ModExt.ExtremeEvents;
  - WARM and WARM+ModExt.ExtremeEvents;
  - OLIVECAN and OLIVECAN+ModExt.ExtremeEvents,

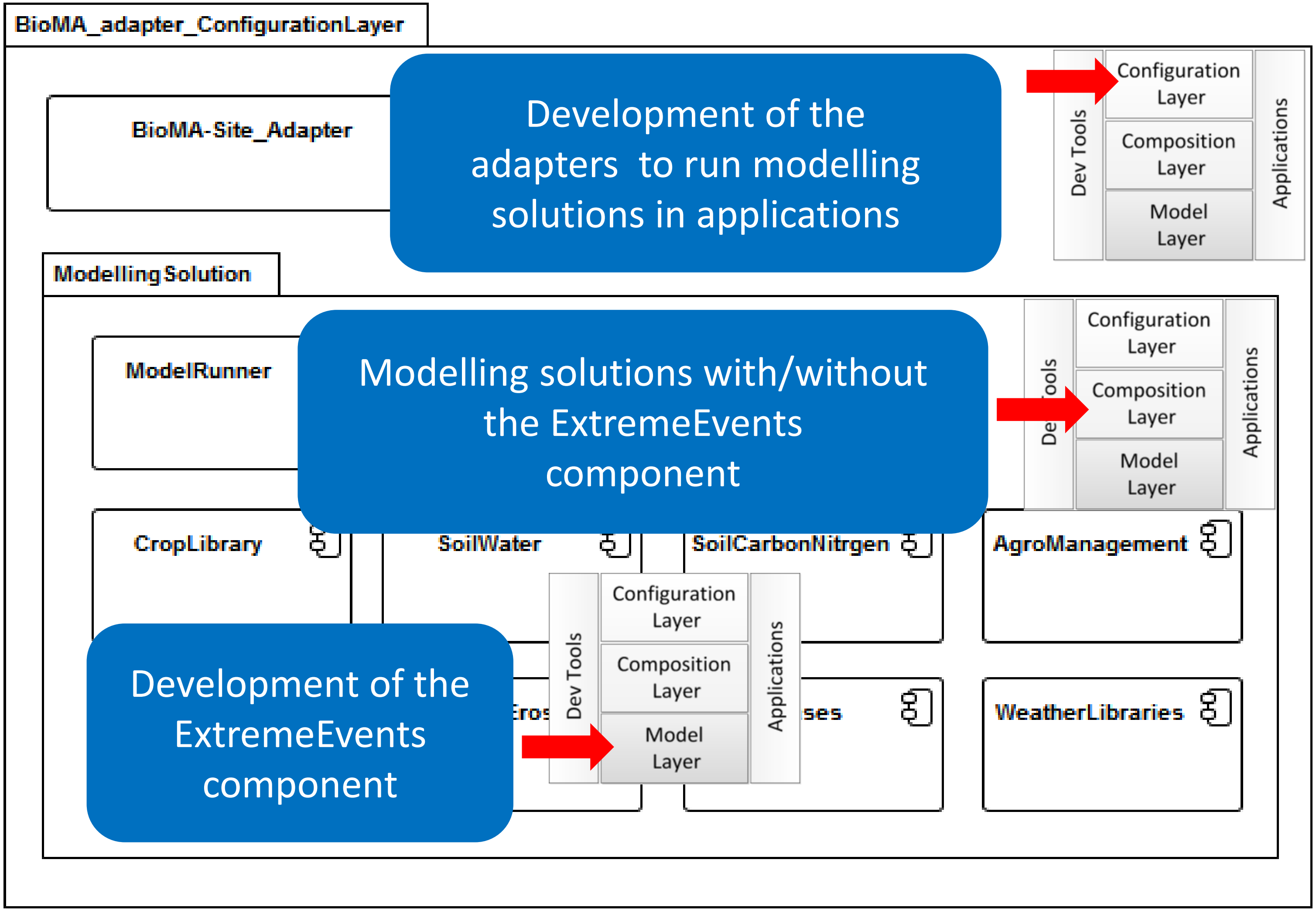
Links between components in modelling solutions (CropSyst water-limited)



Links between components in modelling solutions (CropSyst water-limited + Extreme events)

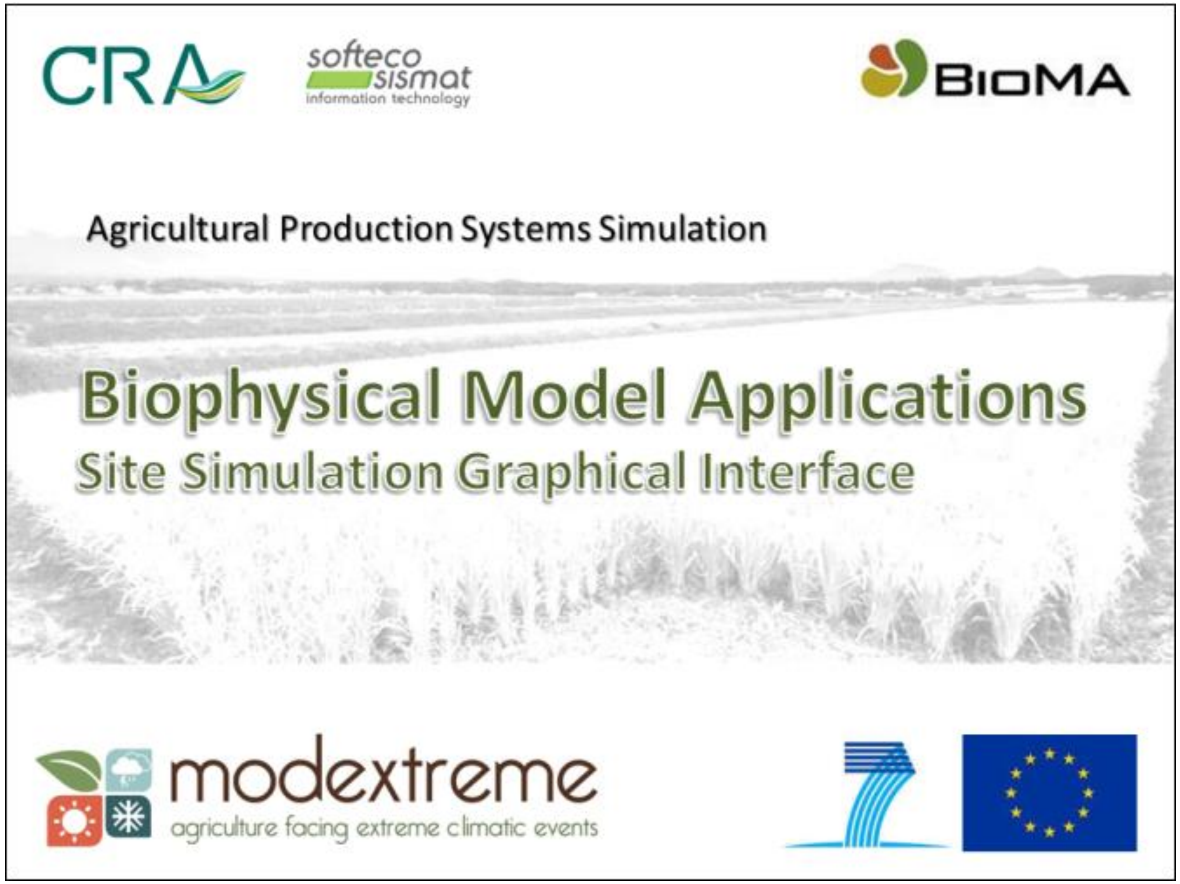


Models in MODEXTREME



Multi-model runner: BioMA-Site

Site



BioMA-Site

- BioMA-Site is a multi-modelling solutions runner, open to load any modelling solution respecting the interface required by the software framework.
- The modelling solutions running in BioMA-Site can also run in the application BioMA-Spatial, that is, iteratively against explicit spatial units.
- Also it is provided of tools and utilities to evaluate models against detailed dataset collected in field experiments.
- Modelling solutions other than the examples provided can be loaded, becoming simulation options; if they are alternate solutions to a specific modelling problem, BioMA-Site can also be used to compare their performance.

BioMA-Spatial

- BioMA Spatial is the Graphical User Interface developed within the BioMA Framework that allows configuring, running, and viewing instances of a Modeling Solution in a spatially and temporally distributed context.
- BioMA Spatial includes several plugins, such as Map Visualizer, Model Parameters Editor (MPE), Graphic Data Display (GDD), and Model
- Component Explorer (MCE) that you can launch and use from within the application for inspecting model interfaces.
- Furthermore, as for BioMA-Site, users can deploy other plugins to be used within BioMA-Spatial, such as Optimizer for optimization and LUISA for sensitivity analysis.

Acknowledgement

"The research leading to these results has received funding from the European Community's Seventh Framework Programme – FP7 (KBBE.2013.1.4-09) under Grant Agreement No. 613817, 2013-2016"