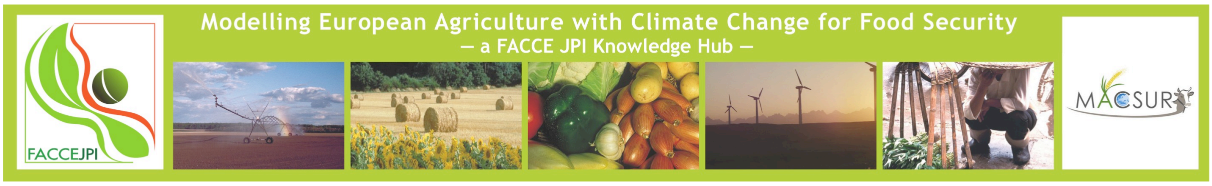


FACCE-MACSUR

D-H1.1 Review of Cloud Computing Opportunities

Jason S. Jorgenson[1]*

[1] University of Reading, PO BOX 237, Reading, RG6 6AR, United Kingdom

*j.s.jorgenson@reading.ac.uk

Instrument:	Joint Programming Initiative
Topic:	Agriculture, Food Security, and Climate Change
Project:	Modelling European Agriculture with Climate Change for Food Security (FACCE-MACSUR)
Due date of deliverable:	Month 12
Submission date:	2013-03-05
Start date of project:	1 June 2012
Duration:	36 months
Theme, Work Package:	Hub WP 1
Deliverable lead partner:	U Reading
Deliverable reference num.:	D-1.1
Deliverable type:	Report
Confidential till:	–

Revision	Changes	Date
1.0	First Release	2013-03-05

Abstract/Executive summary

This paper will begin by defining some of the challenges that we face on the MACSUR project in terms of evaluating model uncertainty and carrying out model integration. I will briefly review what cloud technologies are available, followed with some suggestions about how those cloud technologies can be used in order to contribute to meeting the challenges set out in the first part of the paper.

'Month 12' deliverable for WP1 is a review of the opportunities for using cloud computing to develop the potential for model inter-comparison and interlinking in MACSUR. A challenging aspect of compiling this review is that before an 'opportunity' for any kind of model linking/comparison can be identified, a lot of information about the specifics of extant models and workflows must be gathered from each of the three themes (TradeM, CropM, and LiveM).

This deliverable must, however, be more than just saying 'these are the computing tools that we can use to ...'. There are a number of different challenges at different levels; a hierarchy of challenges, if you like. For example, in order to get models 'talking' to one another, adequate protocols for the transference of data and scaleability will need to be established, and then things like uncertainty analysis for these integrated models will need to be addressed. Further issues exist relating to human behaviour and logistics (e.g. MACSUR is a large project with many members from all over Europe, with substantial distances between many of it's members).

The term "Cloud" is very ambiguous, and Cloud Computing covers a huge range of services, and a number of innovative tools exist which can make international collaborative research more effective. Two examples (already implemented on the MACSUR website) are: a discussion forum (where project members can create topics, make or reply to posts, and upload documents) and a complete surveying platform (to provide an un-restricted and fully featured survey platform for MACSUR members' information gathering needs.)

Table of Contents

Overview

As outlined in the MACSUR project proposal, an important element of WP1 is to make model inter-comparison and interlinkage more straightforward by exploiting developments in IT infra-structure, with a particular emphasis on cloud computing (WP1 Task 1.1). The first deliverable (D1.1) in WP1 is a review of the opportunities for using cloud computing to develop the potential for model inter-comparison and interlinkage in MACSUR. Following a brief discussion of the rather ambiguous term "cloud computing", this paper identifies some of the fundamental challenges associated with integrating and comparing different models, and then identifies several extant cloud based tools that could help facilitate model integration and inter-comparison.

Cloud Computing

> *Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for an optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized SLAs.* -- *Vaquero et.al. (2008)*

The MACSUR consortium is widely distributed across Europe, and this factor alone almost certainly suggests that any attempt to integrate it's members and their work could benefit from some form of cloud based technology. But what is meant by 'cloud'? 'Cloud computing' is a very ambiguous term, and it is difficult to find a succinct and universal definition. When looking for a definition, one comes across the letters "aaS" a lot, which stands for "as a Service".

Software (SaaS), storage (STaaS), data (DaaS), databases (DBaaS) and even entire computing platforms (PaaS) are just a few examples of hardware and software resources that can be delivered 'as a service'. Perhaps more simply, I could argue that any computer tasks done on a remote system via a network connection of some kind from a user's local machine can be called cloud computing. For the purposes of this paper, however, cloud computing is going to be left loosely defined, for the over-arching aim is to explore any potential tools and methods to help facilitate the integration of models (and the people using these models) across the three themes in MACSUR, and could risk precluding some options due to the associated restrictions imposed by a rigorous definition.

Key Challenges

> *'As we look into the short-term future, we are reminded of one of Jim Gray's well-known quotes: "May all your problems be technical." With this ironic comment, Jim was indicating that behind even the most difficult technical problems lies an even more fundamental problem: assuring the integration of the cyberinfrastructure into human workflows and practices. Without such integration, even the best cyberinfrastructure will fail to gain widespread use.'*
>
> -- Van de Sompel, H., & Lagoze, C. (2009)

When we consider that within the MACSUR consortium there are over 75 different models being used by 100+ modellers belonging to three distinct groups spread across the whole of Europe, it is obvious that there will be no shortage of challenges. For the purposes of this report, these challenges will be first quite simply categorised into just two primary categories: technical and non-technical.

Non Technical Challenges

There is an obvious need to consider the socio-technical problems associated with any cyber-infrastructure intended to be useful for, and used by, such a broad variety of people. Success will require knowledge of not just the technology involved, but also an understanding of how this technology integrates into the communities of use.

- **'Non' Technical**
  - Sharing
    - General Reluctance
      - model not yet published
      - model not yet 'finished'
      - too user 'un-friendly'
    - Licensing
      - Uncertain of restrictions
      - Multiple licenses
  - Spoken Language
  - Geographical Distances
  - Lack of awareness

When people are given a new new tool, when they use it they tend to adapt it to the practices and priorities of their own contexts and communities of use i.e. other, older tools and media, and other people (Chalmers 2004). Even so, regardless of how the tool is used, it means that there is a common platform through which their work can be more easily linked, used, compared or shared with other users of the same tool. Some of the challenges are simply going to be overcoming perceptions. How can we overcome, for example, a general reluctance to 'share' a model? A model may have been developed 'to work', which it might do brilliantly, but less attention was paid to ensuring that the source code standards and documentation. Licensing issues, different data sources, and the anticipation of needing to respond to 'support questions' add more complexity, and all of this equals TIME. So for any tool to be useful, the payoff has to be worth the time investment.

One of the first tasks each of the three themes had was to compile an inventory of models. In short, the non-technical challenges are really about communication and standardisation. Recommendations for solutions/tools, however, can only be given once an understanding of current practices has been observed and understood.

The inventory task was very useful in this regard. Collaborative tools for LiveM, CropM and TradeM to decide what information was wanted, and plan what questions to ask, so that identical surveys/questionnaires could have been used would have resulted in much more consistent results (when considered as a whole, across the three themes).

There are cloud based tools which can help with this. For example, Google Docs has a collaboration feature, where multiple users can simultaneously work on the same document, leave comments, and chat in real time. Another possible method might have been to use forums (now been implemented on the MACSUR website).

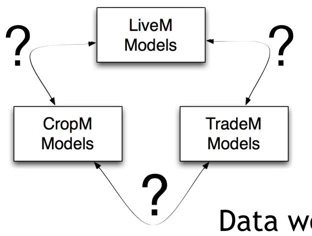
For collecting the data, a powerful survey system has been integrated into the MACSUR website (LimeSurvey). Even if separate surveys had been used by each theme, it still would have produced results in a consistent format, which would make working with the data much easier. Now, in order to proceed, data collected from the three themes has to be integrated, but for this to be meaningful, another poll must be done to fill in the gaps. Had the collaborative tools been agreed on and implemented in time, this could have prevented the need for a second round of enquiries, and thus saved, collectively, a substantial amount of time and effort.

Technical Challenges

Linking models is not the only technical challenge being faced. Legal aspects of data use, software licenses, or a combination of the two need to included in any integration processes. Of course the development of the models and work of the scientists must not be impeded either. As stated before, the possibility of working with less than perfect (or perceived as less than perfect) code, or in a different language (code and/or spoken), or on a different operating system (Linux, Mac OSX, Unix, Windows...) is possible, probably almost certain in fact, and addressing all of these factors presents a pretty daunting technical challenge. That being said, however, there is a light at the end of the tunnel.

- ☺ **Technical**
  - Licensing
    - Restrictions on input data
    - Restrictions on model distribution
    - Multiple licenses
  - Development Environment
    - Different programming languages
    - Different Operating Systems
    - Poor (or absent) documentation
  - Model Inputs
    - File formats
    - Naming conventions

Integrating Models



From a technical point of view, one challenging aspect is going to be linking models together while still preserving each model's associated development process/system, because models are being developed in different programming languages, on different operating systems.

Data would seem to be an obvious area which would benefit from cloud technologies. If all of the data were located centrally on the cloud, the models across the themes could easily be given access. This still leaves the model integration unaddressed, however, and also may be problematic again from a licensing perspective.

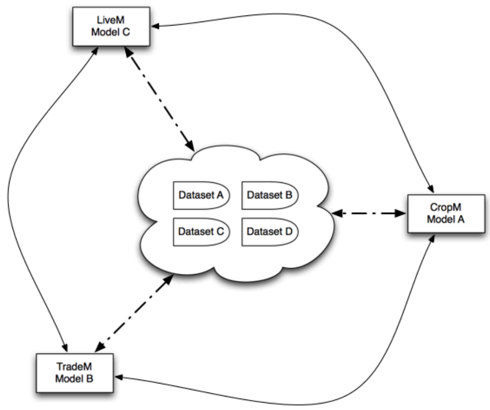


Diagram 1

Models could be integrated by rewriting them into one package in the same language (as in Diagram 2). This would work, but any changes/updates made to the original model

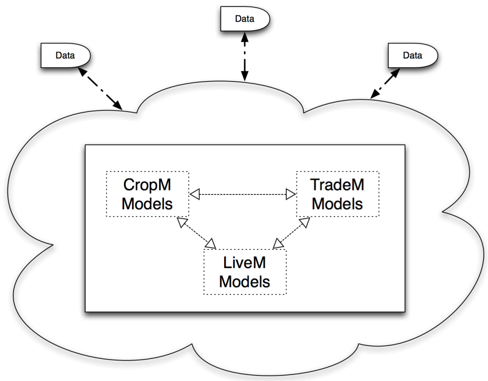


Diagram 2

(upon which the rewritten/integrated version was based upon) would not be reflected in the integrated system. Different data sources/formats could again be problematic from technical and/or legal points of view.

Another option might be to create a 'magic cloud' which models and data connect to that sorts everything out (Diagram 3). This sounds quite nice, and actually it is this route which may prove most fruitful. There would be a massive amount of complexity in this magic cloud, however, as it would have to be able to know the unique details of each and every model and data source it works with. On top of that, it would also need to know how to auto-magically convert the different formats and scales of the inputs and outputs to and from one model to another. This method would be extremely complicated, have massive computational requirements, and be rather expensive. More thought needed!

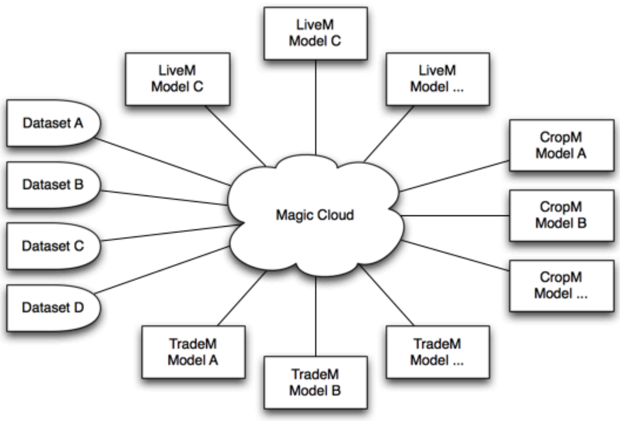


Diagram 3

If a standard was developed which enabled the inputs and outputs of any model to be defined, it could be used to create an integrator (Diagram 4). This would take much of the complexity and overhead out of the magic cloud described above, but again, this doesn't necessarily address the problems associated with different operating systems, bandwidth, etc. Plus, magic probably isn't the best approach to take, scientifically speaking!

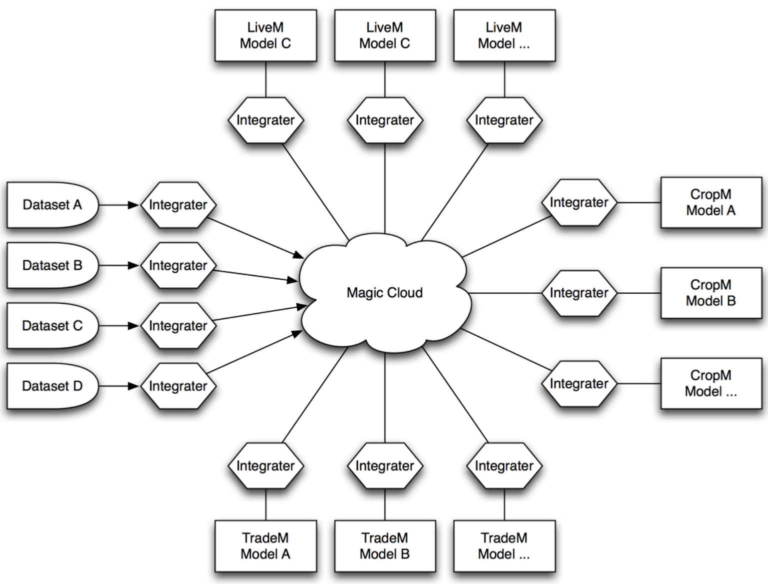
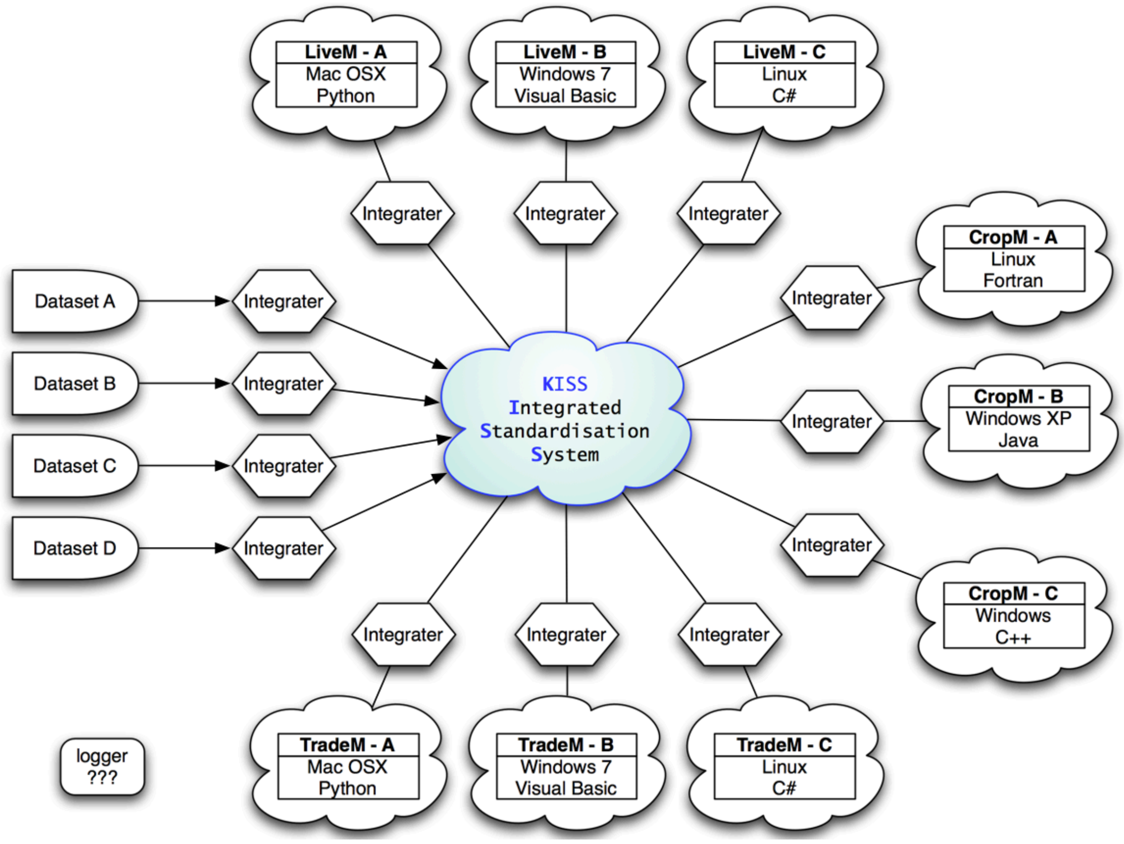


Diagram 4

If models are not able to be made to run on the same computer, how can they be integrated? Different platforms, different versions of libraries for compiling, different languages.... there is a solution.

Keep It Standard & Simple

By using virtual machines (VMs), it makes it possible to consolidate all of the models onto one system. These VMs could be accessed by their users remotely, either for use or development, but because they are all centrally located, bandwidth becomes a non issue.



Additionally, developers and users will not be faced with learning how to use a different operating system, or another programming language. Instead, each VM will be connected to one central system via a standard integrator. This integrator is simply a strict set of protocols which allow inputs, outputs, data, scale, etc to be defined. The model developers need to do two things for this to work.

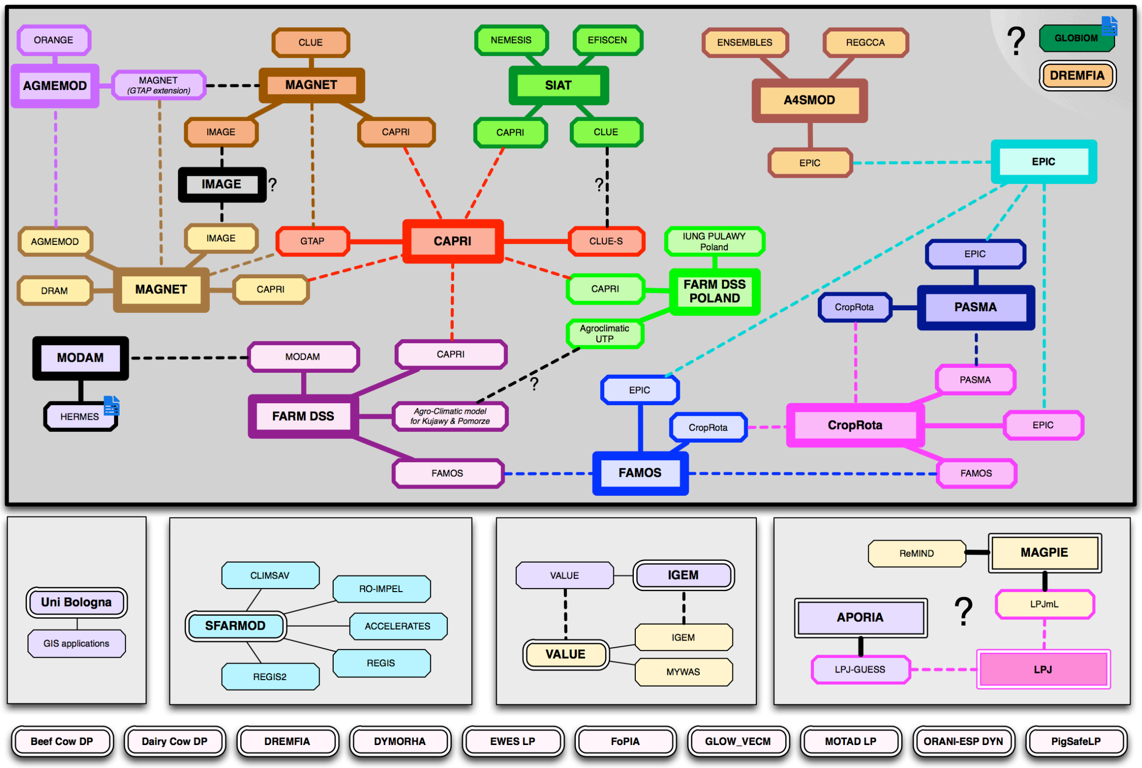
1. Create a VM on the server, and set up their model
2. Create the Integrator (by providing details of their model's parameters using the standardised 'terms')

Using the KISS method will give models written in different languages and on different platforms a platform which will make integration possible. A KISS can lead to a lot of different things; for example, it could save on computing time by indexing and store results of previously run scenarios, or make it possible to perform uncertainty analysis between different models in the background, or log data/results for any number of different reasons.

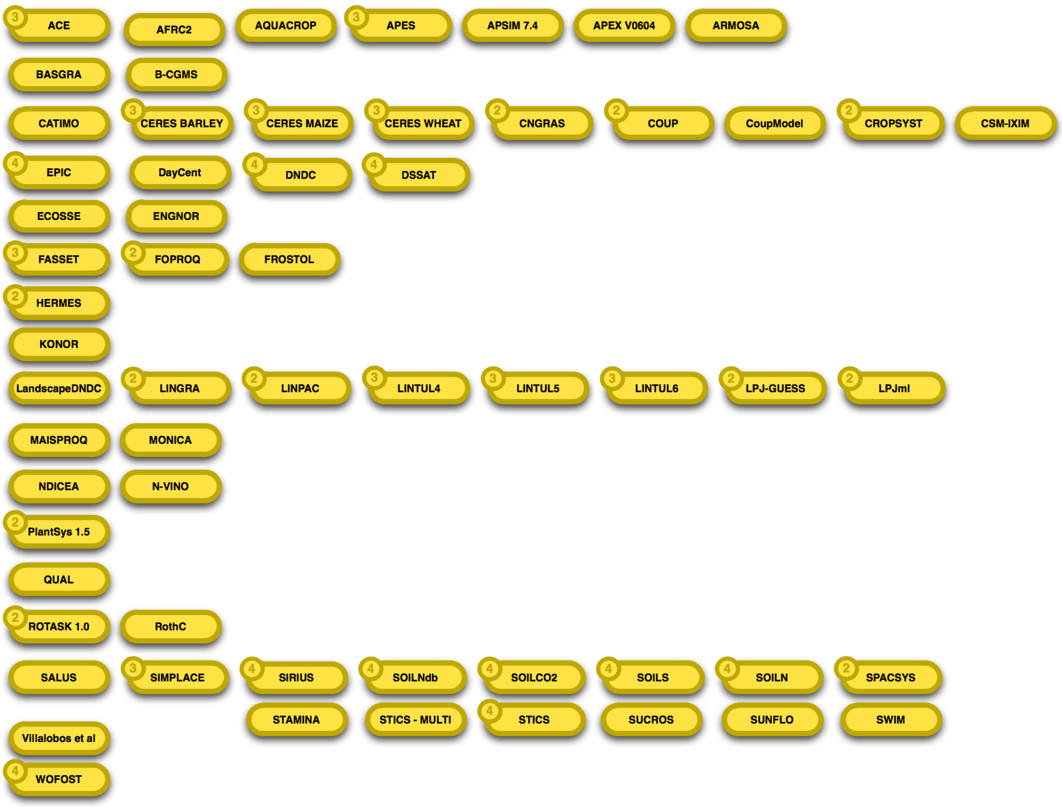
Although this system is still in the conceptual stage, and the relationships between the three themes are still being formed and defined, it is hoped that it will provide a platform that will help these relationships to advance, and facilitate collaboration and integration!

Preliminary CropM/TradeM model inventories/linkages

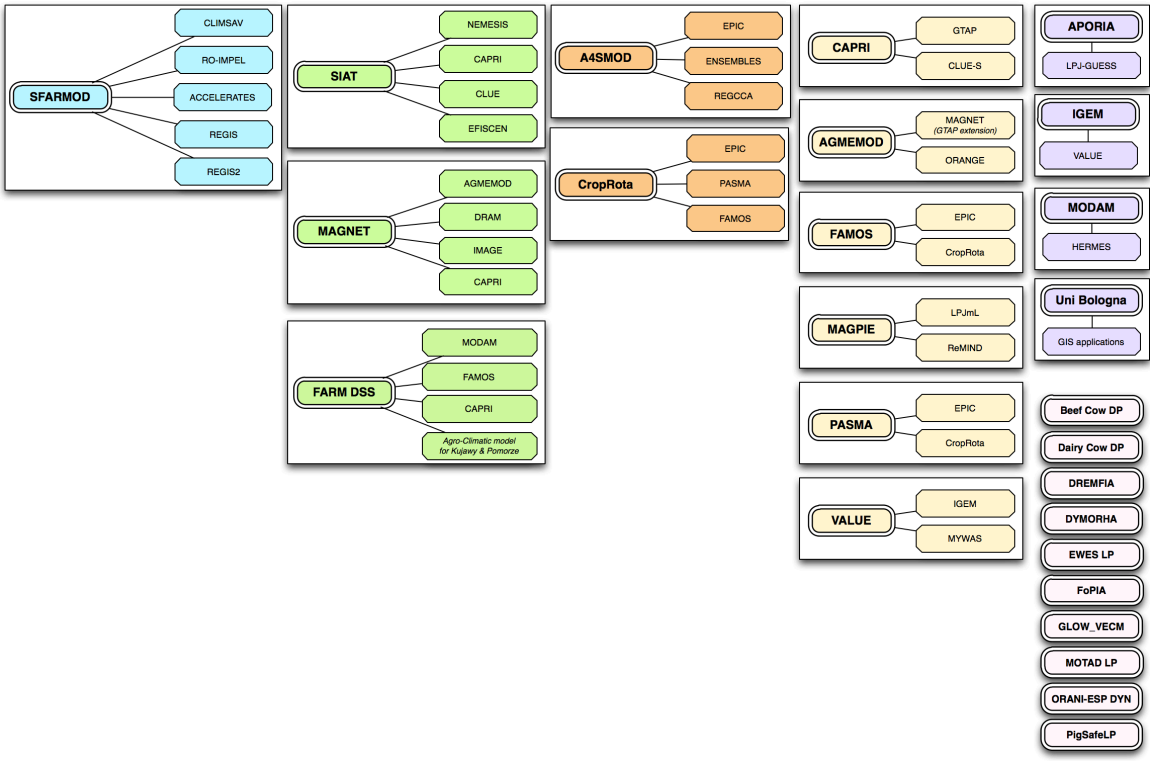
Trade Models (with linkages)



Crop Models



Trade Models (no identifiable connections with external projects)



Acknowledgements

This paper is a contribution to the FACCE MACSUR knowledge hub.
The work was funded by Biotechnology and Biological Sciences Research Council (BBSRC).

References

Chalmers, M. (2004). A historical view of context. *Computer Supported Cooperative Work (CSCW)*, *13*(3-4), 223–247.

Van de Sompel, H., & Lagoze, C. (2009). All aboard: toward a machine-friendly scholarly communication system. *The Fourth Paradigm,* 193-199

Vaquero, L. M., Rodero-Merino, L., Caceres, J., & Lindner, M. (2008). A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, *39*(1), 50–55. doi:10.1145/1496091.1496100