Step 1: Prepare copy of NewTimeBased

1. Copy NewTimeBased with folder name microDemProj
2. Change names of solution files to microDemProj-modgen.sln and microDemProj-ompp.sln
3. Check in to Git, and test build

Step 2: Modgen version using microdata\_csv and simulation framework code

1. Copy all original .mpp source files from microDemProj (replaces PersonCore.mpp and TickerCore.mpp)
2. Remove French language files (to simplify)
3. Incorporate model-specific core functionality from microDemProj.mpp into model.mpp.
4. Add #include omc/microdata\_csv.h to custom\_early.h
5. Add global input and output csv objects
6. Remove microDemProj.mpp
7. Remove MicroInput.mpp
8. Move file open and close from MicroOutput.mpp to model.mpp
9. Create Default scenario, microdata input file in MICRODATA folder, etc.  
   Place input and output microdata files in Parameters/Default folder for now.  
   Enter scenario ‘properties’ as parameters in Framework.odat  
   Reduced pop size to 10,000 for more rapid comparison testing

Step 3: Cross-compatible version

1. Add function calls to manage entity lifecycle (in Start, Finish):  
   initialize\_attributes, enter\_simulation, exit\_simulation.
2. Change declarations ‘Person \*’ to ‘auto’ where C++ compiler gives an error like “cannot convert link<E> to Person \*”
3. Insert casts, COERCE, where C++ compiler gives error (1 location):  
   AGE\_RANGE integer\_age = COERCE(AGE\_RANGE,self\_scheduling\_int(age)))
4. Temporary do-nothing versions of Set\_actor\_weight, Set\_actor\_subsmaple\_weight
5. Temporary population scaling in ompp\_framework.ompp for the model.