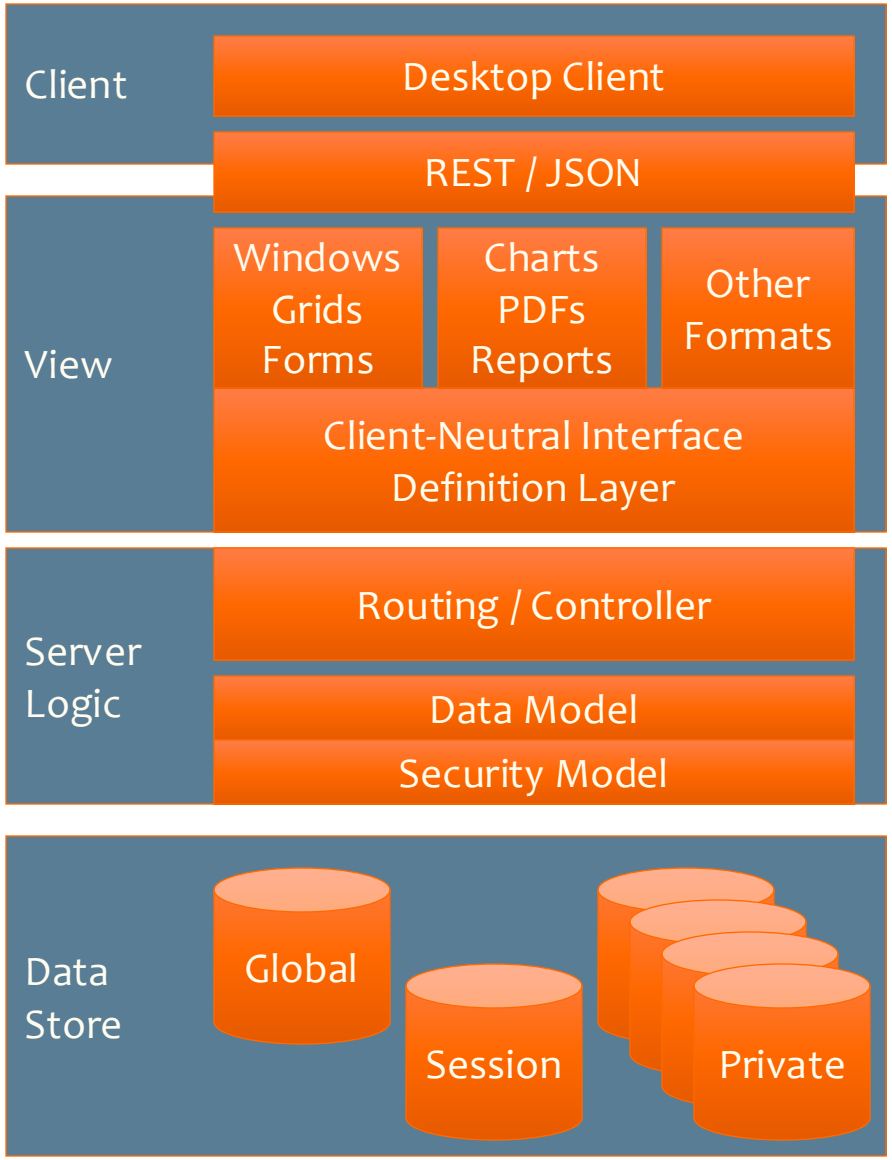
Technical Architecture and Implementation

James Kay

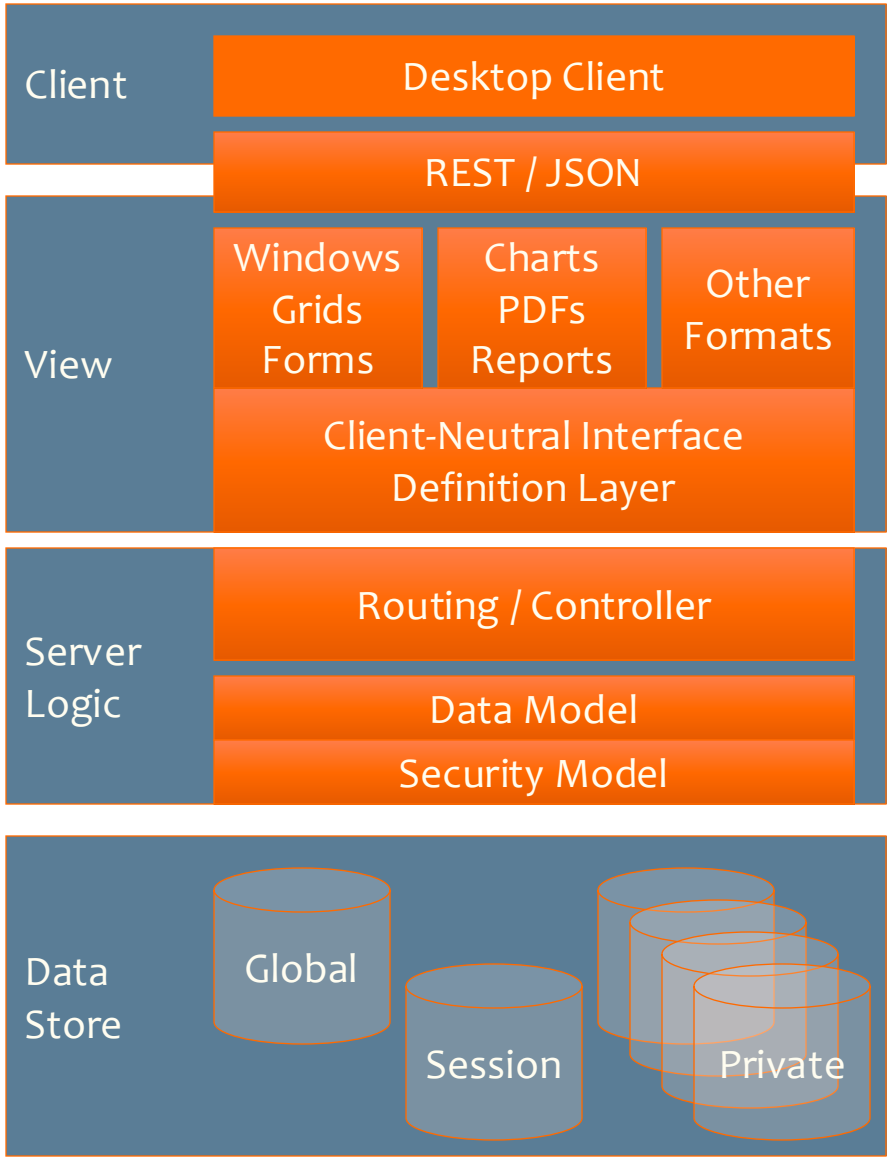
Workbooks

High-Level Architecture



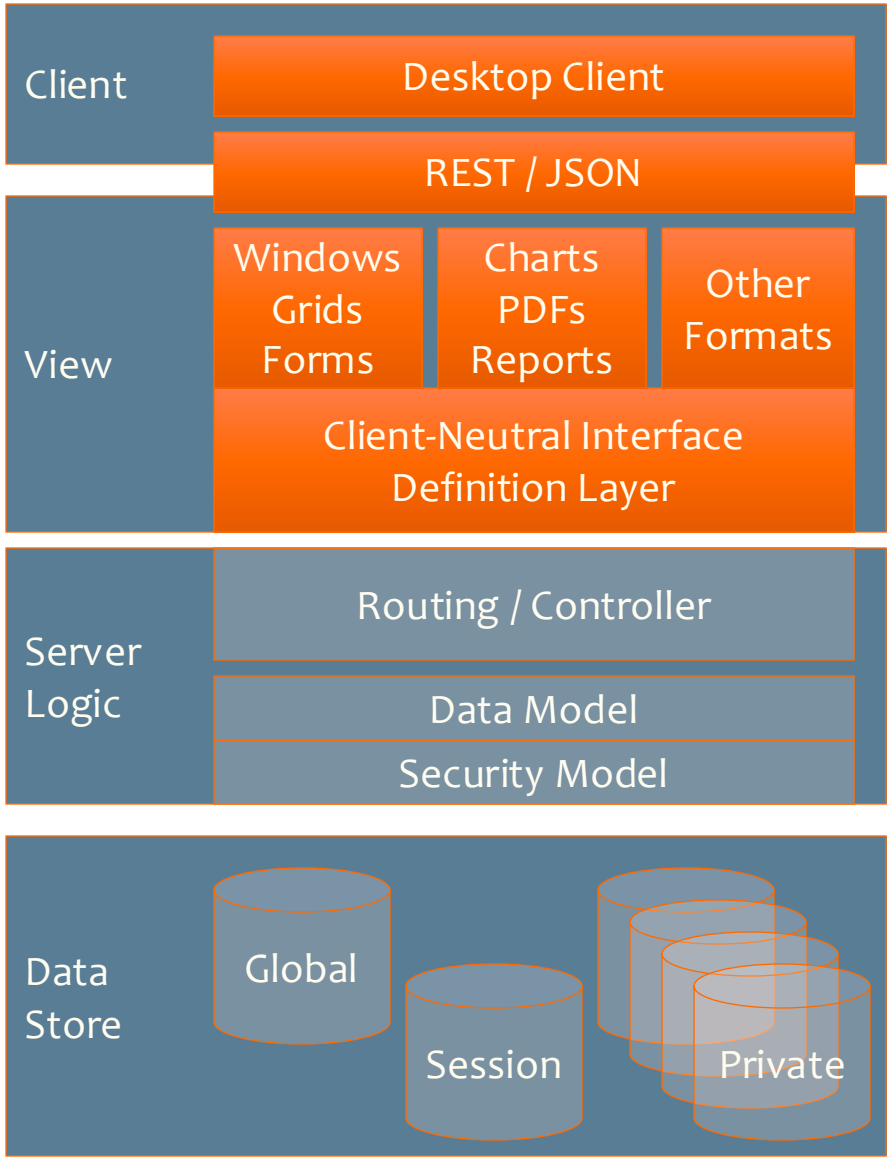
- Flexible Client Support
  - API clients
    - Language-agnostic
    - Specific additional support for PHP, Java, .Net, Ruby
  - Desktop Browser
    - Built on ExtJS framework
  - Mobile Client
    - Built using Sencha Touch framework
  - Process Engine (Hosted PHP)
- Ruby-on-Rails-based application architecture
  - Interface Builder Language (developer productivity and consistency)
  - Rails ActiveRecord and Routing/Controller
- Functional and Row-Level Security, Audit
  - Built into fundamental data model
- Many customer-private databases
- Layered and extensible

Application Server: View



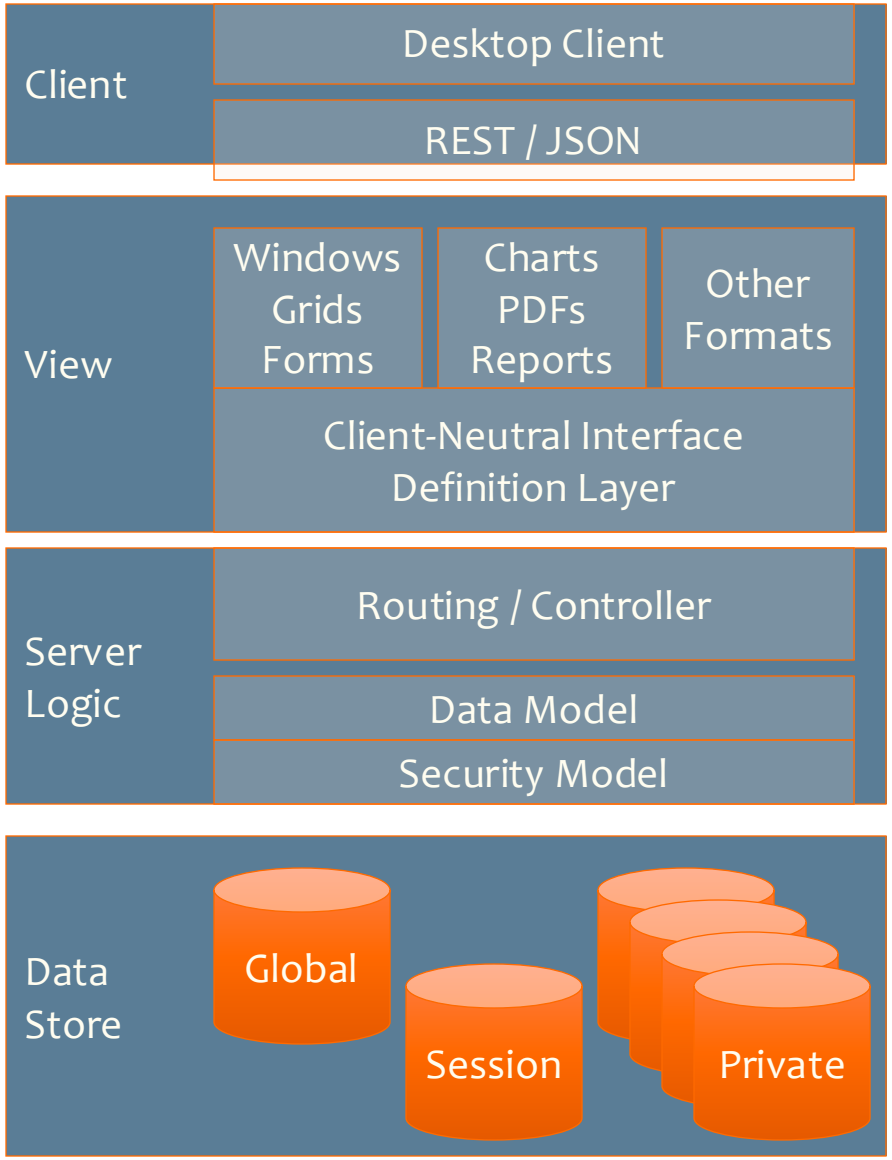
- Predominantly Ruby
  - Goal: maximise Developer productivity (at the cost of some raw performance/server)
  - No local state: load-balance; add more to scale
- RESTful API with batch operation support
  - Every object accessible through HTTP verbs
  - PUT/GET/POST/DELETE
  - All client/server interactions
- Domain-Specific UI Language
  - “WIBL” – OO implementation
  - Widgets described in Ruby
  - UI platform-agnostic
  - Goals: maximise consistency and code uniformity
- Rails Controller and ActiveRecord Model
  - Uses Rails for what it’s good at
  - Controller: 7 methods underlying REST (+update_many)
- Functional and Policy-driven Row-Level Security
  - Customer provisioning, DB access, Licenses (all in Global)
  - Granular Permissions (functions, user/group)
  - Row-level security (R, U, D, ch-own, ch-access)

Client Layer



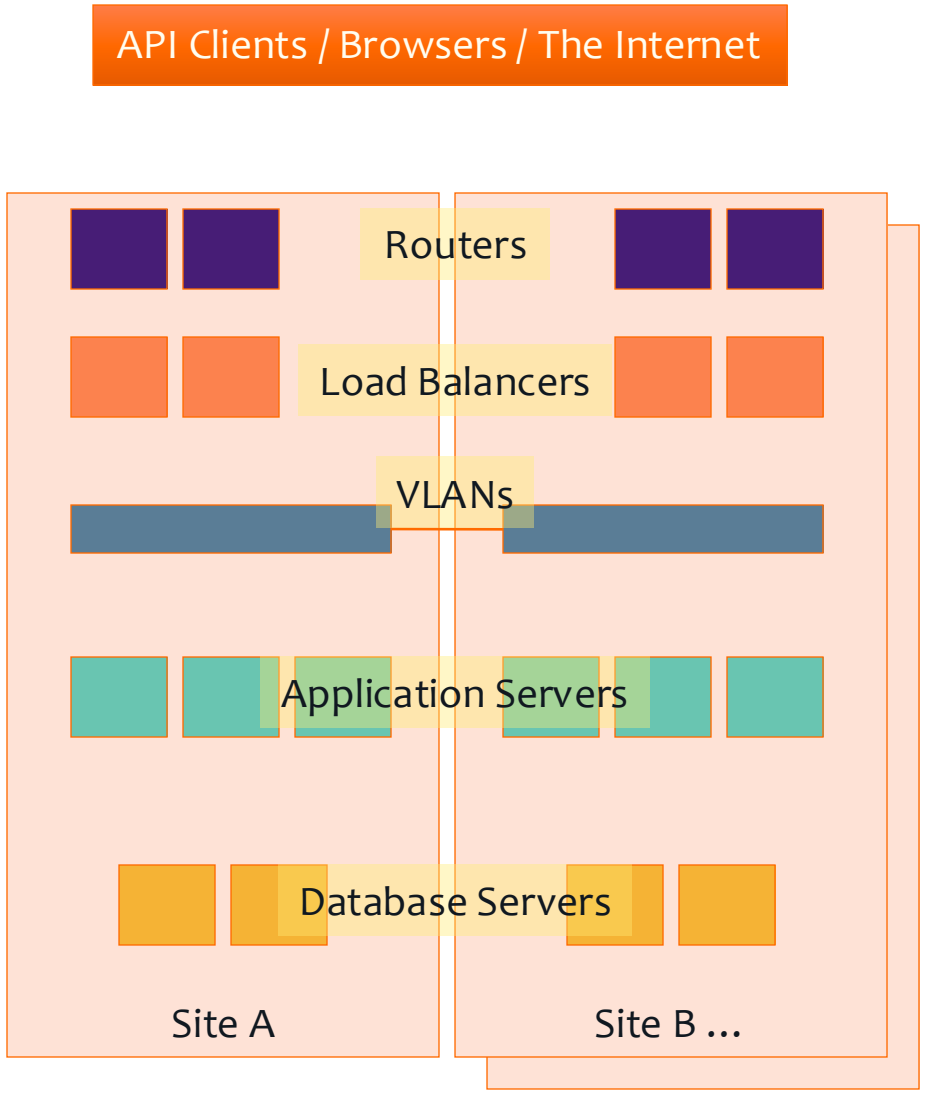
- ExtJS-based Desktop
  - AJAX: responsive
  - Portable across browsers
  - Contemporary Javascript; cached in browser
  - Feature-rich, e.g. Grids
  - Locally responsive, fewer server requests
- Notifications
- Permissions-driven menu
- Registration keeps windows consistent
- Translations driven from server

Database Layer



- Private databases for each customer
  - Security / Customer comfort
  - Reduced recovery time
  - Journaled: point-in-time recovery for past 7 days+
  - Customer sandboxes
  - Horizontal scalability, commodity hardware
  - Following expert advice
- Global database
  - Global-level data only
    - Customer / User login / Licences…
  - Small, fast recovery on failure
  - Less logic so more secure

Deployment Architecture



- Multiple Sites
  - All ISO27001 certified
  - Multi-link, resilient Internet connectivity
  - IPVS load balancing using source routing
  - High-speed inter-site ring (1Gb/s)
- Active/Active Deployment
  - Customers spread across all application servers, no local state on app servers
  - Databases replication between geographically distributed server pairs
  - Can re-locate customer data between servers without service interruption
- Infrastructure
  - Ops support VMs (nagios, dns, vpn, etc)
  - Consistent, fast deployment: tightly controlled OS image (netboot)
  - Extensible: Blade servers, some VMs
  - 64bit Linux, MariaDB