# Births, Deaths, Migration:

We model births + immigration into and deaths + emigration out of each “compartment”

* Migration (immigration and emigration) doesn’t depent on disease state, so it doesn’t change the disease prevalence in each compartment

## Modeling Births:

Birth rate= number of births/ population size

Fertility rate= number of births/number of female in childbearing ages

* JHEEM has used “birth rate” (although it’s called fertility rate in the code).
* SHIELD will use “fertility rate” to capture vertical transmissions more accurately

In JHEEM, the birth rate is static:

* rates= get.location.birth.rates.functional.form(): this computes the birth rate for each location based on age and race

We then take these rates and create a static functional form with log link so that we can model

* create.static.functional.form(value = rates, link = "log", value.is.on.transformed.scale = F) # not giving the log rates; don't need to transform this value after it’s returned

Calibration:

* There are 3 multiplier for birth rate by race that are sampled for calibration:black.birth.rate.multiplier, hispanic.birth.rate.multiplier, other.birth.rate.multiplier

STEPS:

1. **SPECIFICATION:** Defined the birth rate element with scale of rate and get a functional form from a new function X
2. **SPECIFICATION\_HELPER**: define function X based on available data as a dynamic or static function with appropriate link (we haven’t defined the multipliers yet)
3. **PARAMETERS:** defined the multipliers that should be used in calibration (by race, age, etc)
4. **PARAMETERS\_MAPPING**: map the new multipliers as alphas onto your function X

Modeling mortality

State specific death rates

Aging rate

If the people in a compartment have auniform agedistribution, the rate of exit is 1/duration: e.g., 100 people in age group 15-25, the exit rate is 1/10 per year.

However if we have more 15 years old than 25 year olds, the aging rate is not 1/10

We model variable aging rate for each compartment by age, race, sex, and infection state