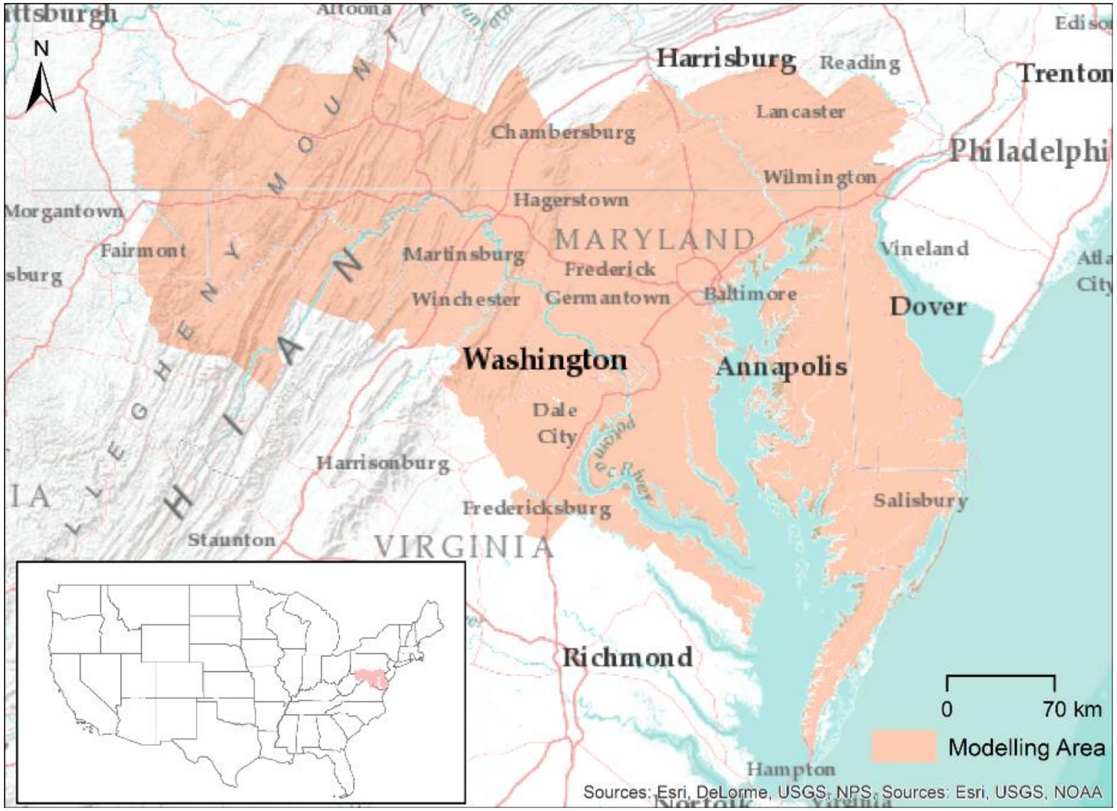
Coupling Diverse Models

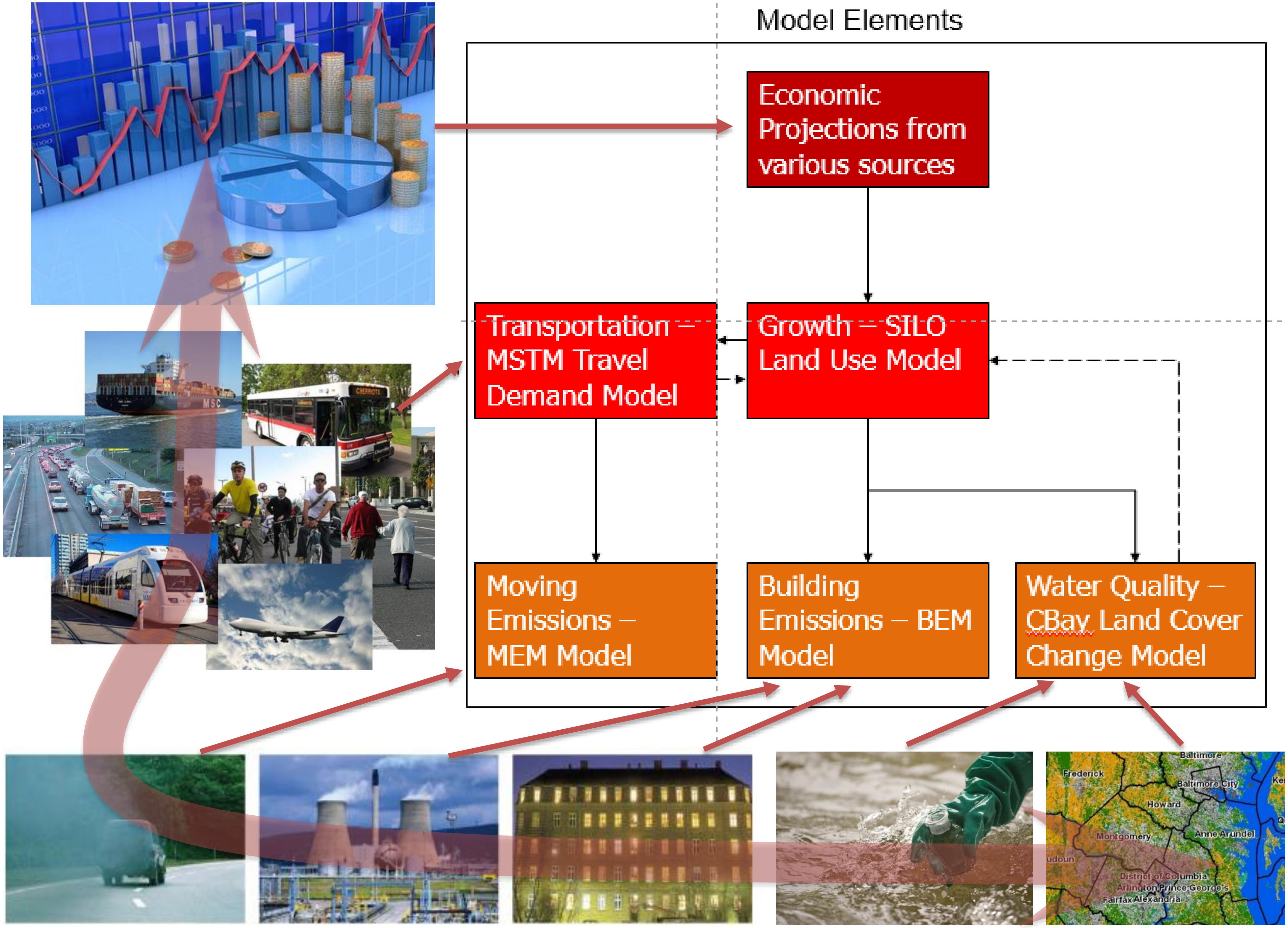
Case Study of Baltimore-Washington Region

Harutyun Shahumyan
Marie Curie Research Fellow
School of Architecture, Planning and Environmental Policy
University College Dublin
harutyun.shahumyan@ucd.ie

Modelling Area

- Chesapeake Bay is the largest estuary in North America stretching across 165,000 sq. km.
- The length of the coastline is longer than the entire US West Coast.
- The Bay's land-to-water ratio (14:1) is the largest in the world; thus land use has a big impact on the Bay's health
- As the host to the US capital and several large metropolitan areas the region is unique in terms of the competing interests among agriculture, land use change, urbanisation and transportation.





Model Elements
Economic Projections from various sources
Transportation – MSTM Travel Demand Model
Growth – SILO Land Use Model
Moving Emissions – MEM Model
Building Emissions – BEM Model
Water Quality – CBay Land Cover Change Model
Baltimore
Frederick
Baltimore City
Howard
Montgomery
Anne Arundel
District of Columbia
Arlington
Prince George's
Fairfax
Alexandria

Model Characteristics

NCSG

Model	Environment	Operation System	Developer/ Licensing	Number of runs	Time per run *	Overall Runtime
MSTM	CUBE	Windows	Scripts: Open source CUBE: CitiLabs	3	3 hour (16 hour)	9 hour
SILO	Java	Multi-platform	Open source	39	9 min	6 hour
MEM	CUBE	Windows	EPA (MOVES) / CitiLabs	2	30 min	1 hour
BEM	R	Multi-platform	Open source	2	30 min	1 hour
CBLCM	C / C++	CentOS Windows	USGS	2	3 hour	6 hour
* Server: 20 x AMD Opteron Processor 6328 @ 3.20GHz, 42GB RAM, Windows 7						**23 hour**

Processing flow order & simulation periods

2001-2007
- SILO

2007
- MSTM

2007-2030
- SILO

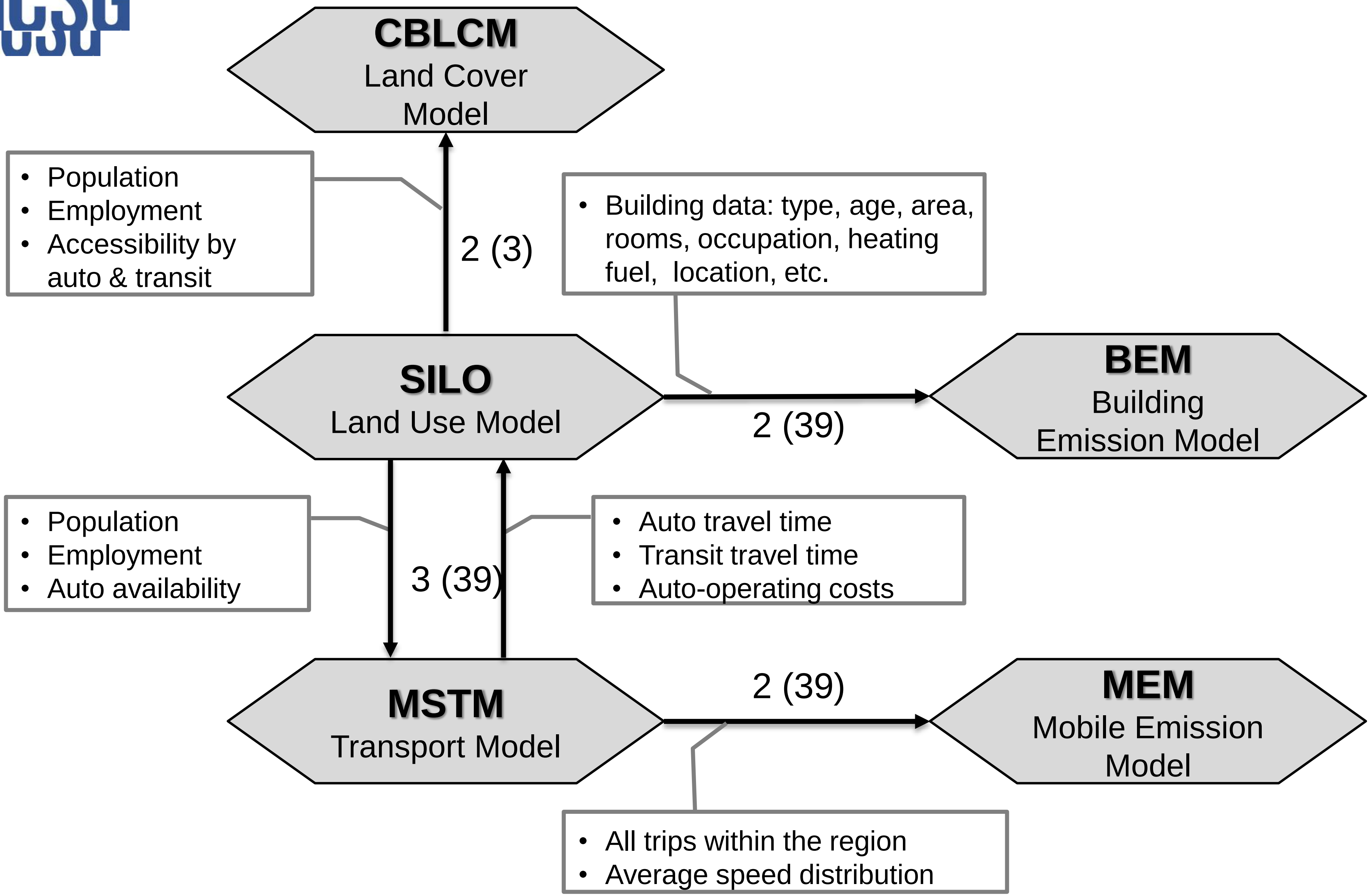
2030
- MSTM
- MEM
- BEM
- CBLCM

2030-2040
- SILO

2040
- MSTM
- MEM
- BEM
- CBLCM

Number of runs (simulated years)	
SILO	39
MSTM	3
MEM	2
BEM	2
CBLCM	2

Data flow between the models

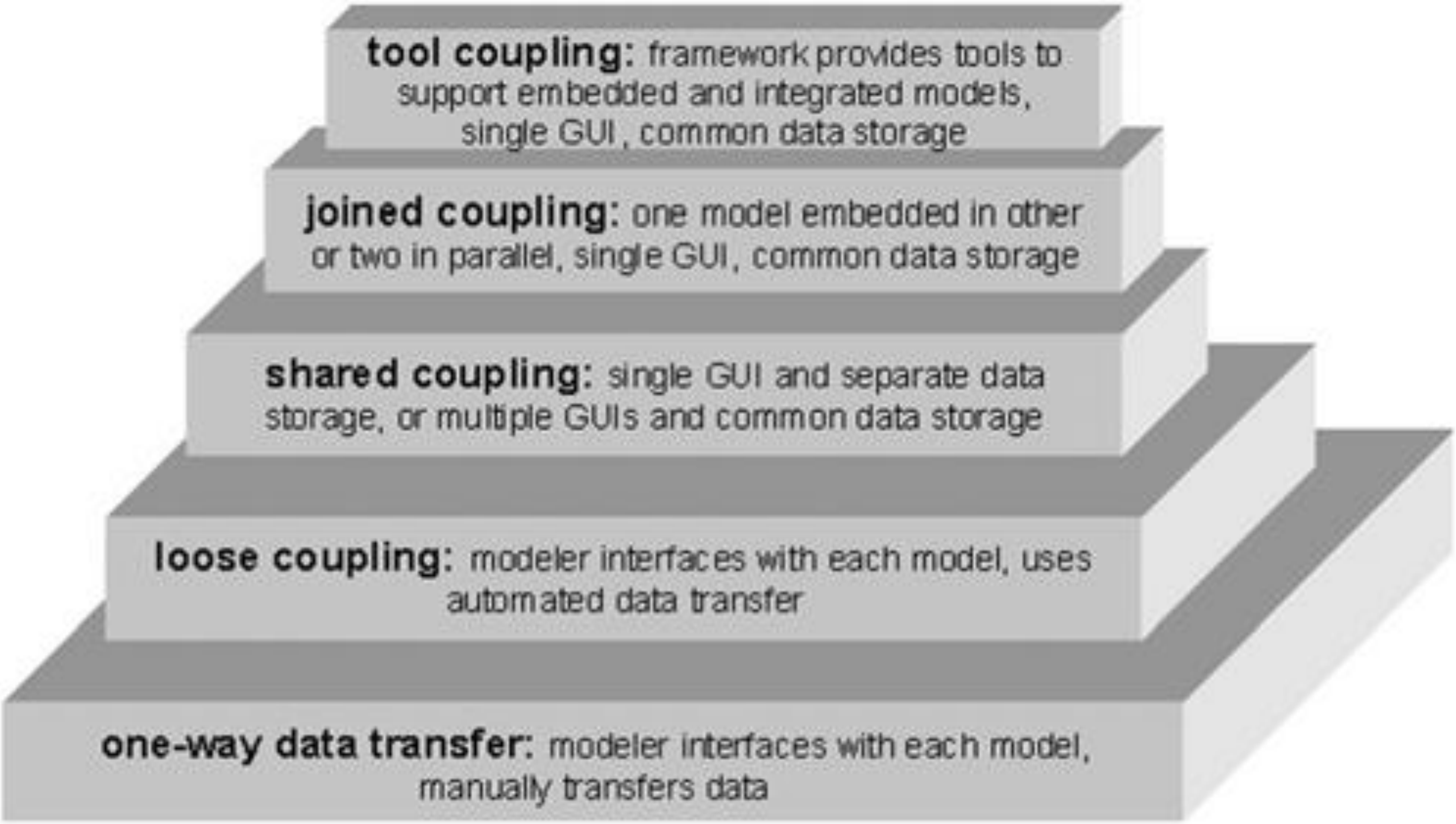


Key Requirements of Integration

NCSG

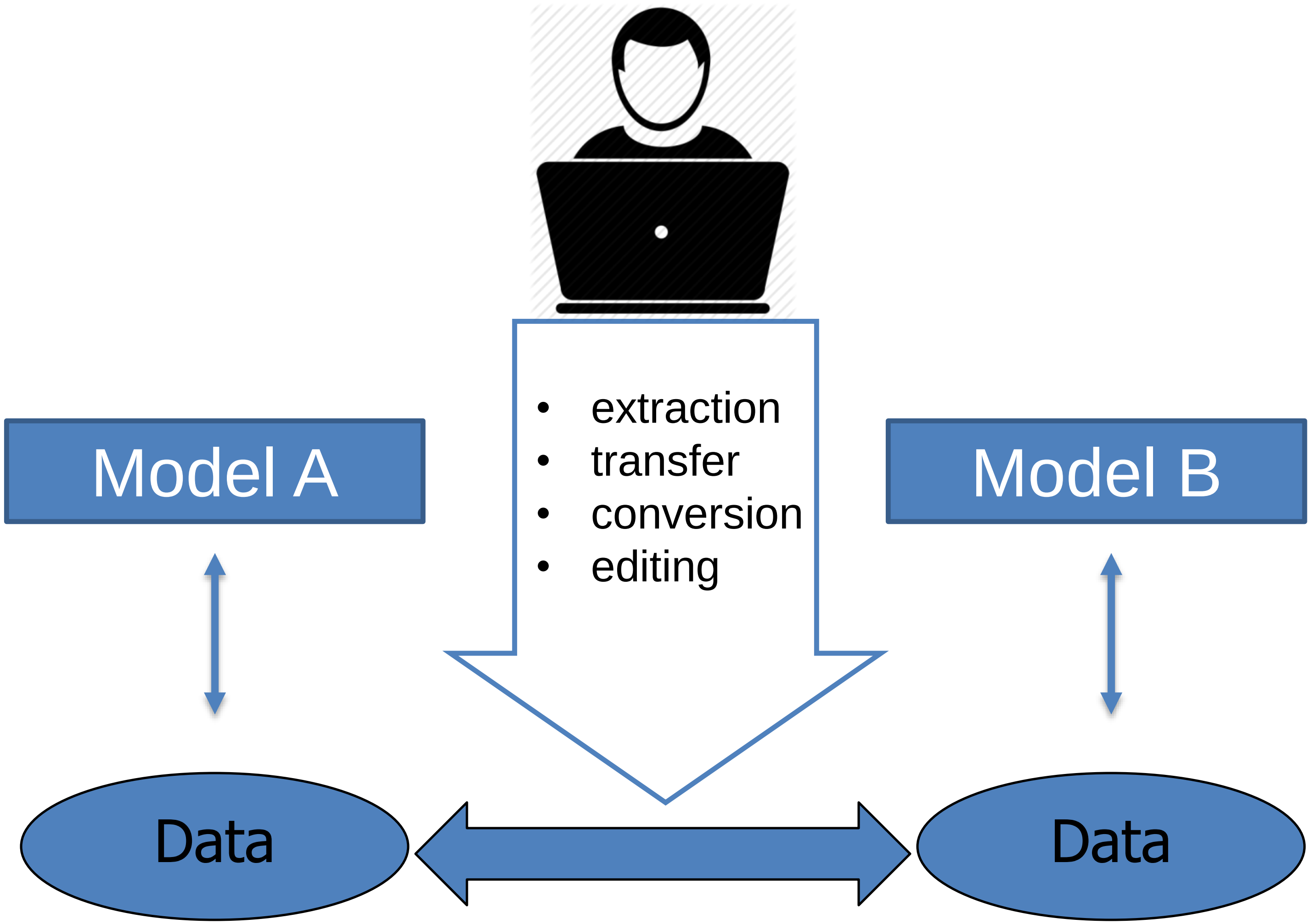
- Ability to develop models independently, such that they may be plugged-in easily.
- A modular approach supporting reusability and adding new components.
- User friendly graphical interface.
- Minimizing manual data transfer.
- Minimal or no change in source codes of the models.
- Capacity to link models developed in different programming languages and environments.
- Ability to deal with different licensing requirements.
- Compatibility with GIS for easy data visualization and spatial analysis.
- Minimal costs and efficient timing for implementation.

Progression of coupling methodologies

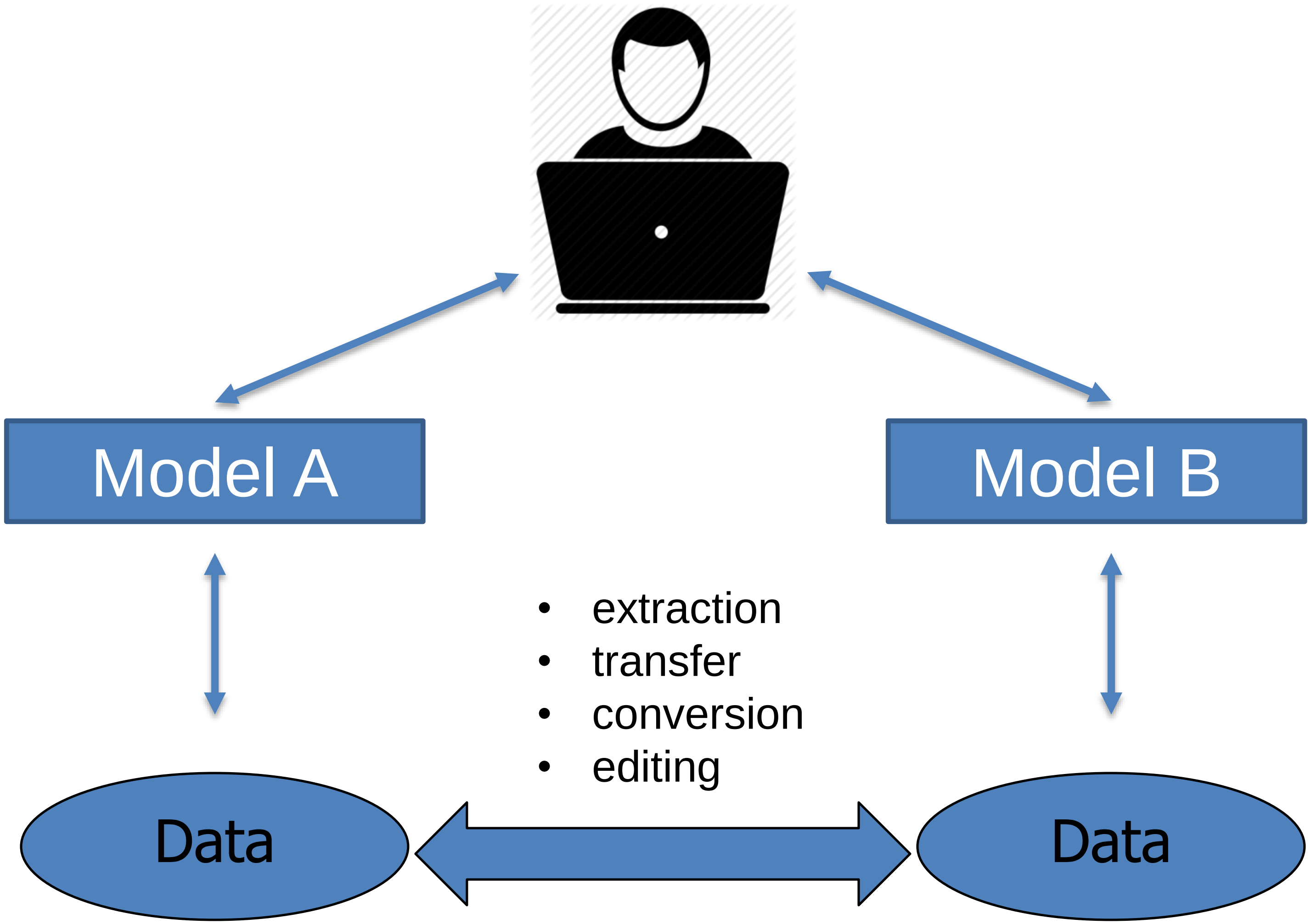


*Source: Brandmeyer, J. E. & H. A. Karimi (2000). "Coupling methodologies for environmental models." Environmental Modelling & Software **15**(5): 479-488.*

Manual Data Transfer

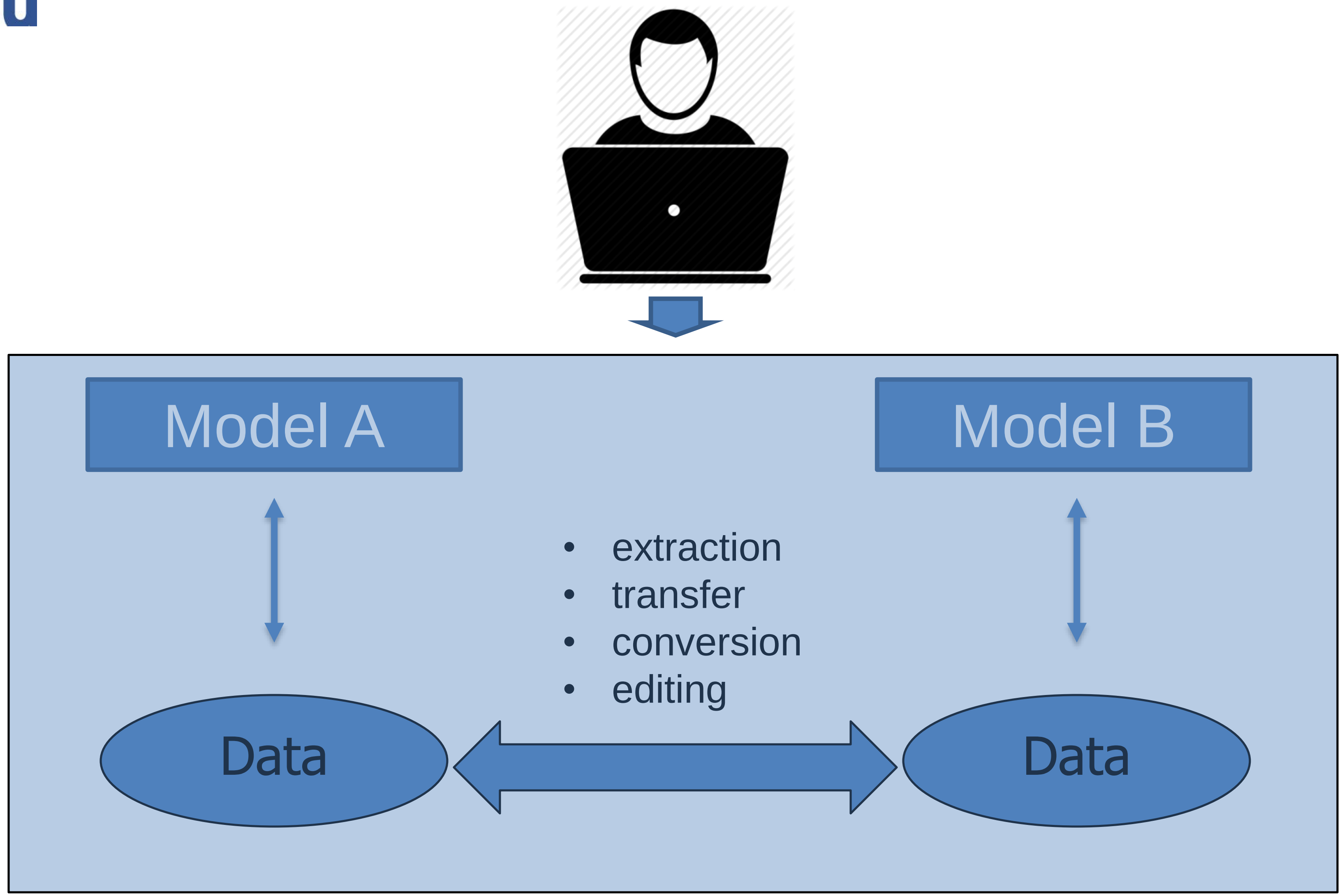


Loose Coupling

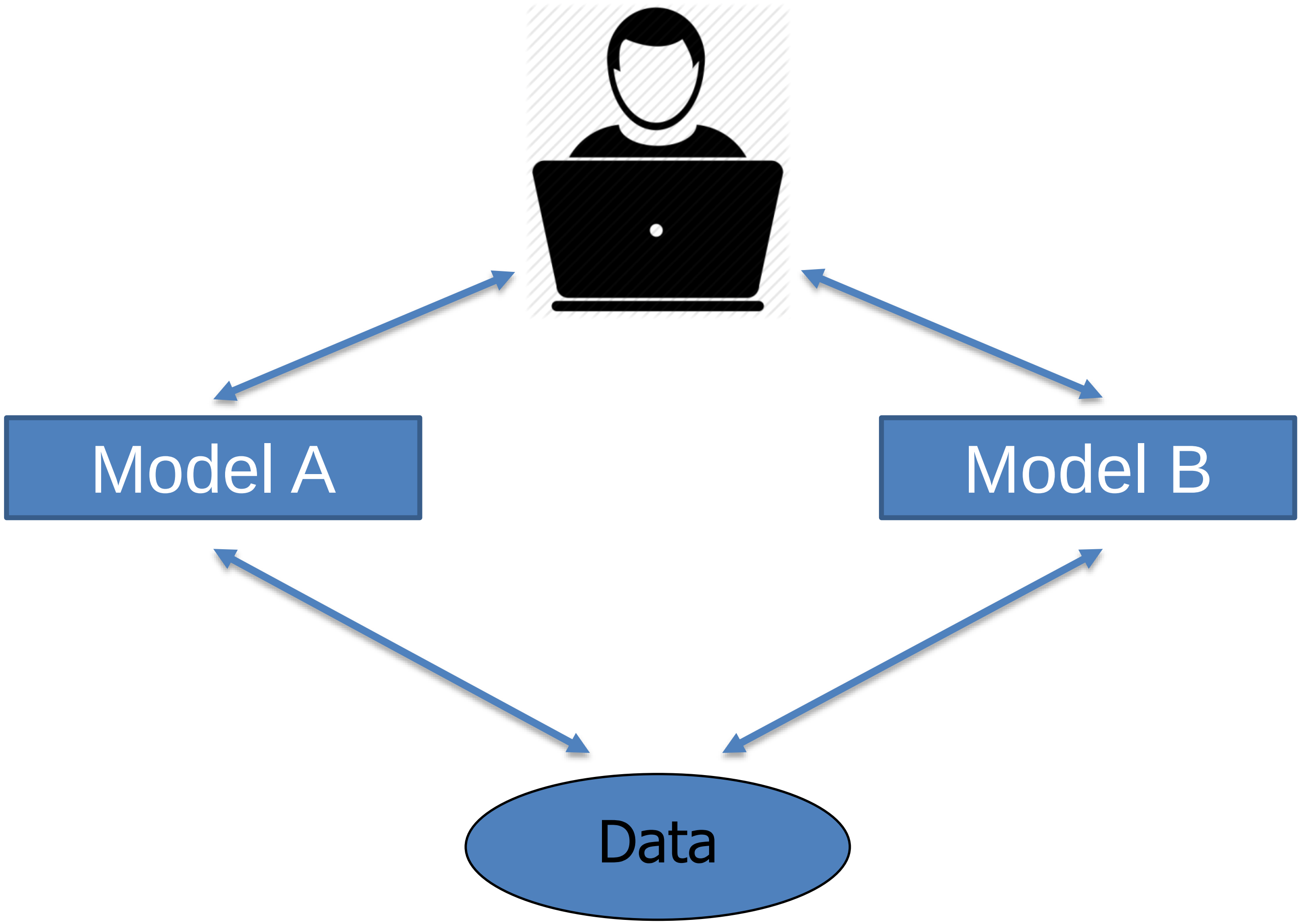


- extraction
- transfer
- conversion
- editing

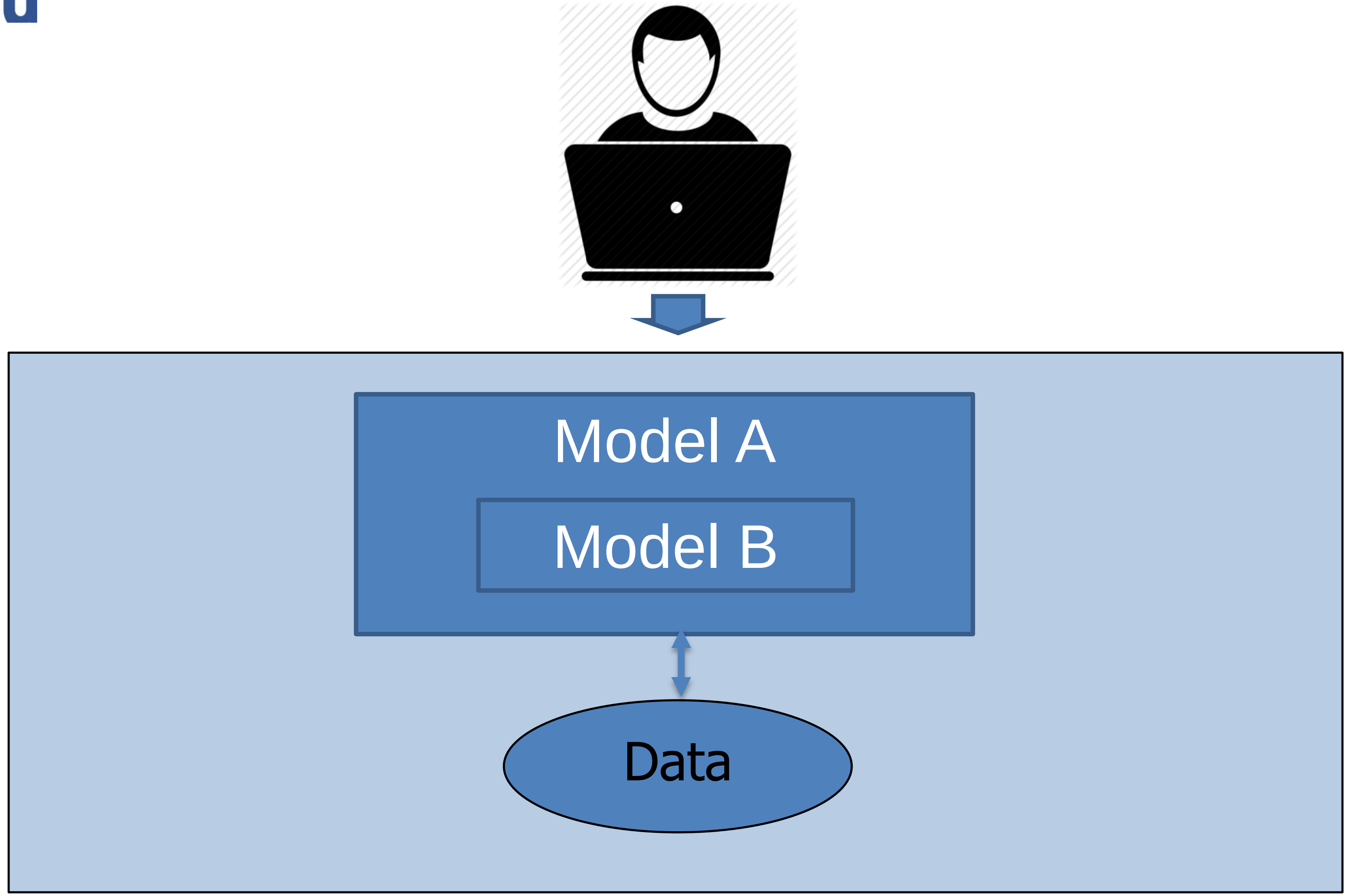
User Interface Coupling



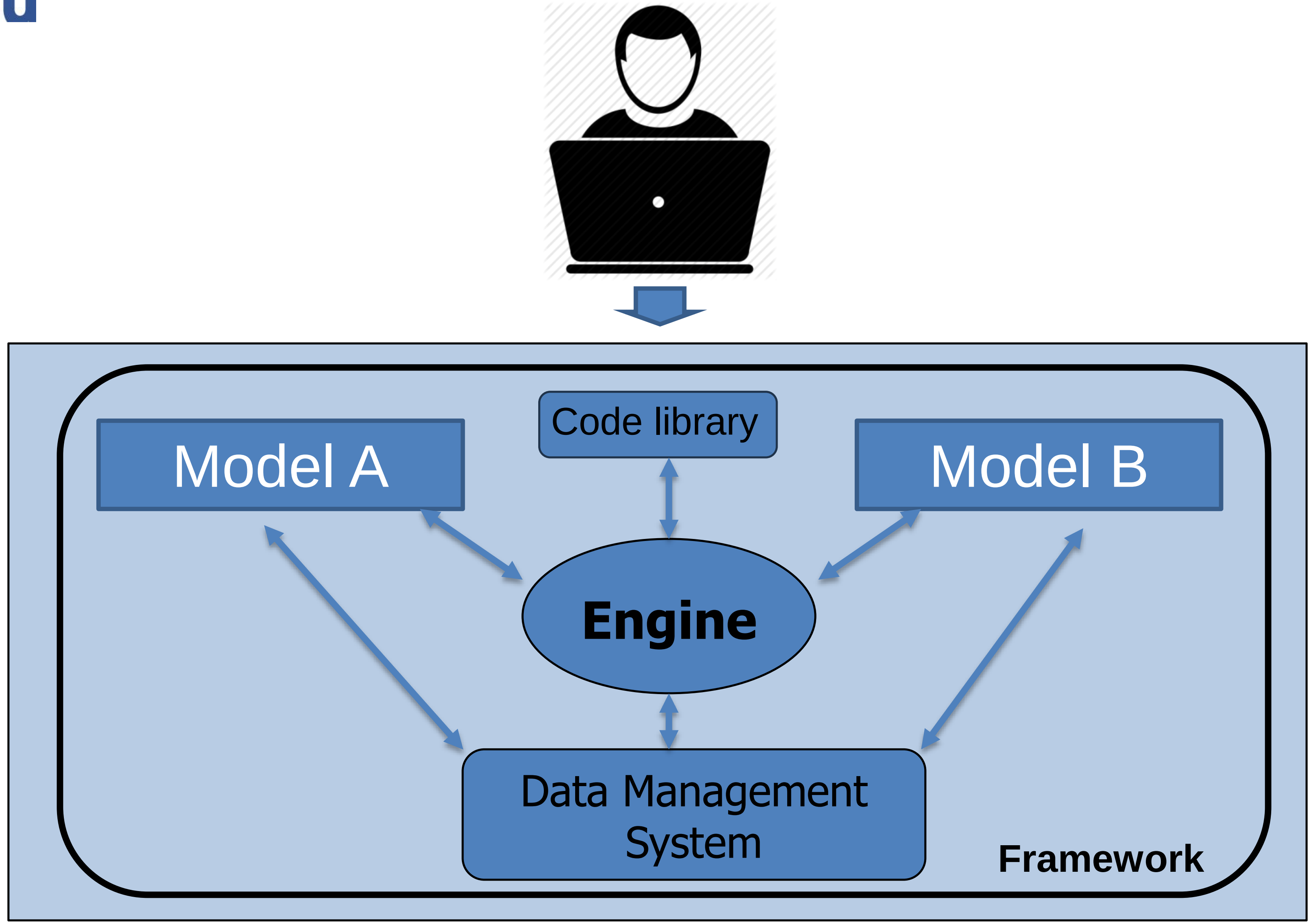
Data Coupling



Embedded Coupling



Tool Coupling



Model Coupling Tools

- Open Modelling Interface (OpenMI)
- Community Surface Dynamics Modeling System (CSDMS)
- Earth System Modeling Framework (ESMF)
- Model Coupling Toolkit (MCT)
- O-PALM
- OASIS
- FLUX
- Kepler

Open Modelling Interface

NCSG

OpenMI

http://www.openmi.org

Compliant components can

- be configured to exchange data during computation (at run-time)
- run simultaneously and share information at each timestep.

Linked components may

- come from different suppliers,
- represent data and processes from different domains,
- be based on different concepts,
- have different spatial and temporal resolutions and representations.

Implementation requirements:

1. Save the model in DLL format.
2. Separate initialization, Perform time step & finalization parts in the source code.

void	**Initialize**(IArgument[] properties)
string	**ComponentID**
string	**ComponentDescription**
string	**ModelID**
string	**ModelDescription**
ITimeSpan	**TimeHorizon**
IInputExchangeItem	**GetInputExchangeItem**(int index)
IOutputExchangeItem	**GetOutputExchangeItem**(int idex)
int	**InputExchangeItemCount**
int	**OutputExchangeItemCount**
void	**AddLink** (ILink link)
void	**RemoveLink**(string linkID)
string	**Validate**()
void	**Prepare**()
IValueSet	**GetValues**(ITime time, string linkID)
ITimeStamp	**EarliestInputTime**
void	**Finish**()
void	**Dispose**()

CSDMS

COMMUNITY SURFACE DYNAMICS MODELING SYSTEM

http://csdms.colorado.edu/

Deals with the Earth's surface - dynamic interface between lithosphere, hydrosphere, cryosphere, and atmosphere.

- Provides open-access to numerical models.
- Converts existing models into plug-and-play components.
- Can be applied for the models developed in C, C++, Fortran, Java & Python.
- Requires specific changes in the code.

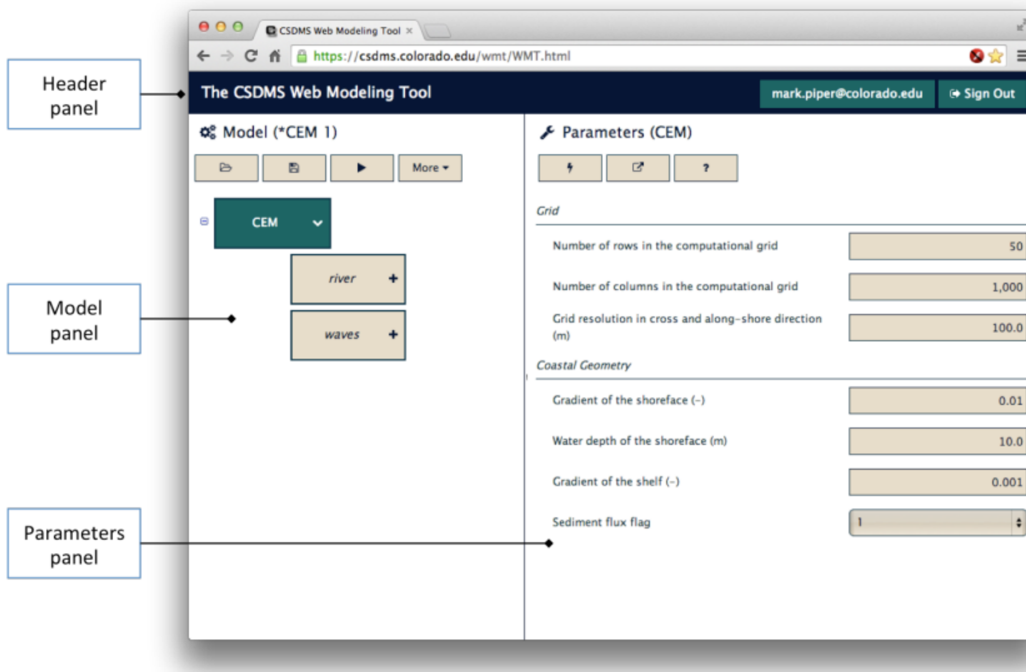


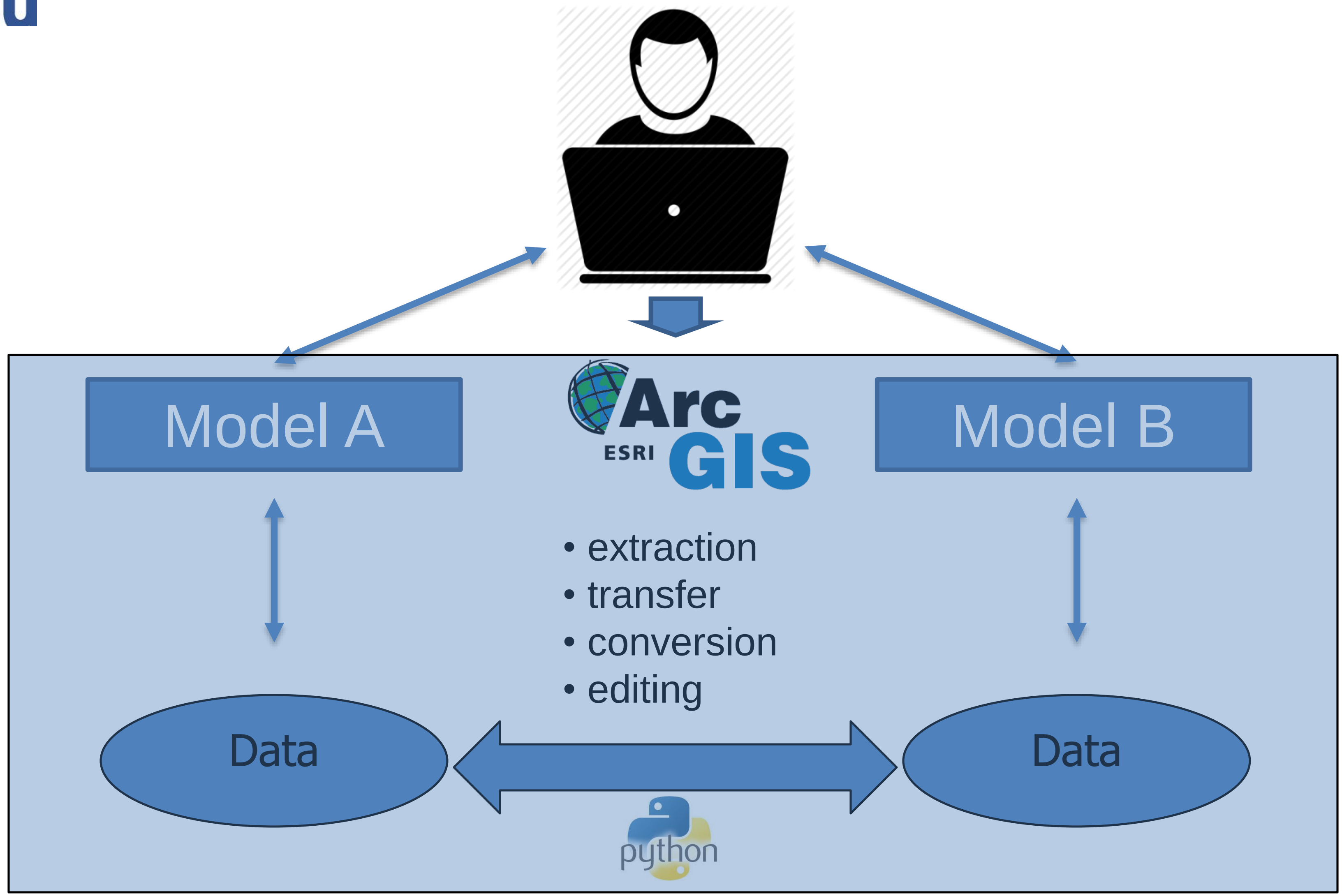
Table 1
Advantages and disadvantages of the five coupling methodologies

Methodology	Advantages	Disadvantages
One-way data transfer	Programming changes to the models unnecessary. Source code not required; suitable for proprietary models. Faster implementation with lower initial cost. Suitable for converting data between model versions.	Data conversion required between spatial and temporal scales, data file formats. Manual data editing. Quality assurance required for data conversions. New conversion procedures required when update model or system. User responsible for documenting all data transfer and conversion steps. Increased modeler, simulation time.
Loose coupling	Lower initial cost.[a] Can link models and components with minimal changes to existing code. Testing protocols address each model, not direct model interactions. Independent model development path. Supports distributed computing. Supports encapsulation for object-oriented programming (OOP).	Data conversion programs required between each set of coupled models. Conversion maintenance when data structure changes for one model. Data redundancy problems. Requires permanent data keys. Performance depends on network speed.
GUI coupling	Potentially reduced training time due to intuitive GUI.[b] Potentially easier to create input files. Potentially reduced execution time through reduced user interaction time. Supports proprietary code.[c]	Additional layer between model and user, without improving the model. Required automation of all model interactions. Model update requires GUI update. Programmers must anticipate all model applications and user needs.
Data coupling	Simpler data maintenance. Supports DBMS for consistency and easier maintenance. Reduced number of file conversion programs. Improved version control for data. Elimination of data redundancy. Supports data queries.	Potential limitations on data types. Rich language supporting geospatial and attribute data types, relationships. Overall model performance depends upon DBMS, server speed. Model interfaces depend upon DBMS.
Embedded coupling	Reduced development cost. Access to master model capabilities. Reusability for master model's code. Eliminates network communications.	Requires single computing system. Source code required for embedded model. Functionality limited to language provided by the master. Difficulty of code optimization. Changing the master may require changing embedded models. Increased computer requirements if all possible techniques and models are embedded.[d]
Integrated coupling	Promotes code reusability. Supports distributed, heterogeneous computing environments. Reduced model development cost.	Higher initial cost to facilitate integration of additional components. Network affects component performance. Shared routines required throughout the heterogeneous environment.
Tool coupling	Supports community model development. Supports both legacy and new models. Supports version control for data and code. Supports encapsulation for OOP. Supports distributed computing. Supports automated data backup. Supports DBMS with data dictionary.	Higher initial cost due to framework design and development. Relies on network and server speed. Model applications and user needs must be anticipated. Requirement for rich data language.

[a] Charnock et al. (1996).
[b] Mandel (1997).
[c] Blodgett et al. (1995).
[d] Arentze et al. (1996).

Source: Brandmeyer, J. E. & H. A. Karimi (2000). "Coupling methodologies for environmental models." Environmental Modelling & Software ***15****(5): 479-488.*

Loose Coupling with Shared User Interface



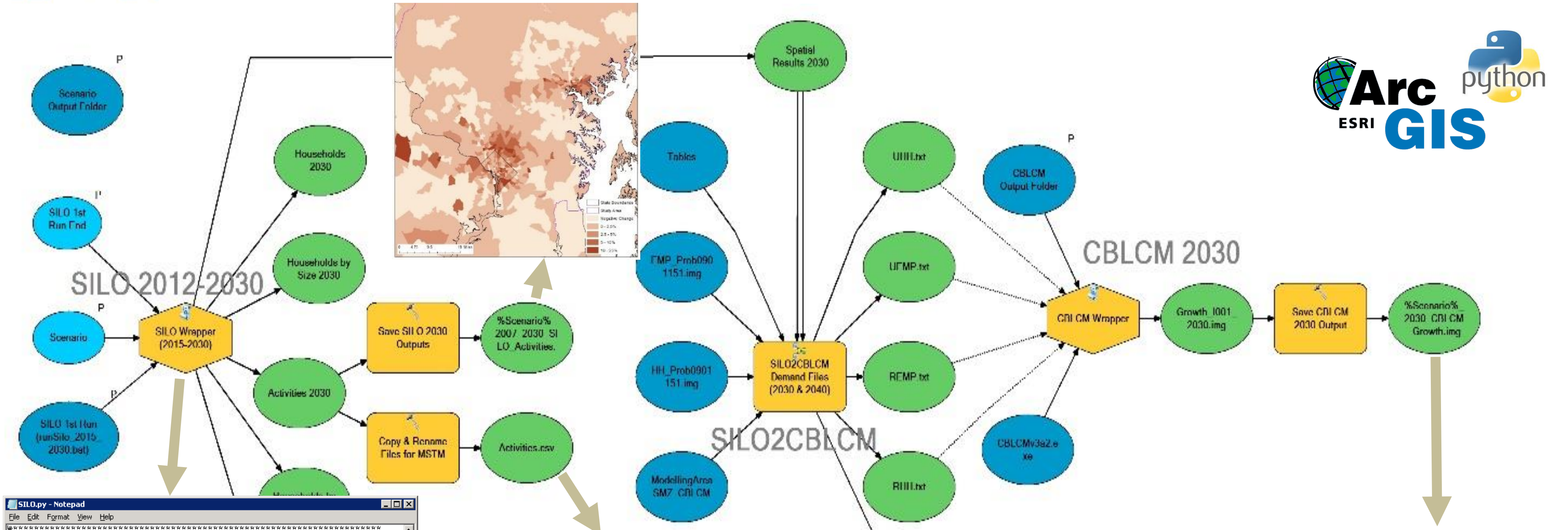
Python for Model Integration

- Specific libraries:
  - scientific programming (SciPy),
  - modeling and data analysis (Pandas),
  - visualizations and parallel computing (IPython)

- Language interoperability - often used to glue other programming languages:
  - MatLab (MLabWrap), R (RPy), Excel (OpenPyxl), FORTRAN (F2PY, PyFort), Delphi (Python4Delphi), Java (Jyton, JPype, Jepp), Perl (PyPerl), PHP (PiP), C/C++ (Ctypes, Cython, SWIG)

- Runs natively on Windows, Mac and Linux.

Python Wrappers & ArcGIS Model Builder



SILO.py - Notepad

File Edit Format View Help

```
#*********************************************************************
# Description:
# Run SILO (model)
#
#
# Arguments:
#  0 - Model file
#  1 - Data exchange folder
#  2 - Shared file
#  3 - Exported shared file
#
# Created by: Harutyun Shahumyan
#*********************************************************************

# Standard error handling
try:

    import arcpy
    import shutil
    import time
    import os
    import string

    start = time.time()
    arcpy.AddMessage("")
    arcpy.AddMessage("SILO model start time: %s" % time.strftime('%X %x %Z'


    # Get input arguments
    in_Program = arcpy.GetParameterAsText(0)
    in_SILOPropertyFile = arcpy.GetParameterAsText(1)
  #  in_ExchangeFolder = arcpy.GetParameterAsText(1)


    # Check that the program exist
    if not arcpy.Exists(in_Program):
       raise Exception, "Input program does not exist"

    # Run the model / program
    arcpy.AddMessage("")
    arcpy.AddMessage("Running %s" % (in_Program))

    import subprocess
    #subprocess.call([in_Program])
    #subprocess.call(['java', '-Xmx40000m', '-classpath C:/models/silo/mstm

    #propFile=string.replace(in_SILOPropertyFile, '\\', '/')
    #progFile=string.replace(in_Program, '\\', '/')
    #arcpy.AddMessage(propFile)
    #runjava = "java -Xmx40000m -classpath "+in_Program+";C:\models\silo\ms
    #os.system(runjava)
    #arcpy.AddMessage(runjava)

    siloClassPath="C:\models\silo\mstm;C:\models\silo\mstm\javaFiles\siloMs
    os.system("java -Xmx40000m -classpath "+siloClassPath+" com.pb.siloMstm
```

Activities.csv - Excel

FILE HOME INSERT PAGE L FORMU DATA REVIEW VIEW ACROB Harutyun...

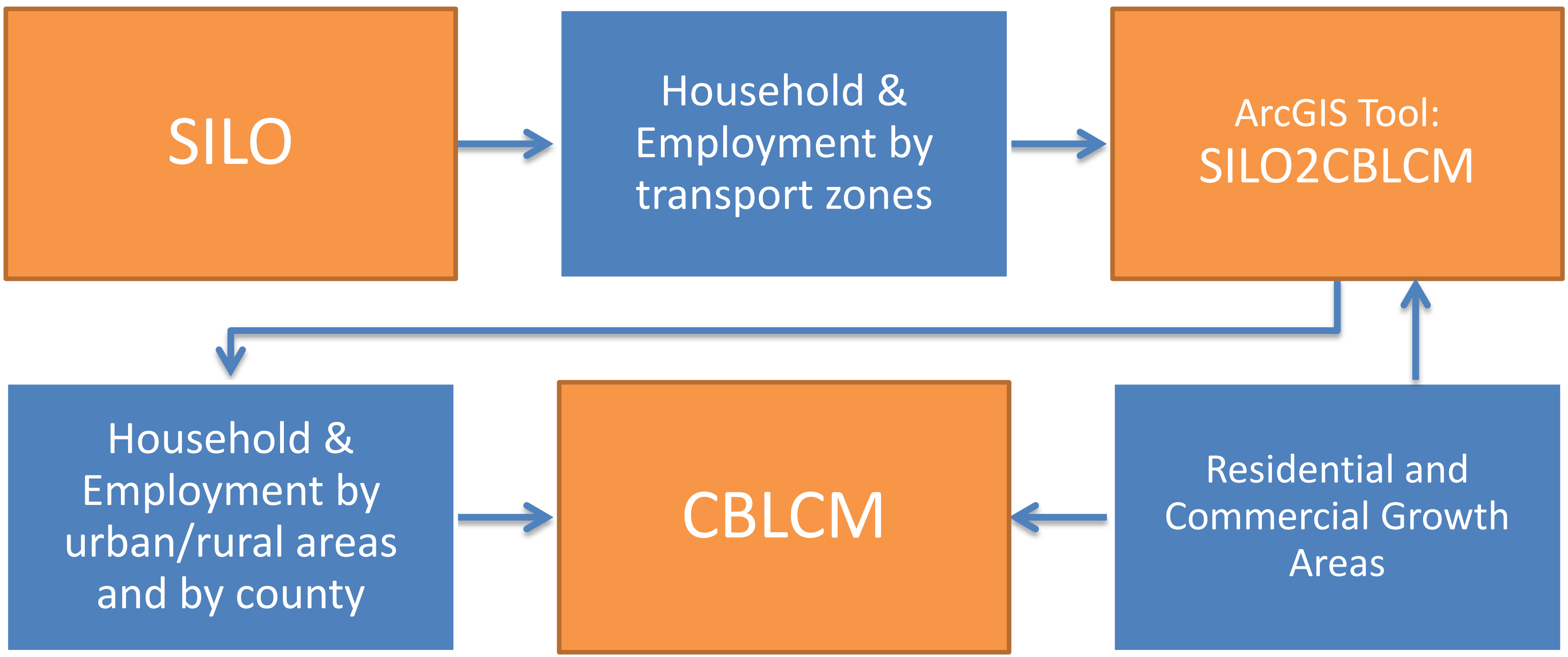
A1 ;SMZ_N

	A	B	C	D	E	F	G	H
1	;SMZ_N	ACRES	HH2007	ENR	RE2007	OFF2007	OTH2007	TOT2007
2	1	1603.51	10222	1706	373	999	829	2326
3	2	907.2264	1885	1062	525	998	1005	2626
4	3	758.8692	1049	1447	92	1200	740	2357
5	4	170.1724	637	0	96	628	390	1173
6	5	820.4843	4356	304	584	2532	3188	6533
7	6	1596.518	3552	3272	456	1239	1154	2998
8	7	421.9648	1651	913	420	431	251	1154
9	8	472.3039	3814	1247	103	173	145	447
10	9	553.1476	2819	882	74	1015	1275	2425
11	10	490.9006	3561	724	457	292	211	1009
12	11	715.3842	5897	1418	184	831	843	1969
13	12	763.0019	2502	5198	177	715	710	1703
14	13	447.3607	3013	1373	26	698	1052	1817

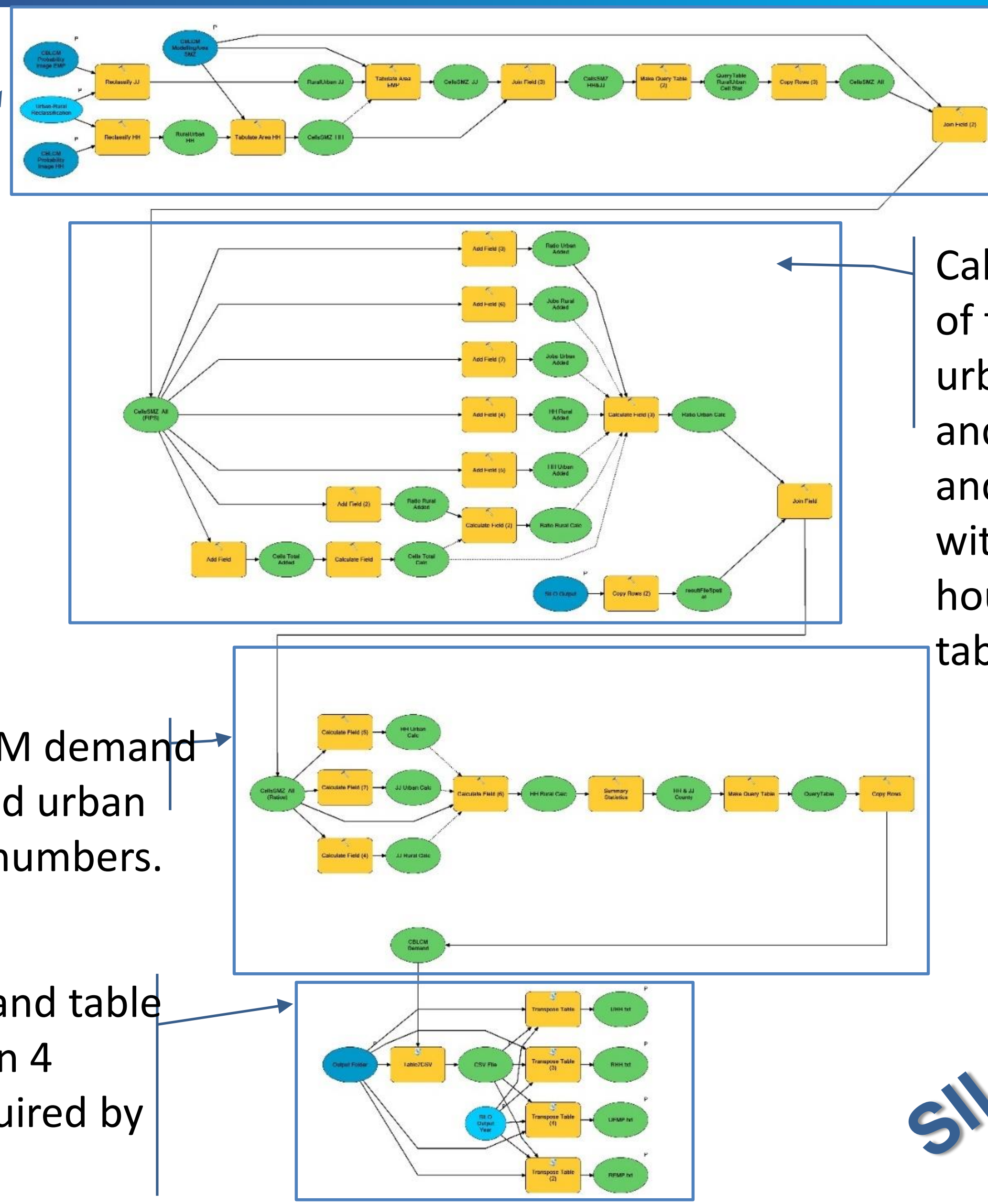
Activities

READY 100%

Organizing Data Flow



Adapting SILO Output for CBLCM



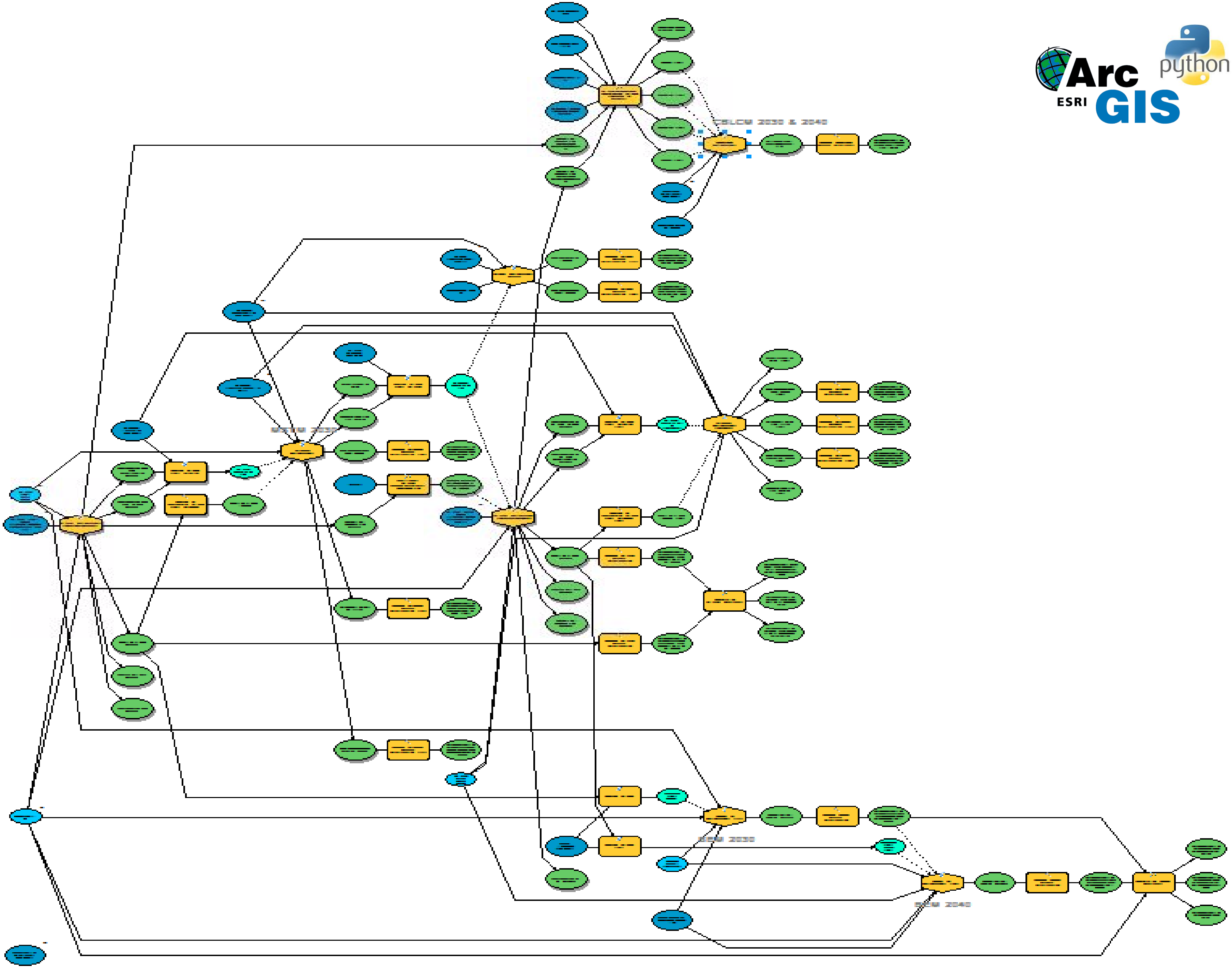
Counts commercial (JJ) and residential (HH) cells in rural and urban areas based on CBLCM growth images.

Calculates the ratio of the rural and urban residential and commercial cells and joins the table with SILO households and jobs table

Calculates the CBLCM demand table as the rural and urban job and household numbers.

Exports the CBLCM demand table as a csv file and saves it in 4 separate text files as required by CBLCM

SILO2CBLCM



Python Wrappers

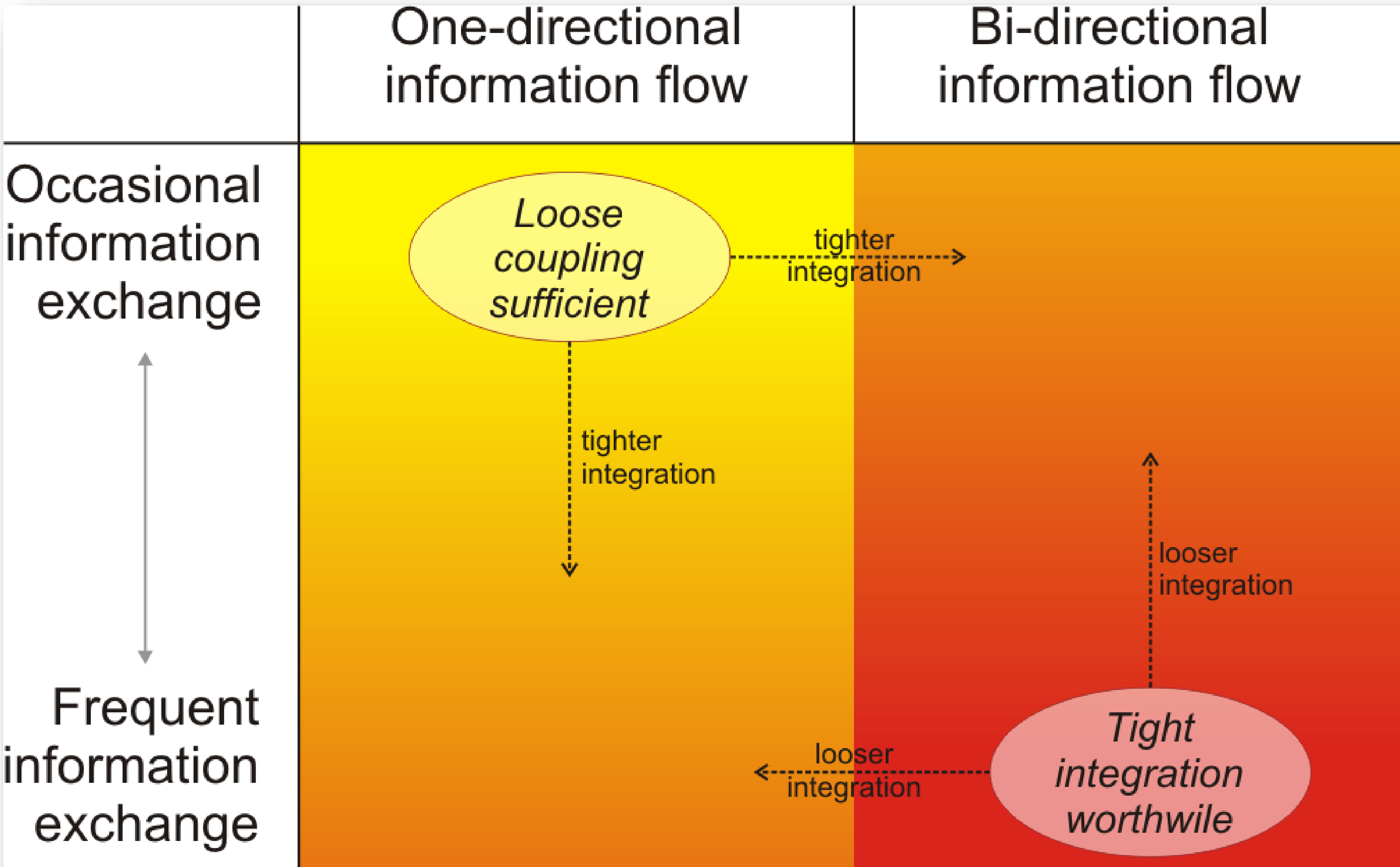
Benefits

- No need to change the source codes of the models.
- Runs models developed in different environments.
- Can be extended with additional models over time.
- General user interface showing process flow.
- Rich visualisation & mapping capabilities with ArcGIS.
- Easy to implement.

Limitations

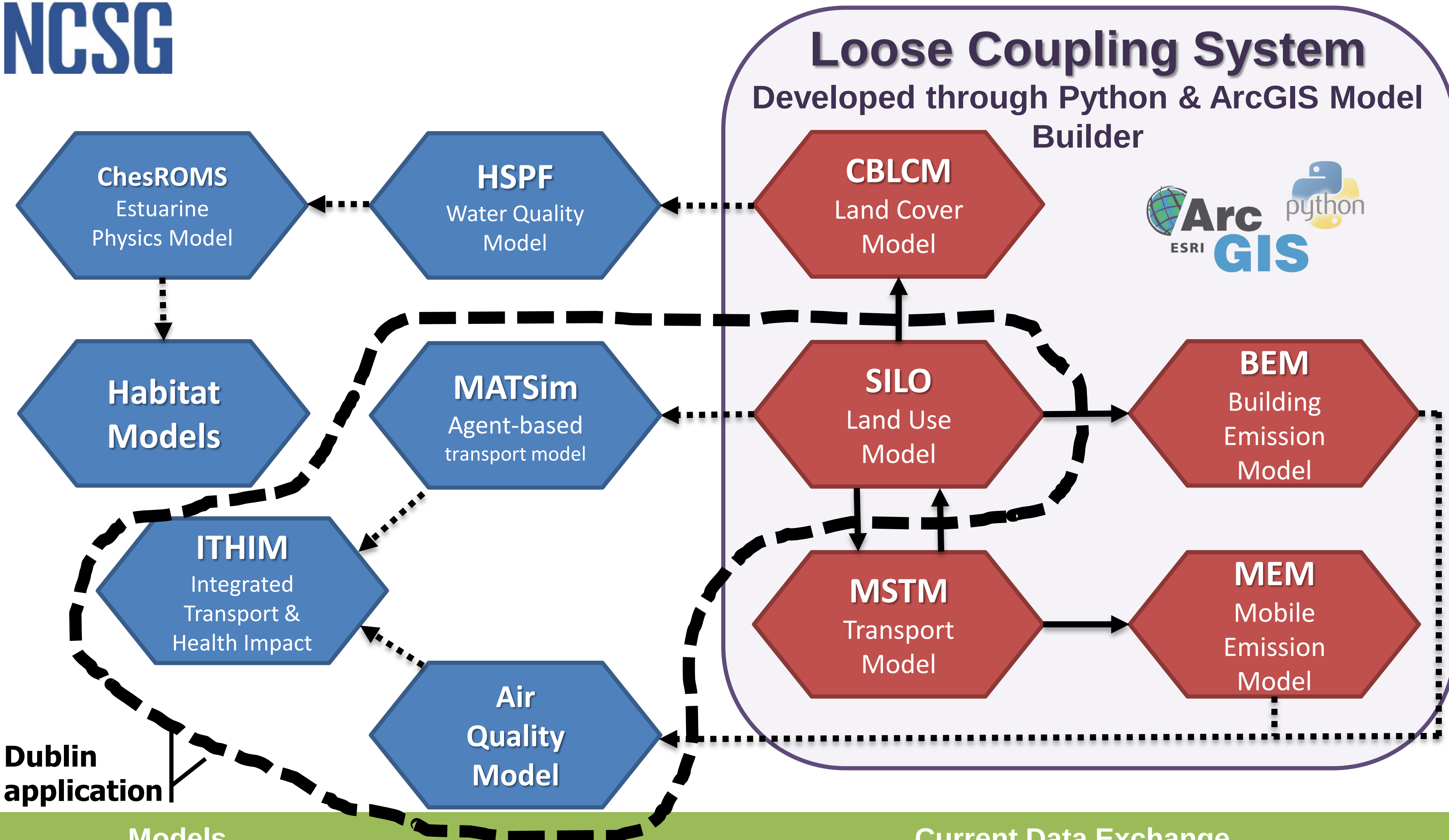
- Parallel model runs and dynamic data exchange during simulation time steps are not supported.
- Model processes run independently from one another.
- Data exchanged between modules are written to and read from a hard drive. No in-memory data exchange.

Reasons for loose coupling and tight integration



Courtesy of Dr Rolf Moeckel

Status & Potential Enhancement



Models	Current Data Exchange
Installed and coupled on the SESYNC server	Automatic data transfer (loose coupling)
Installed on different remote servers or PCs	Potential links

Environment and Planning B
Planning and Design

Impact Factor: 1.195 | **Ranking:** Environmental Studies 69 out of 104 | **5-Year Impact Factor:** 1.582 | **5-Year Ranking:** Environmental Studies 63 out of 104

Integration of land use, land cover, transportation, and environmental impact models: Expanding scenario analysis with multiple modules

Harutyun Shahumyan⇑

University of Maryland, USA

Harutyun Shahumyan, University of Maryland, 1112J Preinkert Field House, College Park, MD 20742, USA. Email: shahumyan@hotmail.com

Rolf Moeckel

Technical University Munich, Germany

Abstract

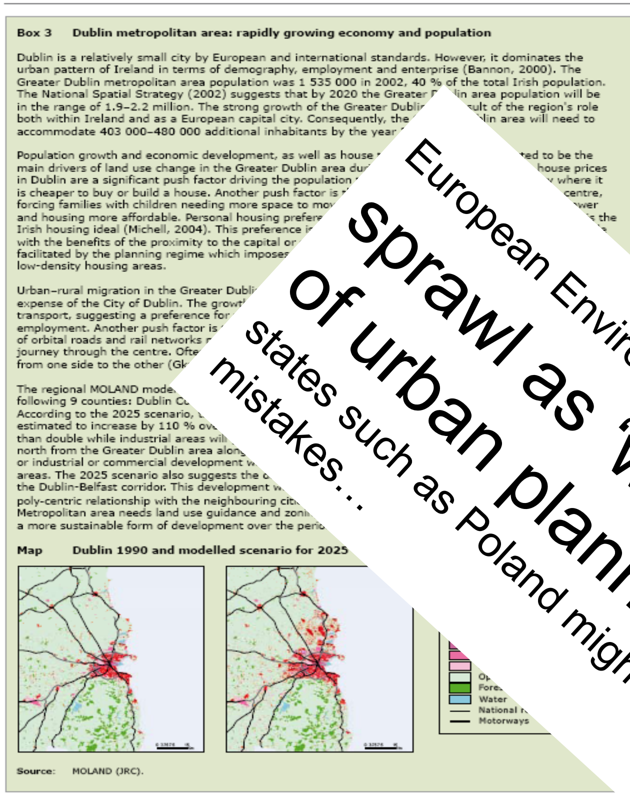
It is an expensive and time-consuming task to develop a new model. Furthermore, a single model often cannot provide answers required for complex decision making based

This Article

Published online before print May 10, 2016, doi: 10.1177/0265813516647062

Environ Plann B Plann Des **May 10, 2016** 0265813516647062

Dublin Case Study

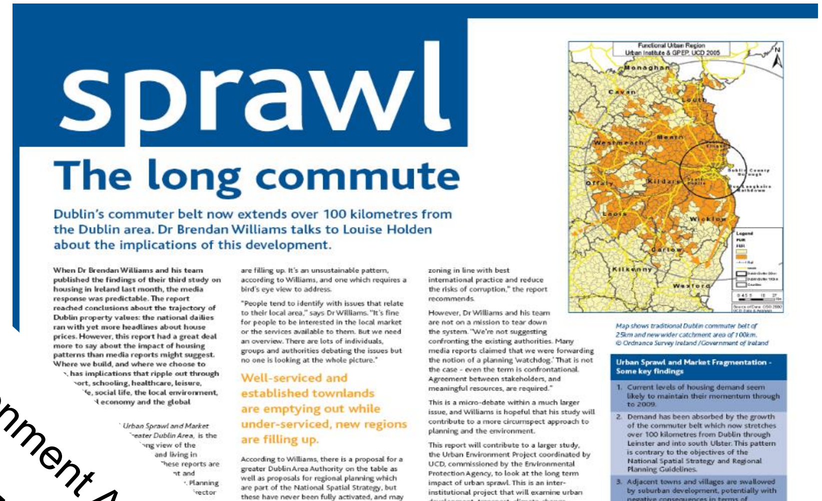

The drivers of urban sprawl

Box 3 Dublin metropolitan area: rapidly growing economy and population

Dublin is a relatively small city by European and international standards. However, it dominates the urban pattern of Ireland in terms of demography, employment and enterprise (Bannon, 2000). The Greater Dublin metropolitan area population was 1 535 000 in 2002, 40 % of the total Irish population. The National Spatial Strategy (2002) suggests that by 2020 the Greater D... lin area population will be in the range of 1.9–2.2 million. The strong growth of the Greater Dubli... ult of the region's role both within Ireland and as a European capital city. Consequently, the ... lin area will need to accommodate 403 000–480 000 additional inhabitants by the year ...

Map Dublin 1990 and modelled scenario for 2025

Source: MOLAND (JRC).

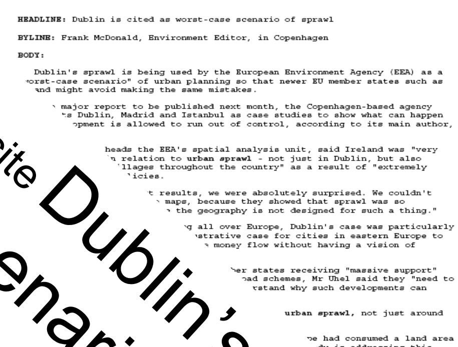

sprawl

The long commute

Dublin's commuter belt now extends over 100 kilometres from the Dublin area. Dr Brendan Williams talks to Louise Holden about the implications of this development.

Well-serviced and established townlands are emptying out while under-serviced, new regions are filling up.

Urban Sprawl and Market Fragmentation - Some key findings

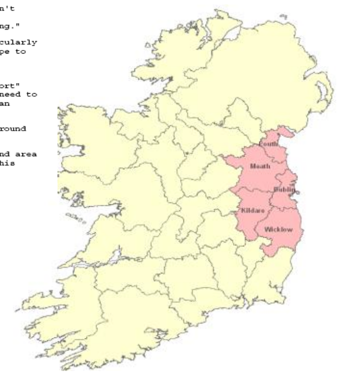

HEADLINE: Dublin is cited as worst-case scenario of sprawl

BYLINE: Frank McDonald, Environment Editor, in Copenhagen

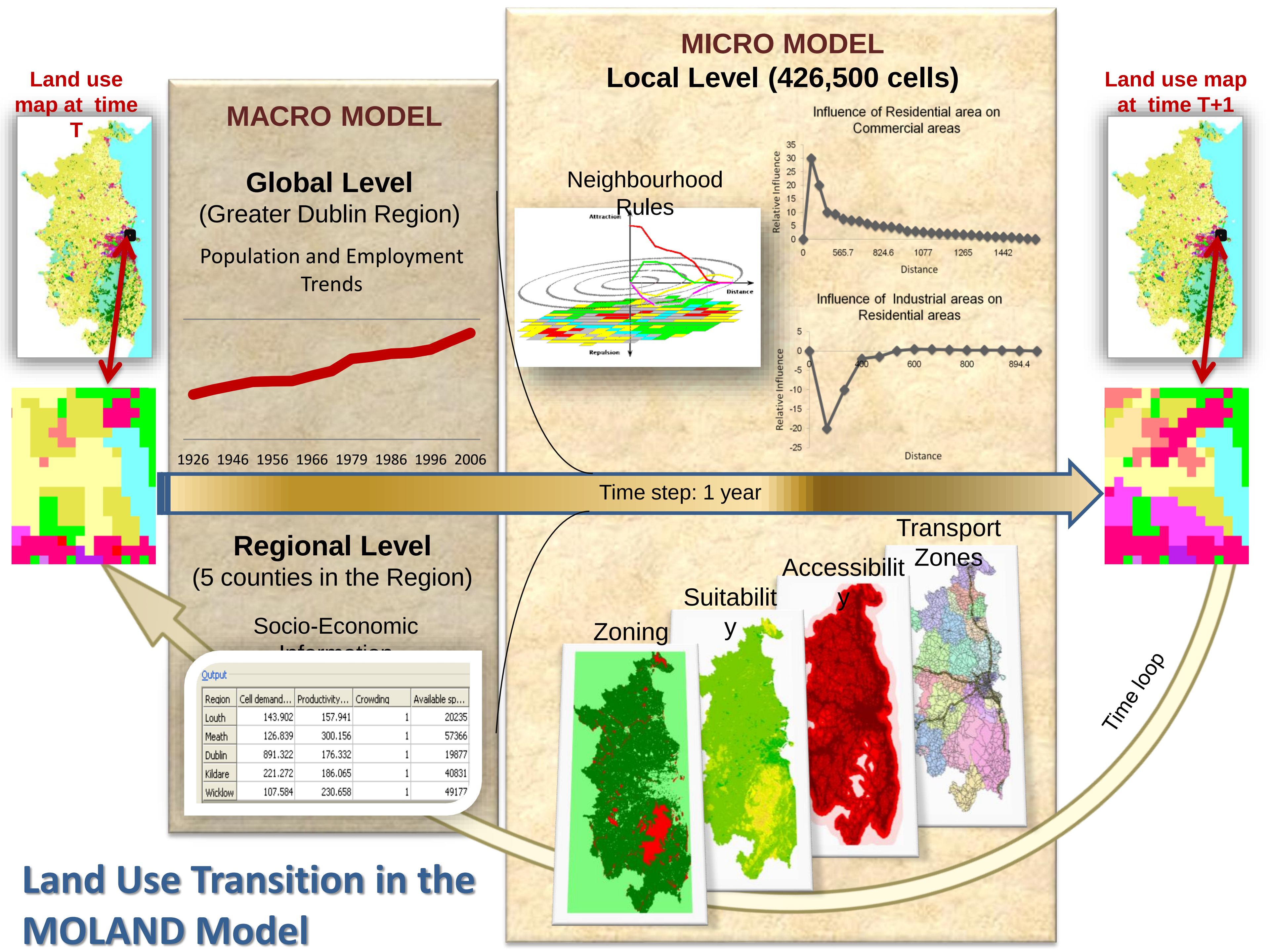
BODY:

Dublin's sprawl is being used by the European Environment Agency (EEA) as a ...orst-case scenario" of urban planning so that newer EU member states such as ...and might avoid making the same mistakes.

European Environment Agency (EEA) cite Dublin's sprawl as 'worst-case scenario' of urban planning so that newer member states such as Poland might avoid making the same mistakes…



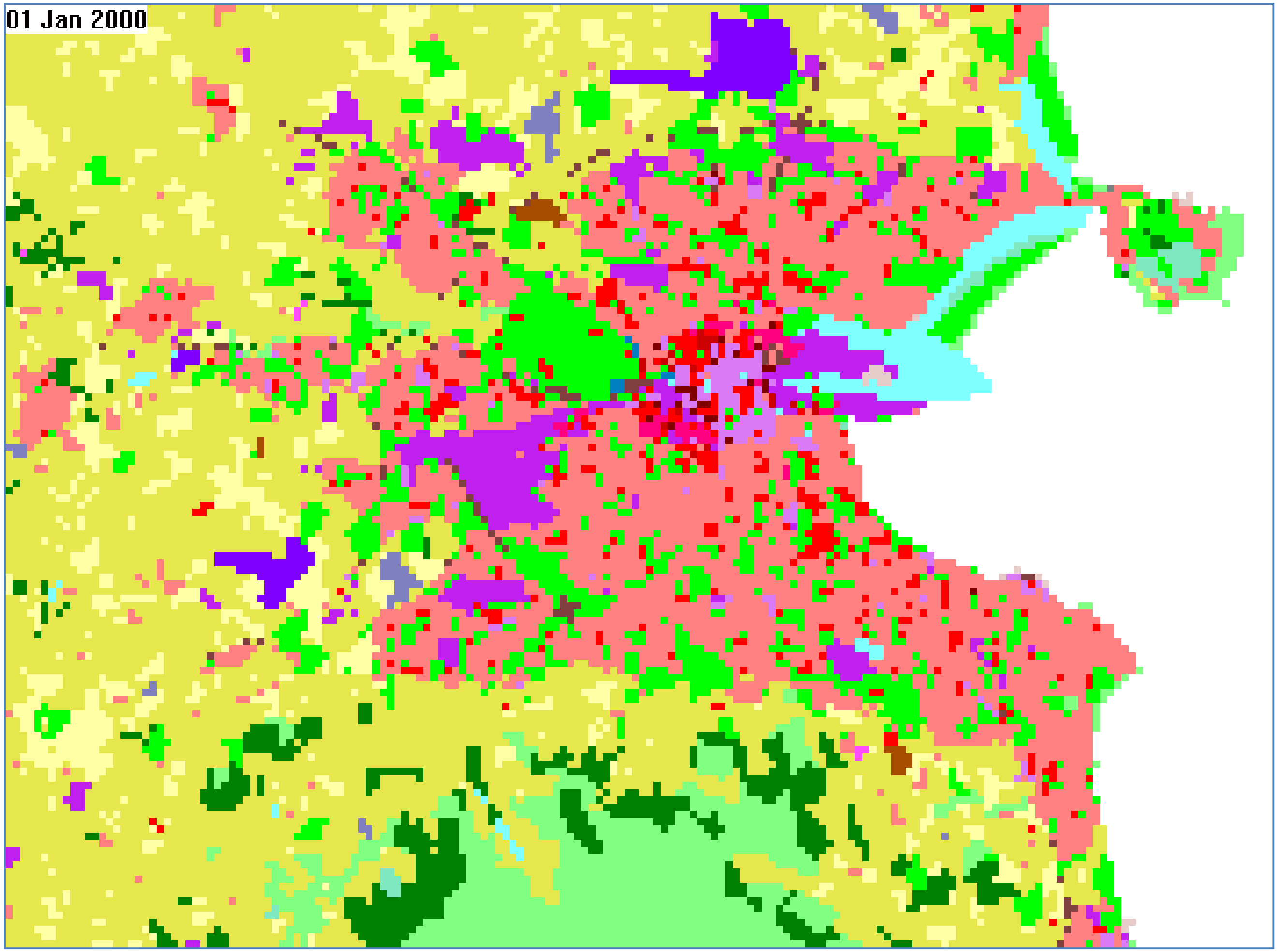
Three-quarters of all Europeans now live in urban areas and this is expected to rise to 90 per cent by 2020 based on current trends (EEA)



Region	Cell demand...	Productivity...	Crowding	Available sp...
Louth	143.902	157.941	1	20235
Meath	126.839	300.156	1	57366
Dublin	891.322	176.332	1	19877
Kildare	221.272	186.065	1	40831
Wicklow	107.584	230.658	1	49177

Sample MOLAND Simulation

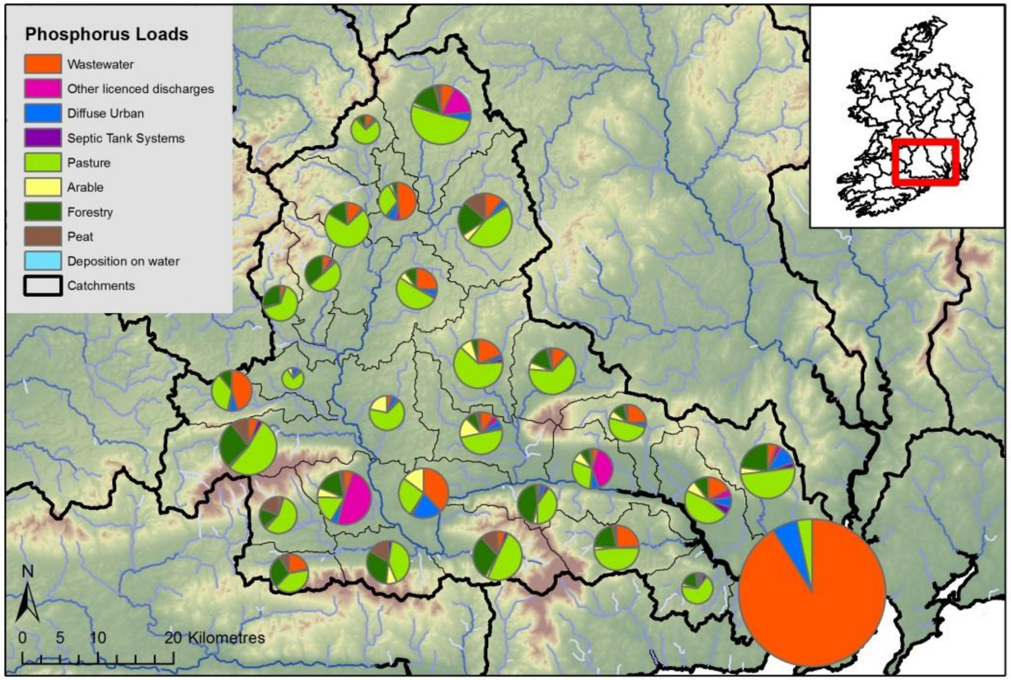
- Arable land
- Pastures
- Heterogeneous agricultural areas
- Forests
- Semi-natural areas
- Wetlands
- Abandoned
- Residential continuous dense urban fabric
- Residential continuous medium dense urban fabric
- Residential discontinuous urban fabric
- Residential discontinuous sparse urban fabric
- Industrial areas
- Commercial areas
- Public and private services
- Port areas
- Construction sites
- Road and rail networks and associated land
- Airport
- Mineral extraction sites
- Dump sites
- Artificial non-agricultural vegetated areas
- Restricted access areas
- Water bodies
- Outside area



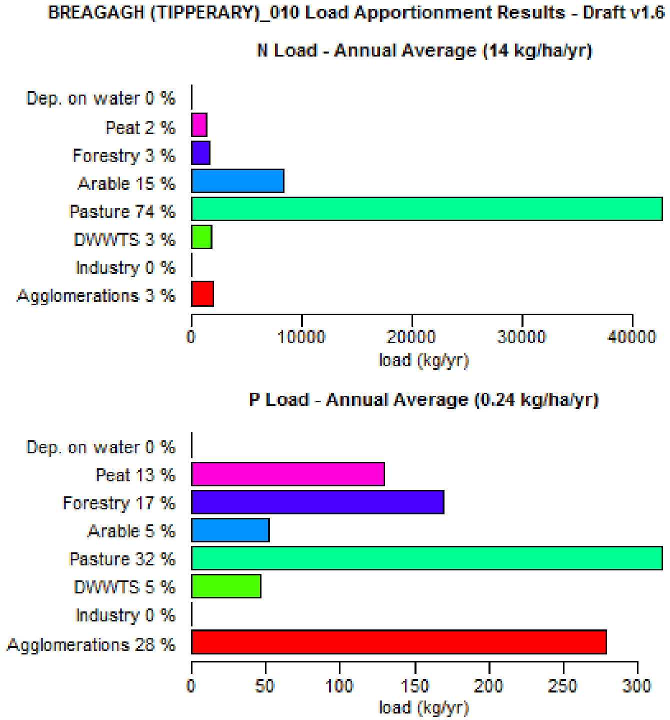
SLAM: Source Loading Apportionment Model

A source-oriented model that calculates the nitrogen & phosphorus losses to surface water from each sector in a catchment using monitoring data where available and GIS datasets.

- Purpose: To rank the sources (e.g. Agriculture, UWWTP) contributing to nutrient loads in a catchment.
- Output: Maps & charts showing proportion of nutrients attributed to each sector.

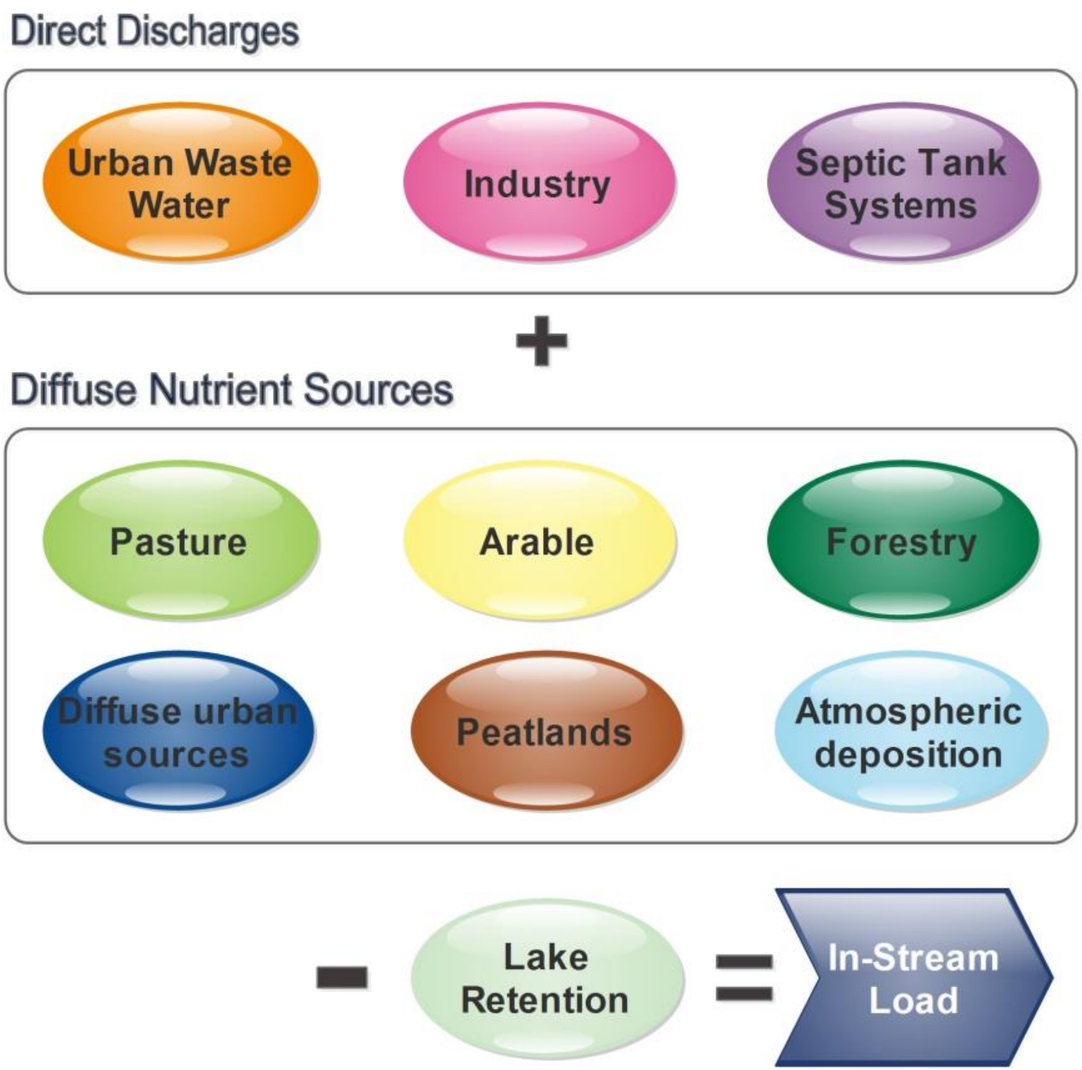


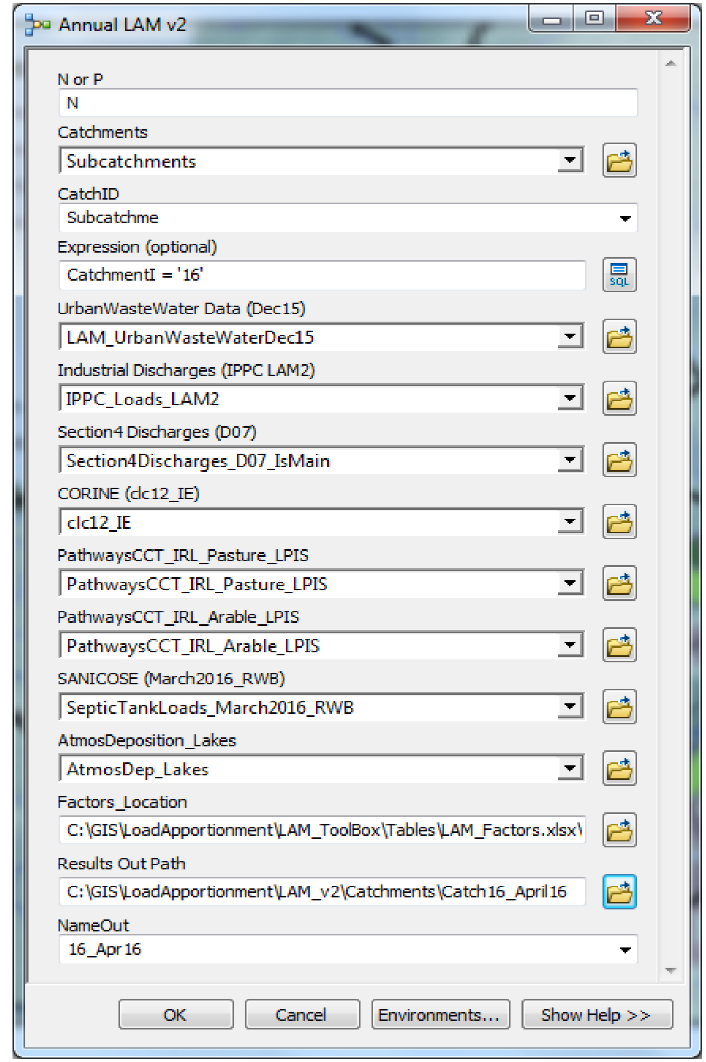
SLAM output example for Suir



SLAM output for Ara sub-catchment

SLAM Sub-Models

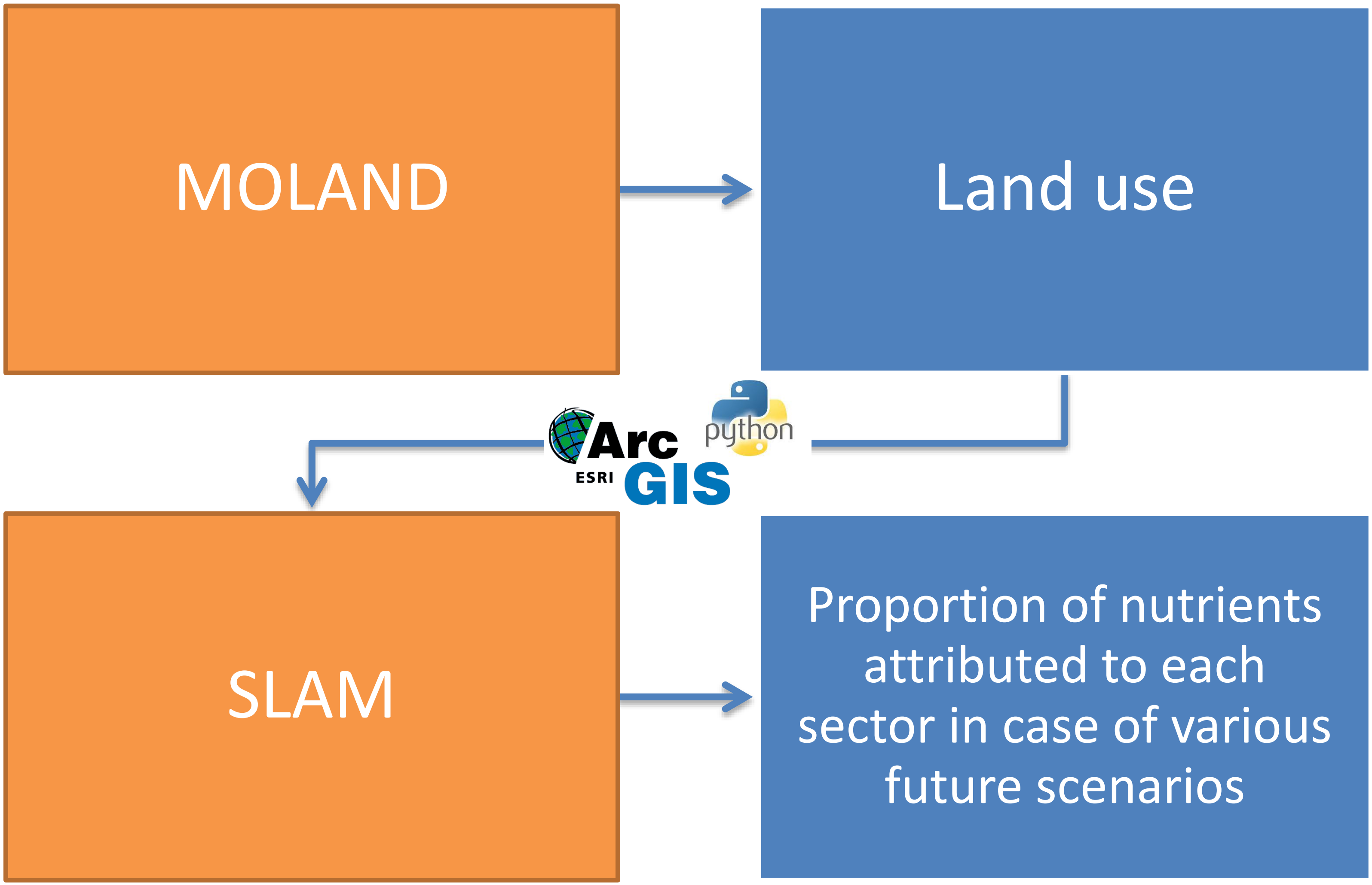




Input screen for SLAM ArcGIS toolbox.

Courtesy of Dr. Eva Mockler

Coupling MOLAND and SLAM



Thanks

harutyun.shahumyan@ucd.ie
www.shahumyan.org

Acknowledgement

This research was supported by a Marie Curie International Fellowship within the 7th European Community Framework Program and by the National Science Foundation program SESYNC. The research presented here benefited from many discussions with Uri Avin, Dan Engelbert, Sevgi Erdogan, Gerrit Knaap, Eva Mockler.