dsr

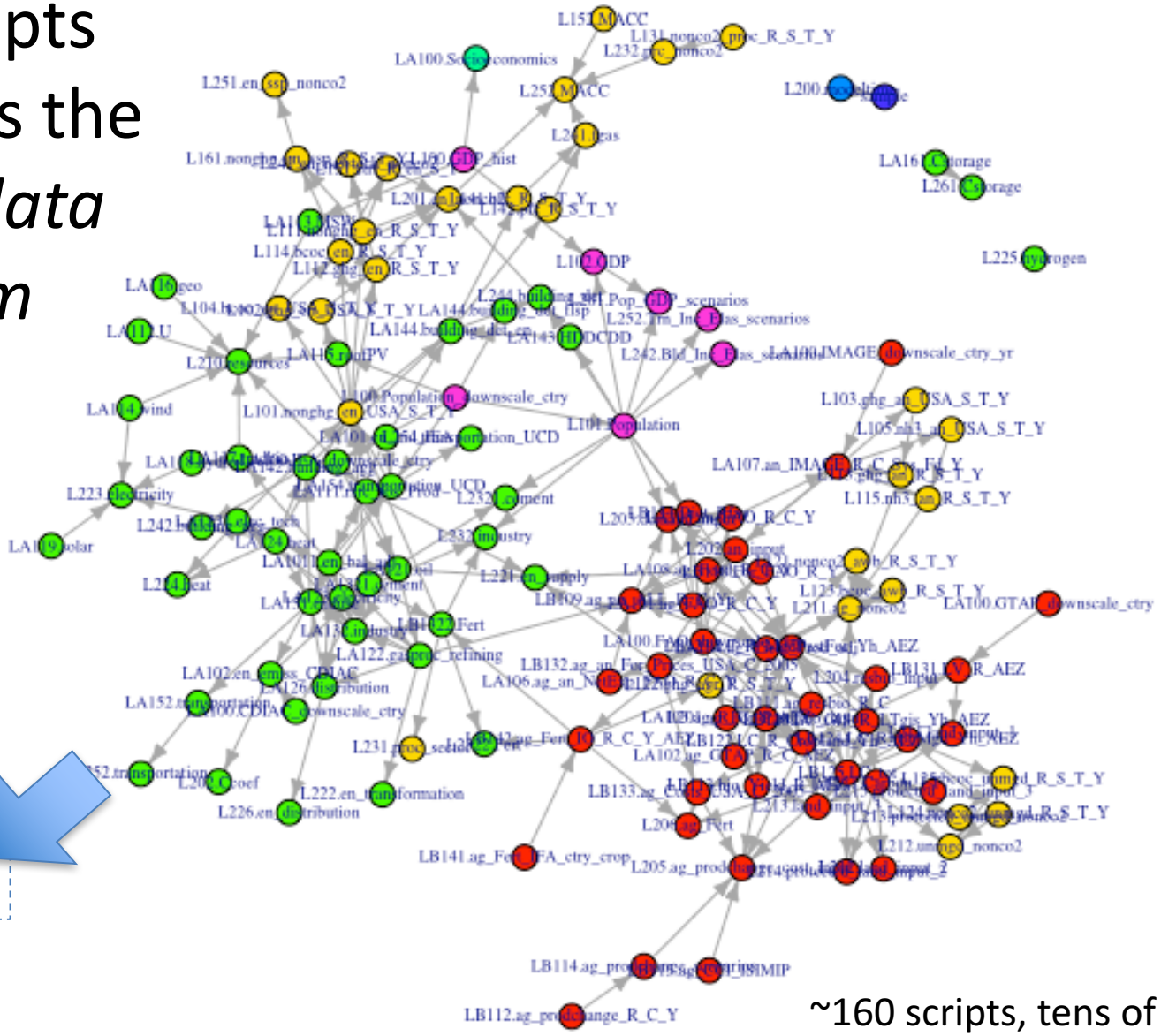
GCAM Data System Rewrite (dsr)

November 8, 2017

Data -> GCAM -> Downscaling

- GCAM is a big and complex model
- The entire structure of the model can be defined and dynamically changed by its inputs
- These inputs were once hand-constructed from Excel files but now there's a "build" step just like the model software itself

A complex system of R scripts comprises the GCAM *data system*



xml

GCAM

~160 scripts, tens of thousands of lines of code

Motivation for dsr work

- Putting the data system into R was a huge advance
  - Replicates Excel original workbooks
  - Automation, reproducibility
- Limitations of the current system
  - Not flexible in its assumptions and behavior
  - Insufficient group knowledge
  - Hard to maintain and inspect
- Ambitious science goals require robust, flexible data system to support GCAM

Goals for dsr v1

- Better documentation throughout
- Flexibility (change assumptions)
- Robustness
  - quickly know when things go wrong
- Code clarity
- Tools to diagnose, explore, modify, test
- Easy to use
- Speed

Principles

- Clear and clean code, documentation, abstracted common code, discrete functions
- R *package*: lots of good things for free
- Need known starting point
  - Identical output to current data system
  - I.e., *first* move code to more sustainable structure; then more ambitious things

New code aims for clarity and speed

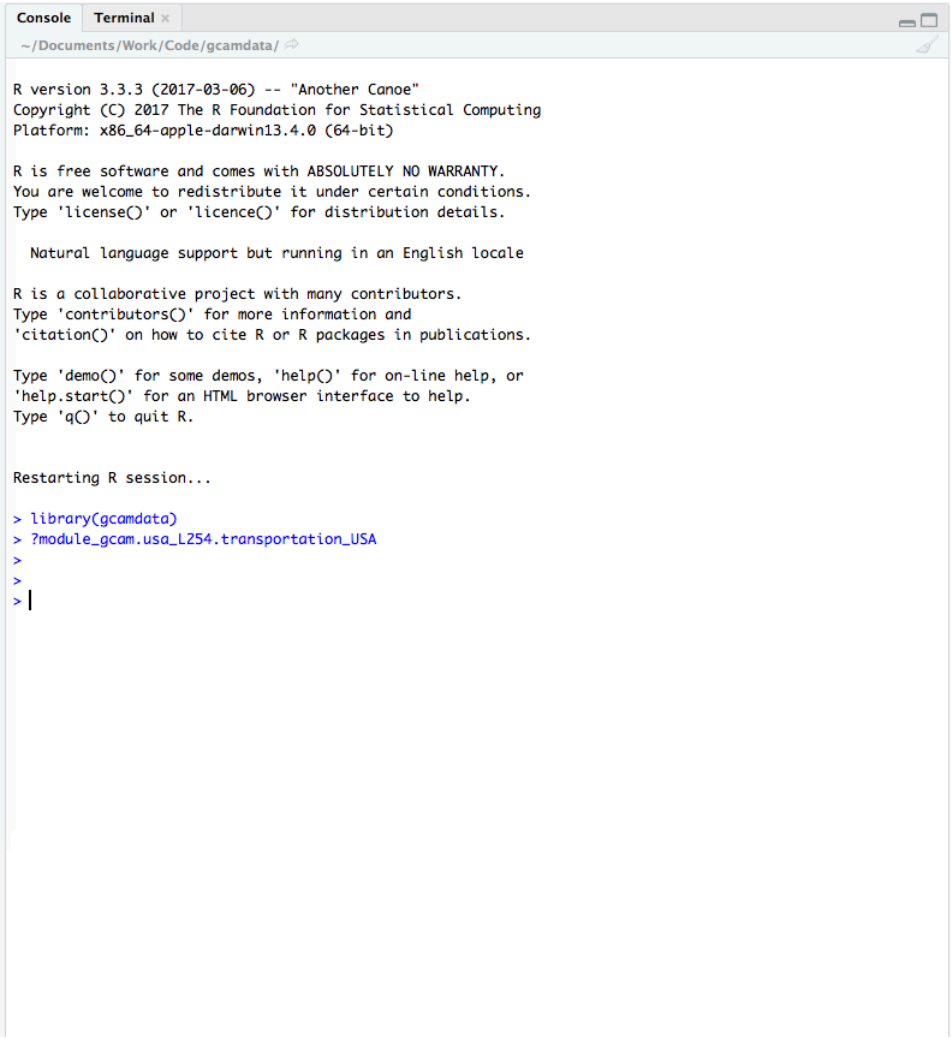
```
# Bind the '97 and '09 GDP datasets to get a continuous time series
BEA_pcGDP_97USD_state %>%
  ungroup %>%
  filter(!year %in% unique(BEA_pcGDP_09USD_state$year)) %>%
  bind_rows(BEA_pcGDP_09USD_state) %>%
  # merge with state name/codes
  left_join_error_no_match(states_subregions, by = c("Area" = "state_name")) %>%
  select(state, year, value) %>%
  # merge with census data, and compute total GDP (population * per capita GDP)
  left_join(Census_pop_hist, by = c("state", "year")) %>%
  mutate(value = value * 1e-6 * population) %>%
  arrange(state, year) %>%
  select(-population) %>%
  # compute by-state shares by year
  group_by(year) %>%
  mutate(share = value / sum(value)) %>%
  select(-value) ->
  L100.GDPshare_state
```

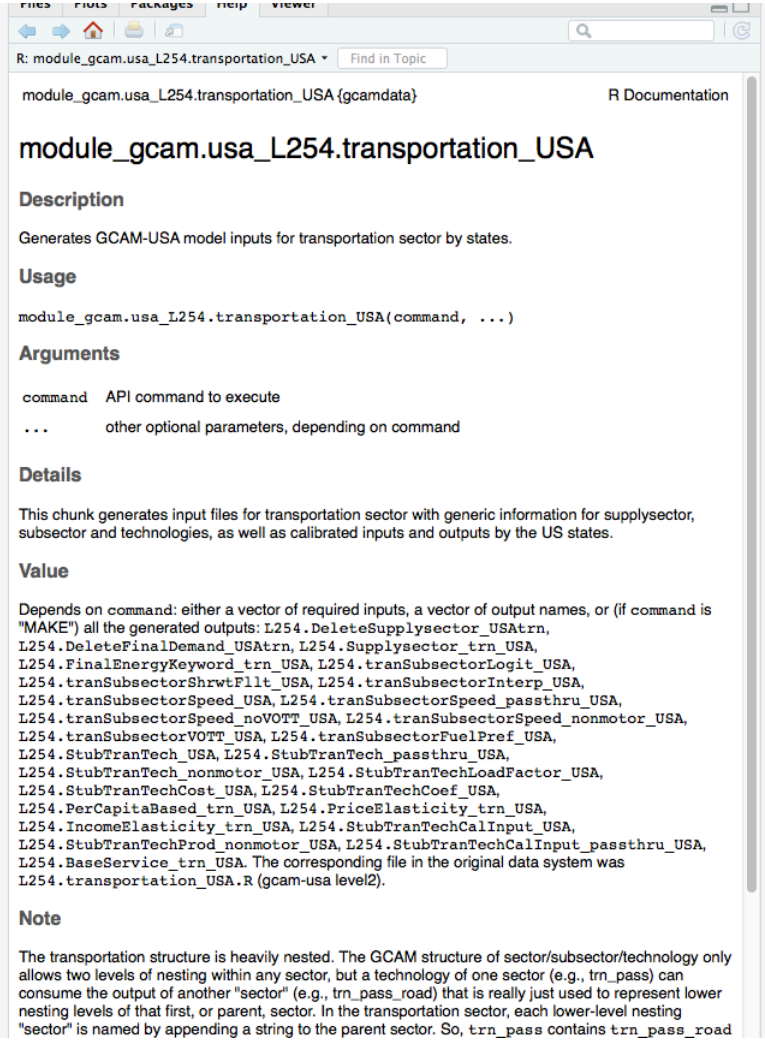
Very easy to run the new code

```
> library(gcamdata)
> driver()
GCAM Data System v0.4
Found 190 chunks
Found 1345 chunk data requirements
Found 833 chunk data products
[1]"module_aglu_L2242.land_input”
[1]"- make 0.10”
[1]"module_aglu_LA100.0_LDS_preprocessing”
[1]"- make 2.69”
[1]"module_aglu_LA100.FAO_downscale_ctry”
[1]"- make 3.81"
```

(etc.)

Integrated documentation





Testing and its benefits

- With what's called "continuous integration", i.e. continuous automated testing, we can enforce *lots* of good things
  - Correct outputs
  - Robust behavior
  - Code reviews
  - Documentation – system *will not build* with incomplete documentation

Testing and its benefits

What inputs does it need? Does it use them all?

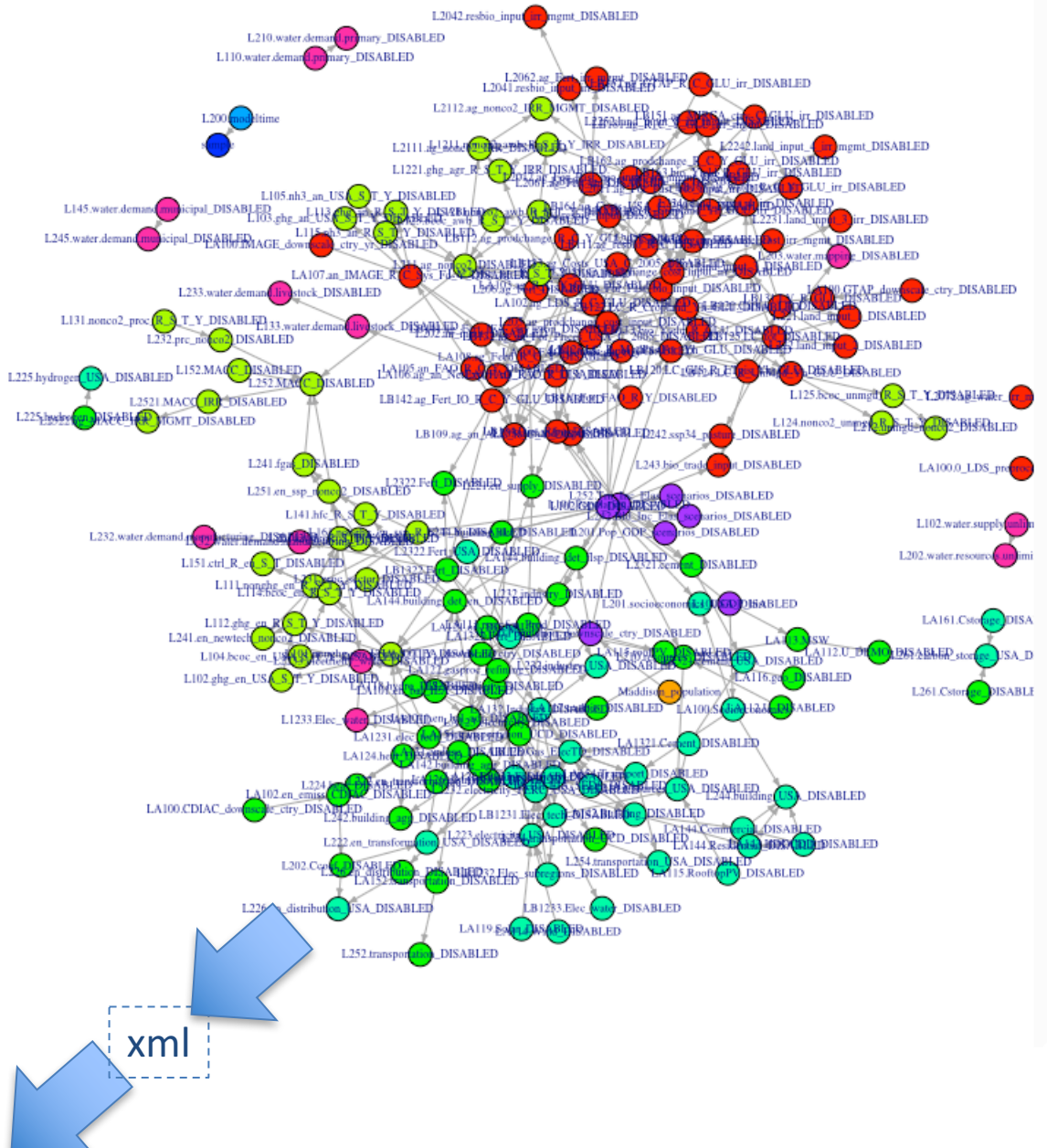
Is the code well behaved and well documented?

Chunk of code that does some particular data processing step

Does it produce promised outputs? Are they documented? Do they match old data system? What dependencies do they have?

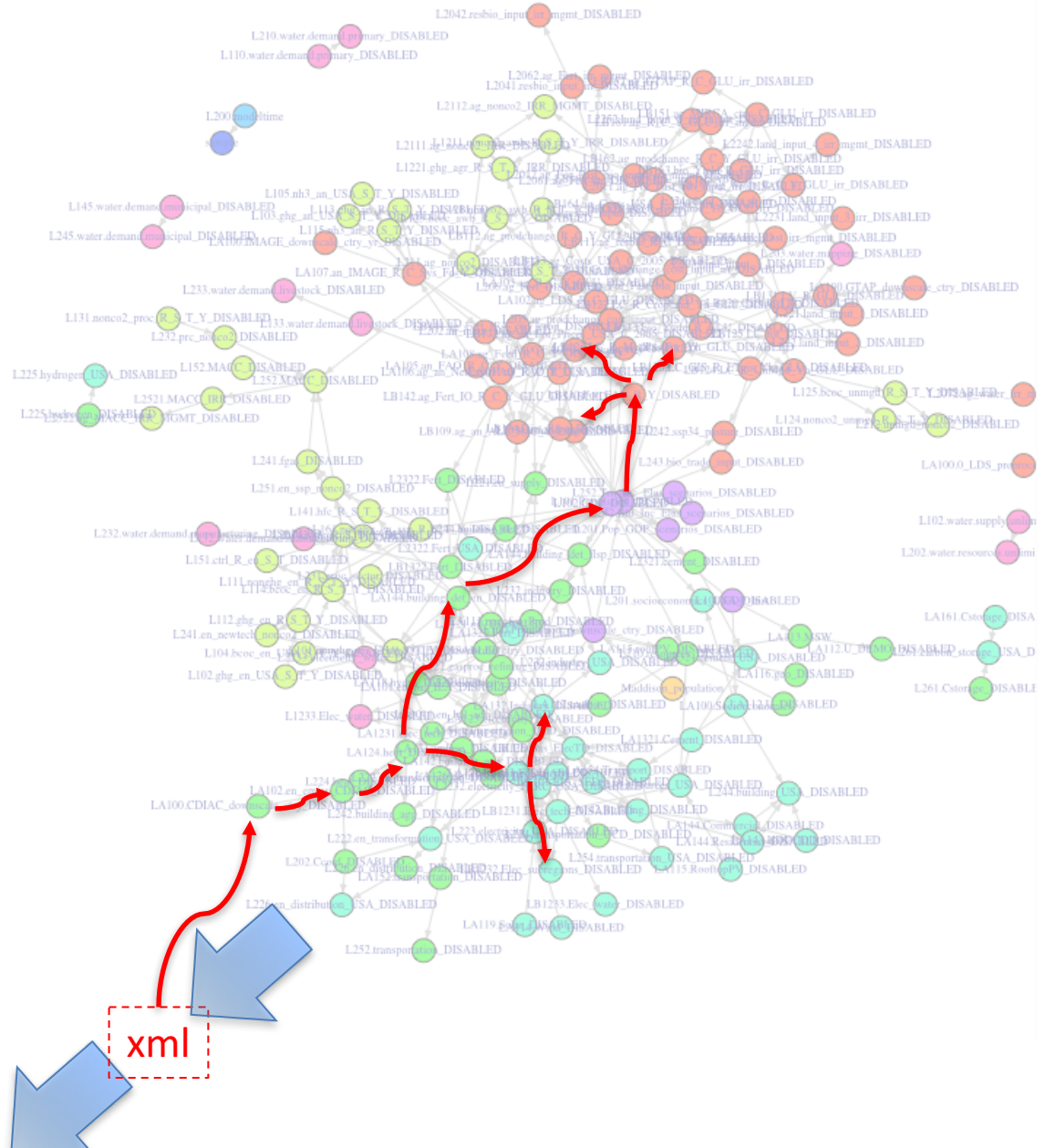
New capabilities enable better and faster research

- Because we define and enforce chunk behavior, we can do useful things:
  - Graph data flow
  - Get the provenance of any piece of data
  - Trace data and identify problems
  - 'Shim' into the data system to examine/change/test
- I.e., this can be a useful tool not just cumbersome scripts



xml

```
> trace("bld_agg_gSSP1.xml")
1 - bld_agg_gSSP1.xml
        produced by module_socio_batch_bld_agg_
        Precursor: L242.IncomeElasticity_bld_gS
2 - L242.IncomeElasticity_bld_gSSP1
        produced by module_socioeconomics_L242.
        Building Income Elasticity: gSSP1 (Unit
        Uses previously calculated per-capita G
        Building income elasticity for each GCA
        Precursor: common/GCAM_region_names (#3
        Precursor: socioeconomics/A42.inc_elas
        Precursor: L102.pcgdp_thous90USD_Scen_R
3 - common/GCAM_region_names - read from file
        GCAM 32-region names (NA)
        Maps GCAM region IDs to region names
        Read from extdata/common/GCAM_region_na
        No precursors
4 - socioeconomics/A42.inc_elas - read from fi
        Building sector income elasticity, (pcg
inc_elas:unitless (% change in service demand ,
        aggregate buildings sector income elast
        Read from extdata/socioeconomics/A42.in
        No precursors
5 - L102.pcgdp_thous90USD_Scen_R_Y - produced
        Gross Domestic Product (GDP) per capita
of 1990 USD (MER))
        Computed as GDP/population.  Values pri
        historical; values subsequent are from
        Precursor: common/iso_GCAM_regID (#6 be
        Precursor: socioeconomics/SSP_database_
        Precursor: socioeconomics/IMF_GDP_growt
        Precursor: L100.gdp_mil90usd_ctry_Yh (#
        Precursor: L101.Pop_thous_R_Yh (#10 bel
        Precursor: L101.Pop_thous_Scen_R_Yfut (
6 - common/iso_GCAM_regID - read from file
        ISO to GCAM region mapping (NA)
        Maps iso codes to GCAM regions (includi
        ----------,,Former GCAM regions,
        Read from extdata/common/iso_GCAM_regID
        No precursors
```



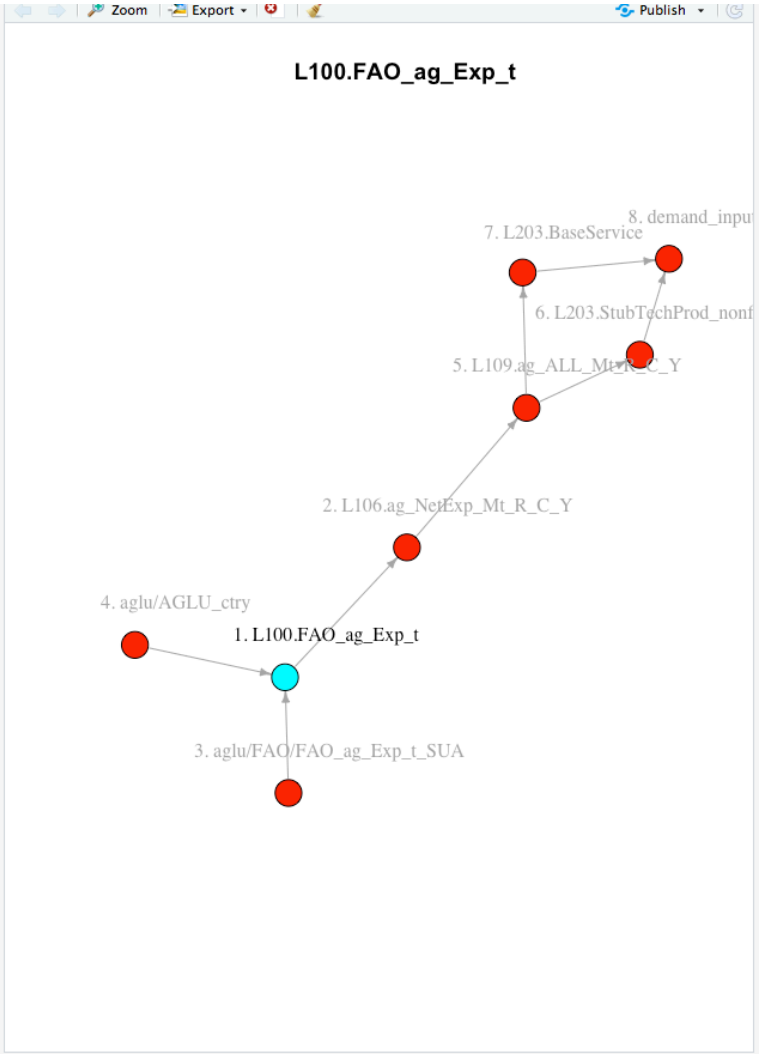

```
> trace("bld_agg_gSSP1.xml")
1 - bld_agg_gSSP1.xml
        produced by module_socio_batch_bld_agg_
        Precursor: L242.IncomeElasticity_bld_gS
2 - L242.IncomeElasticity_bld_gSSP1
        produced by module_socioeconomics_L242.
        Building Income Elasticity: gSSP1 (Unit
        Uses previously calculated per-capita G
        Building income elasticity for each GCA
        Precursor: common/GCAM_region_names (#3
        Precursor: socioeconomics/A42.inc_elas
        Precursor: L102.pcgdp_thous90USD_Scen_R
3 - common/GCAM_region_names - read from file
        GCAM 32-region names (NA)
        Maps GCAM region IDs to region names
        Read from extdata/common/GCAM_region_na
        No precursors
4 - socioeconomics/A42.inc_elas - read from fi
        Building sector income elasticity, (pcg
inc_elas:unitless (% change in service demand
        aggregate buildings sector income elast
        Read from extdata/socioeconomics/A42.in
        No precursors
5 - L102.pcgdp_thous90USD_Scen_R_Y - produced
        Gross Domestic Product (GDP) per capita
of 1990 USD (MER))
        Computed as GDP/population.  Values pri
        historical; values subsequent are from
        Precursor: common/iso_GCAM_regID (#6 be
        Precursor: socioeconomics/SSP_database_
        Precursor: socioeconomics/IMF_GDP_growt
        Precursor: L100.gdp_mil90usd_ctry_Yh (#
        Precursor: L101.Pop_thous_R_Yh (#10 bel
        Precursor: L101.Pop_thous_Scen_R_Yfut (
6 - common/iso_GCAM_regID - read from file
        ISO to GCAM region mapping (NA)
        Maps iso codes to GCAM regions (includi
        ----------,,Former GCAM regions,
        Read from extdata/common/iso_GCAM_regID
        No precursors
```

Tracing – a specific example

~/Documents/Work/Code/gcamdata/

```
Restarting R session...

> library(gcamdata)
> ?module_gcam.usa_L254.transportation_USA
>
>
> ?dstrace
> dstrace("L100.FAO_ag_Exp_t", direction = "both", graph = TRUE)
1 - L100.FAO_ag_Exp_t - produced by module_aglu_LA100.FAO_downscale_ctry
        FAO agricultural exports by country, item, year (t)
        Downscale countries; calculate 5-yr averages
        Dependent: L106.ag_NetExp_Mt_R_C_Y (#2 below)
        Precursor: aglu/FAO/FAO_ag_Exp_t_SUA (#3 below)
        Precursor: aglu/AGLU_ctry (#4 below)
2 - L106.ag_NetExp_Mt_R_C_Y - produced by module_aglu_LA106.ag_an_NetExp_FAO_R_C_Y
        Net exports of primary agricultural goods by GCAM region / commodity / year (Mt)
        Aggregate FAO primary agricultural goods gross exports and imports and calculate net
        exports by GCAM region, commodity and year; Gross exports are adjusted so that global net
        exports add to zero; Re-calculate regional net exports using adjusted gross exports minus
        gross imports
        Dependent: L109.ag_ALL_Mt_R_C_Y (#5 below)
5 - L109.ag_ALL_Mt_R_C_Y - produced by module_aglu_LB109.ag_an_ALL_R_C_Y
        Primary agricultural good mass balances, by region / commodity / year. (Mt)
        Calculate primary agricultural good mass balances by GCAM region, commodity and year;
        Adjusts global and regional net exports to remove net negative other uses
        Dependent: L203.StubTechProd_nonfood_crop (#6 below)
        Dependent: L203.BaseService (#7 below)
6 - L203.StubTechProd_nonfood_crop - produced by module_aglu_L203.demand_input
        Crop non-food demand by technology and region (Mt)
        Map in crop non-food demand in calibration years by region / commodity; Remove any regions
        for which agriculture and land use are not modeled
        Dependent: demand_input.xml (#8 below)
8 - demand_input.xml - produced by module_aglu_batch_demand_input_xml
        XML data structures to be parsed by GCAM
        No dependents
7 - L203.BaseService - produced by module_aglu_L203.demand_input
        Base service of final demands (Pcal/Mt/bm3)
        Calculate the total final demands by supply sector in each region; Remove any regions for
        which agriculture and land use are not modeled
        Dependent: demand_input.xml (#8 above)
3 - aglu/FAO/FAO_ag_Exp_t_SUA - read from file
        FAO agricultural exports by country, item, year (tons)
        Read from extdata/aglu/FAO/FAO_ag_Exp_t_SUA.csv.zip
        No precursors
4 - aglu/AGLU_ctry - read from file
        Mapping of countries in AGLU databases to ISO codes (NA)
        Read from extdata/aglu/AGLU_ctry.csv
        No precursors
>
```



Status and future plans

- Initial work is 90%+ done
- The v1 Data System will be part of the next release of GCAM (early 2018)
- This will enable easier experimentation and greater transparency for the entire GCAM community