

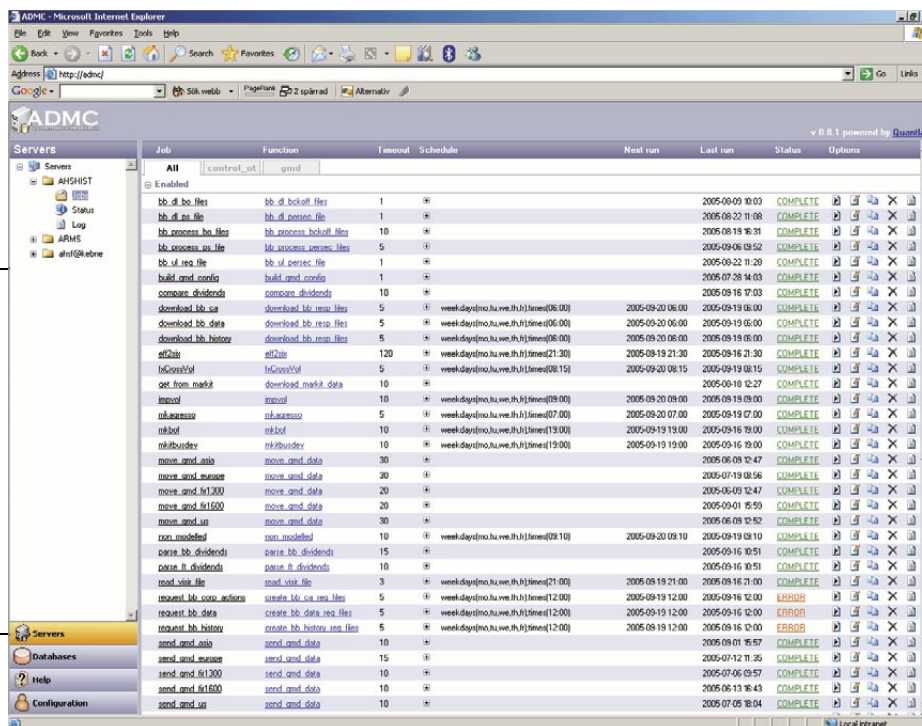
Quantlab® Batch Server

PRODUCT BRIEF

Quantlab® Batch Server — enterprise-wide calculation and distribution of analytics

The Quantlab® Batch Server provides a robust server environment for running scheduled tasks that require calculated quantitative output, using a comprehensive analytics library. Used in combination with the His-

tory Server, the system handles a wide array of data management tasks with processed data delivered in many formats. The intuitive web interface gives complete control of all tasks with integrated security.



Control multiple servers

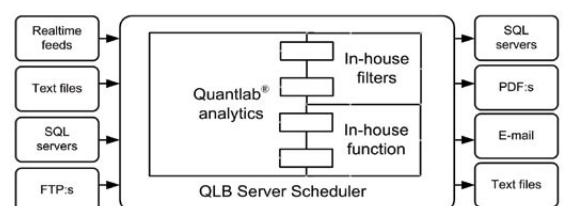
Manage job params, scheduling and job status

Control multiple databases

Integrated Windows security give action control and complete audit trail

Quantlab Batch Server - typical tasks

- End-of-day calculation of zero coupon curves
- Creation of RiskMetrics' style datasets
- E-mail distribution of pdf risk reports
- Corporate bond pricing and distribution to mainframe



The focus on user-friendly interfaces in the Quantlab® Batch Server minimises the risk for mistakes in daily operations and data management.

The use of the Quantlab® analytics library ensures quality assured calculation of financial data, and helps to economize on the use of scarce analysts resources.

Rapid filter coding

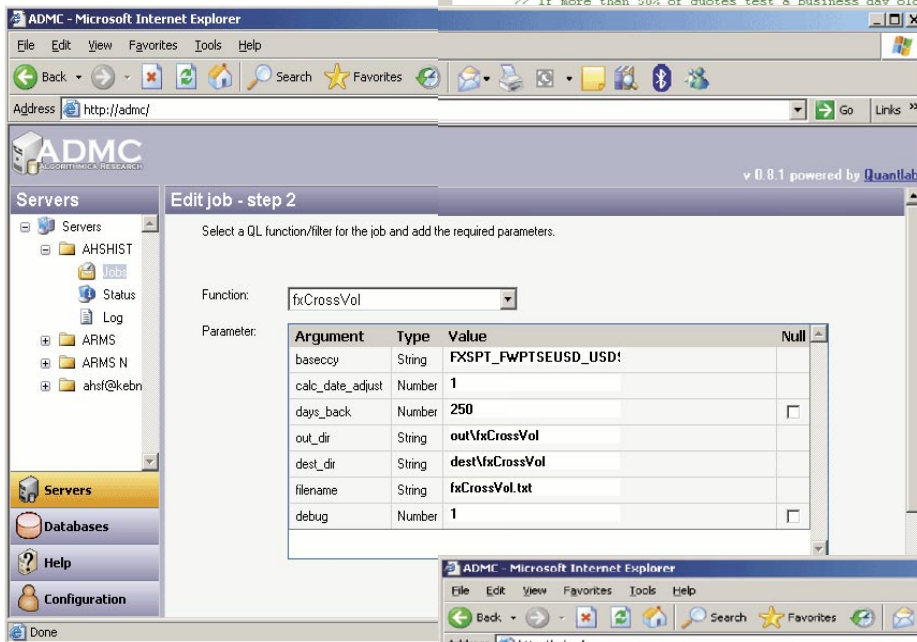
```

Expression *
logical zc_fit_govt(date calc_date, curve_name short_curve, curve_name long_curve,
  out curve my_blended_curve, out fit_result fr, out number days_back,
  number max_days_back, logical debug_mode)
{
  // special setting for Nelson-Siegel-Laquerre to replicate old solution
  vector(number) p_guess = [0.0953, 0.0, 0.4, 999];
  vector(number) p_min = [-0.1, -5, -5, -5, 1];
  vector(number) p_max = [0.3, 5, 5, 5];
  model nsl_scb = ns_laquerre_bound_params(ns_laquerre_parameter_names, p_guess, p_min, p_max);

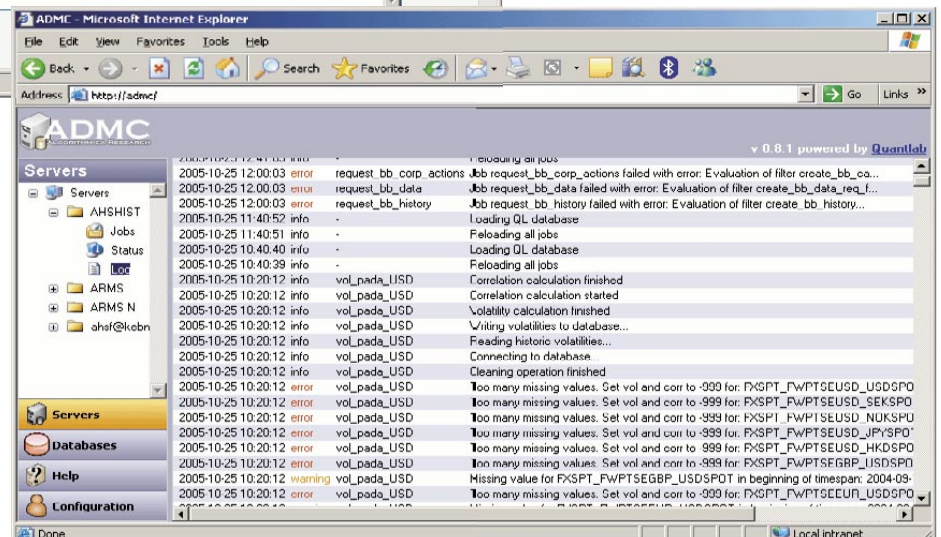
  vector(string) zc_names = ['zc02y', 'zc03y', 'zc05y', 'zc07y', 'zc10y'];
  vector(string) zc_maturities = {'2Y_NONE', '3Y_NONE', '5Y_NONE', '7Y_NONE', '10Y_NONE'};
  vector(number) zc_mat = [2, 3, 5, 7, 10];
  vector(instr_def) zcinstr_def = instr_def(instr_def('GENERIC ZERO COUPON'), zc_names, zc_maturities, 100, 0);

  curve my_long_curve;
  curve my_short_curve;
  curve my_zc_curve;
  logical no_success = true;

  // A bit of testing. If less than 50% of quotes on curve missing = exclude the missing and redo.
  // If more than 50% of quotes test a business day older
  curve, short_curve, my_short_curve, days_back, max_days,
  }
  
```



Easy scheduling



Log monitoring