

# Period Models Versus Models with Continuous Time



PillarOne Community Event, St. Gallen, 9./10. September 2010

Stefan Kunz, Intuitive Collaboration AG



- Model types
- Pure period models
- Period models with time sensitive components
- Models with valuation dates
- Other time issues
- Conclusions
- Discussion

- Typical uses are Solvency II or risk management models for relatively simple portfolios
- Even with pure period models some time issues arise
  - To properly deal with non-proportional aspects of reinsurance, the claims must have an occurrence date:
    - Merging claims of different perils/lines of business
    - Several non proportional treaties in serial order
- In pure period models time aspects are consequences of other modelling aspects.



- Typical uses are situations in which the reinsurance contracts periods do not coincide with the model periods
  - Annual risk management models for portfolios which are covered by reinsurance contract which runs from mid-year to mid-year.
  - Seasonal effects (hurricanes, ...)
- Time sensitive components are used to introduce local time aspects into pure period models.



- Modelling cash flows requires valuation dates
- If a model consists of different sub models each of which must be run at different time intervals
  - Modelling a securitization: Modelling claims from the original policies is one part and modelling the investors cash flow is another part. The latter comes at well-specified dates which do not necessarily coincide with the claims model.



- Especially multi-period and continuous time models use (calculated) historic and projected figures. Therefore it has to be possible to access data of other periods/time stamps.  
Examples:
  - generate ultimate claims and calculate the paids according to payment patterns. Future paids won't appear as such in the current period, but will be required in future periods.
  - multi period contract cover with term clauses:
    - used up/remaining cover
    - Permium written/earned
- To avoid any side effects components in RiskAnalytics are stateless by default. Sharing of data among periods is optional and requires a period store.

- Neither one of the types is better than the others  
→ The only thing that matters is the purpose of the model
- PillarOne as a framework does not preference any of the three time modelling approaches.
  - Complexity of continuous time models tends to be higher than pure period models.
- The component libraries to build period models or models with continuous time are not the same.
- Here be dragons!
  - date calculations

Stefan Kunz; Partner, Actuarial Tools  
+41 44 926 14 07; +41 76 370 31 86  
stefan.kunz@intuitive-collaboration.com

Intuitive Collaboration AG; Seestrasse 16; CH-8712 Staefa

# Period Models Versus Models with Continuous Time



Technical Appendix



- Time sensitive  
Simple time sensitivity is achieved with double properties, the integer part defines the period, the fractional part the time point within a period. Using a double allows to use such a component with different period lengths. Example: 2.5 in an annual model is interpreted as end of June of the third year.
- Continuous time, working with valuation dates  
Date calculations are rather tricky. PillarOne stands on Giants shoulders and uses a solid library: [Joda Time](#). Additionally, we have created some helper methods, most of them are available in the class **DateTimeUtilities**.

- The simulation engine is working with three different scope levels: simulation, iteration and period.
- The PeriodScope may contain either a ContinuousPeriodCounter, LimitedContinuousPeriodCounter, VariablePeriodLengthCounter. By overwriting `createPeriodCounter()` in a model class it is possible to define specific valuation dates. Adding one of these period counters allows components to access the current period start and end date.  
For interest accruals it is recommended to assume that intervals are left closed and right opened.

- In RiskAnalytics components should be kept stateless to avoid any side-effects.
- In multi period models its therefore necessary to attach a PeriodStore allowing to keep information over several periods.
- Period stores are governed by the framework and cleared after every iteration. Use an IterationStore to keep information for a whole simulation.
- Any kind of objects can be kept in a **PeriodStore**. They are accessible using a relative period index.
- Its not allowed to write to past periods.
- Example claims development: split an ultimate claim in several claim packets, send out the packet of the current period and store packets of future periods within the **PeriodStore**.

- Normally packets (e.g. claim information) are created and persisted in a specific period and iteration.
- There are use cases requiring results of the whole projection horizon in order to calculate a value for the current period.  
This use case is covered with the **PeriodStore** concept and by setting the **period** value of a packet.