Building and validating modular urban transportation models using scientific workflow systems

J. Raimbault[1,2,3] and M. Batty[1]

*j.raimbault@ucl.ac.uk

[1]Center for Advanced Spatial Analysis, University College London
[2]UPS CNRS 3611 Complex Systems Institute Paris
[3]UMR CNRS 8504 Géographie-cités

Applied Urban Modeling 2021
Session 8: Modelling method (2)
January 28th, 2021

Urban transportation models

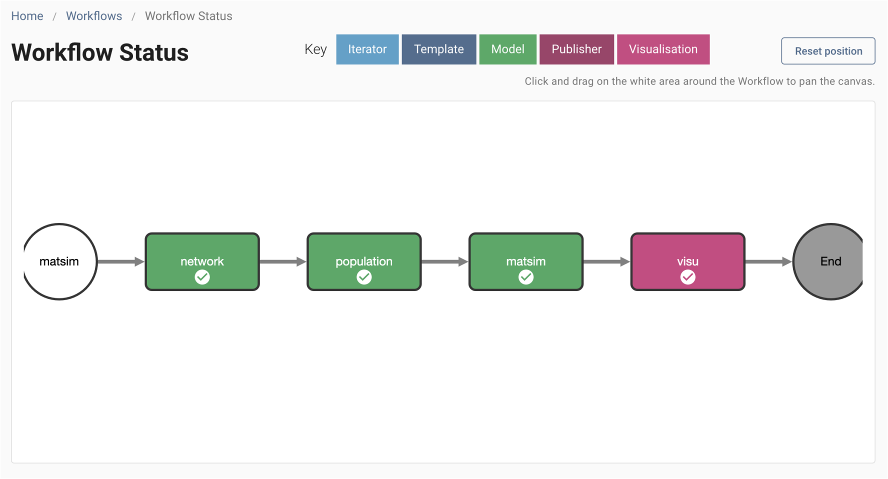
Towards modular models using workflow systems

Integrated models

Model coupling structure

DAFNI facility

DAFNI workflow for coupled model

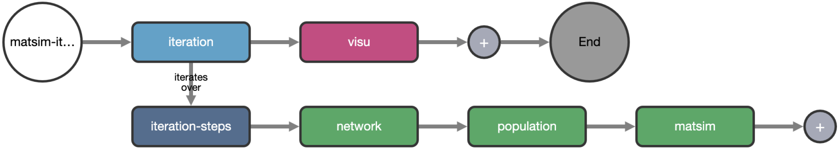


Home / Workflows / Workflow Status

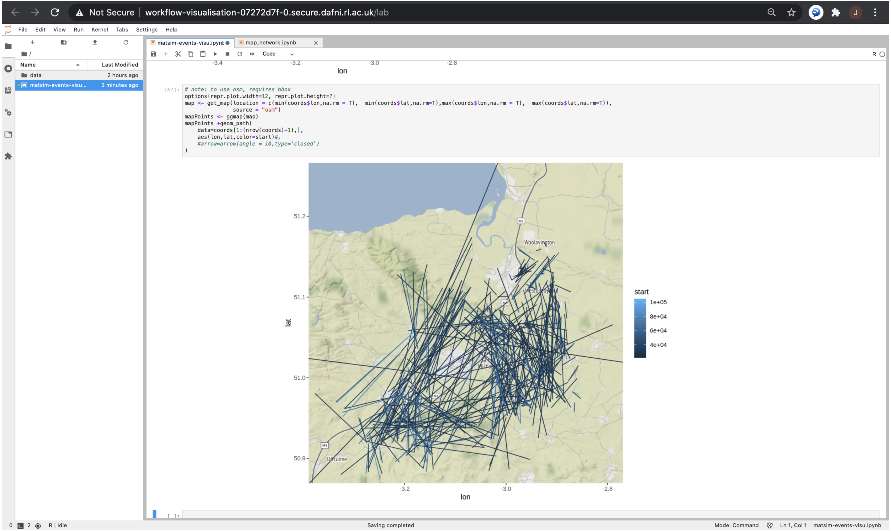
Workflow Status

Click and drag on the white area around the Workflow to pan the canvas.

Monte Carlo experiments



Visualization within DAFNI



```
[67]: # note: to use osm, requires bbox
options(repr.plot.width=12, repr.plot.height=7)
map <- get_map(location = c(min(coords$lon,na.rm = T),  min(coords$lat,na.rm=T),max(coords$lon,na.rm = T),  max(coords$lat,na.rm=T)),
                   source = "osm")
mapPoints <- ggmap(map)
mapPoints +geom_path(
    data=coords[1:(nrow(coords)-1),],
    aes(lon,lat,color=start)#,
    #arrow=arrow(angle = 10,type='closed')
)
```

Simulation results: travel distances

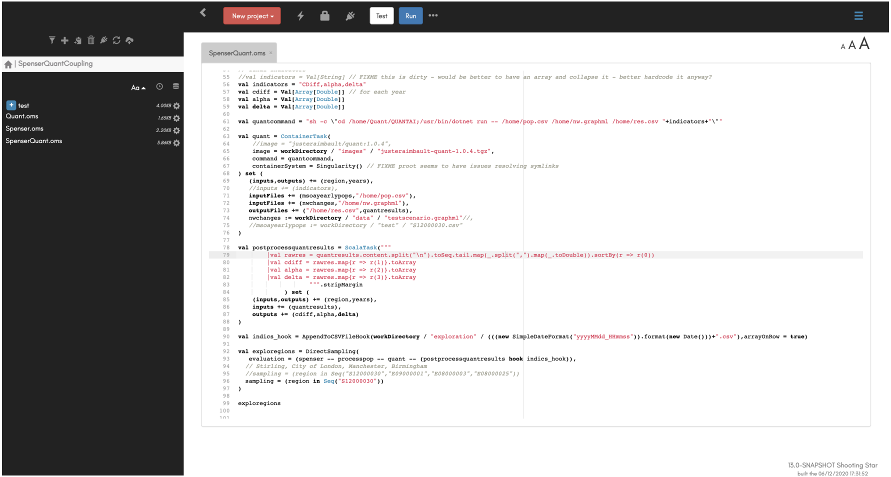
Travel patterns

Role of stochasticity

Validation: towards spatial sensitivity analysis

OpenMOLE workflow engine

Coupling SPENSER and QUANT with OpenMOLE



```
//val indicators = Val[String] // FIXME this is dirty - would be better to have an array and collapse it - better hardcode it anyway?
val indicators = "CDiff,alpha,delta"
val cdiff = Val[Array[Double]] // for each year
val alpha = Val[Array[Double]]
val delta = Val[Array[Double]]

val quantcommand = "sh -c \"cd /home/Quant/QUANTAI;/usr/bin/dotnet run -- /home/pop.csv /home/nw.graphml /home/res.csv "+indicators+"\""

val quant = ContainerTask(
    //image = "justeraimbault/quant:1.0.4",
    image = workDirectory / "images" / "justeraimbault-quant-1.0.4.tgz",
    command = quantcommand,
    containerSystem = Singularity() // FIXME proot seems to have issues resolving symlinks
) set (
   (inputs,outputs) += (region,years),
   //inputs += (indicators),
   inputFiles += (msoayearlypops,"/home/pop.csv"),
   inputFiles += (nwchanges,"/home/nw.graphml"),
   outputFiles += ("/home/res.csv",quantresults),
   nwchanges := workDirectory / "data" / "testscenario.graphml"//,
   //msoayearlypops := workDirectory / "test" / "S12000030.csv"
)

val postprocessquantresults = ScalaTask("""
        |val rawres = quantresults.content.split("\n").toSeq.tail.map(_.split(",").map(_.toDouble)).sortBy(r => r(0))
        |val cdiff = rawres.map{r => r(1)}.toArray
        |val alpha = rawres.map{r => r(2)}.toArray
        |val delta = rawres.map{r => r(3)}.toArray
                  """.stripMargin
             ) set (
    (inputs,outputs) += (region,years),
    inputs += (quantresults),
    outputs += (cdiff,alpha,delta)
)

val indics_hook = AppendToCSVFileHook(workDirectory / "exploration" / (((new SimpleDateFormat("yyyyMMdd_HHmmss")).format(new Date()))+".csv"),arrayOnRow = true)

val exploregions = DirectSampling(
   evaluation = (spenser -- processpop -- quant -- (postprocessquantresults hook indics_hook)),
  // Stirling, City of London, Manchester, Birmingham
  //sampling = (region in Seq("S12000030","E09000001","E08000003","E08000025"))
  sampling = (region in Seq("S12000030"))
)

exploregions
```

13.0-SNAPSHOT Shooting Star
built the 06/12/2020 17:31:52

Towards advanced validation experiments

Discussion

Conclusion

→

Open repositories

`https://github.com/JusteRaimbault/UrbanDynamics` for workflows

Workflow engines

References I