Actus Architecture and APIs Working Group

Agenda: WG call

Monday 27th Feb 2023 10 -11 am EST

Francis Parr

A “*useful*” Containerized and Componentized ACTUS Demo

Next Generation Regulatory Technology:
Architecture and an MVP to stimulate ACTUS adoption

Agenda for this week is to review this proposal for a “next step”: how to implement our ACTUS Risk Factor APIs in a form which could attract interest from (non-technology oriented) executives in banks and regulatory organizations

Principles / requirements for “Next Gen Regulatory Tech”

- Next generation regulatory analysis should be *cashflow* rather than *ledger* based
- It should be uniformly application to (essentially) all contract types
  - i.e. standard formulation of contracts across contract types
- It should allow analysis and reporting at multiple levels of aggregation down to individual contracts
- It should allow any risk scenario modelling to be applied to any contract portfolio
  - clean separation of contract definition and risk scenarios for analysis
  - Additional non-ACTUS contract terms to identify relevant risk factor models
- The package of reporting and analysis tools should be available at low cost
  - Individual small institutions will stress test their holdings and use it for generating regulatory and other reports
  - Regulators will used same base package but broader data collection, more complex risk models and systemic analysis
  - The base version of the package must be able to generate useful reports on a basic portfolio and be easily downloaded/installed
- In the resulting systemic analysis by the regulator
  - Risks and possible benefits from DeFi/Blockchain products blockchain based stable coins should be included i
  - Regulatory data must be sufficient to estimate critical parameters for (theoretical) systemic risk models

- => *ACTUS contract specification technology* is the start point for several of these needs
- => a *Containerized* ACTUS Demo will provide an “instantly understandable” set of basic analysis+reports for bankers / regulators
- => the *implementation* of the MVP demo will create an incremental path into more sophisticated scalable cashflow analytics
- => … but ACTUS cash flow generation will be “inside” the demo – not directly visible to the MVP user

What is in the Containerized ACTUS Demo MVP for a banker?

- Instant (free) startup on windows/mac/linux workstation
  - Prereq Docker Desktop – then a one click download/start
  - Browser download of Containerized ACTUS Demo docker image
  - Auto starts with local shiny demo localhost:3000

- Prompts user to enter
  - model bank description
    - EITHER aggregated - in form of simplified aggregated contracts (one per category)
    - OR granular – csv/xsl file(s) actus terms + department and ACTUS user tags
  - selection from choice of simple risk scenarios: interest rate shifts, prepayment rates, default model

- Generates – analysis, reports for the bank
  - simulate projected cashflows, displaying *profit, valuations, liquidity projections* – selected scenario
  - risk analysis … other report generation fpr the bank
  - aggregation/or drilldown by dept or other user+ACTUS category

- Generates - cashflow oriented (ACTUS) report for the regulator
  - A granular contract listing for the bank with tags for aggregation and risk selection as required by regulator
  - Anonymized consumer counterparties, LEI identified corporate counterparties

- => Banks can use this for additional analytic and regulatory reports *without change to their current systems*

What is in the Containerized ACTUS Demo MVP for a regulator ?

- Same facilities / report as for each individual bank + generate reports for a “set of banks”
  - Instant (free) startup on windows/mac/linux workstation
  - Prompts entry of sample bank granular holdings data , bank risk scenario selections
  - Generates projected profit, liquidity, valuation, risk for each bank/institution, selected scenarios
  - Generates sample cashflow oriented (ACTUS) regulatory report for each bank

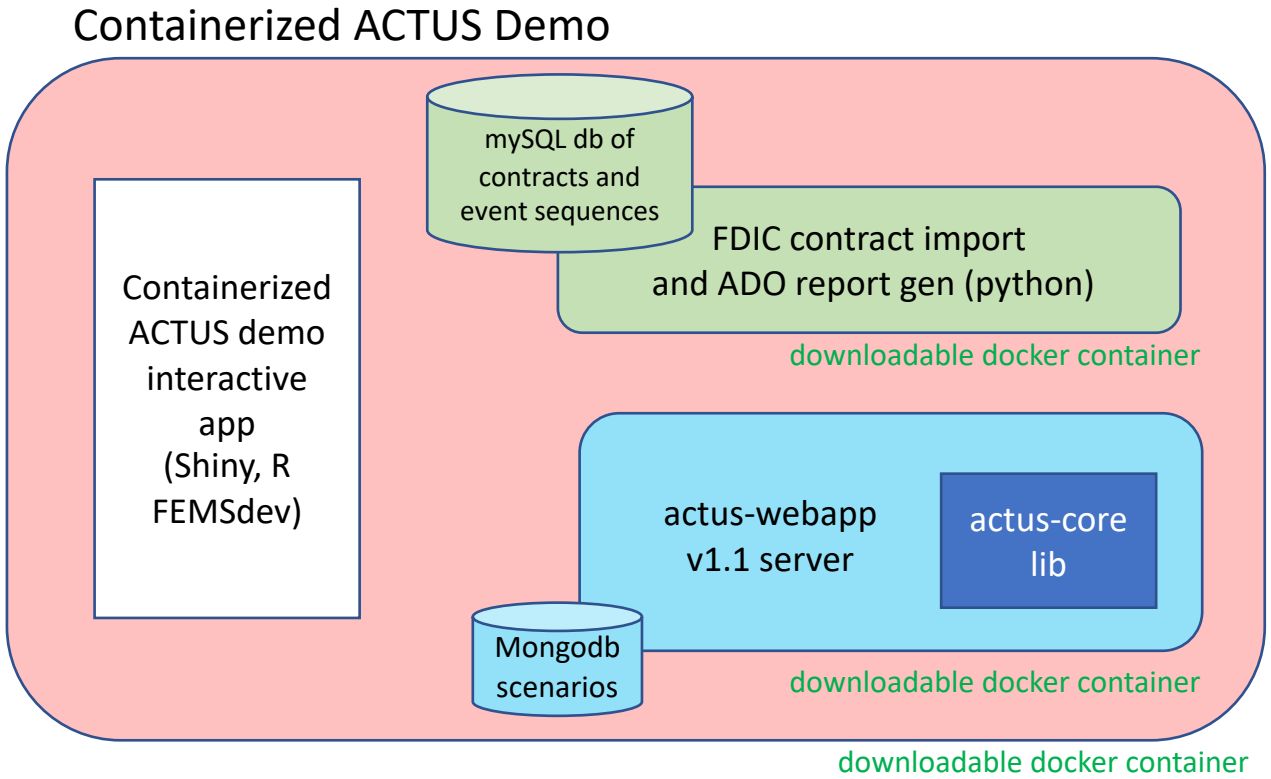
- Prompts regulator user to enter/gather
  - a set of cashflow oriented (ACTUS) regulatory reports for collection of banks
  - selection from choice of (more sophisticated) systemic risk scenarios: interest rate shifts, prepayment rates, default models

- Generates – analysis, reports for the bank
  - simulate projected cashflows, *profit, valuations, liquidity projections* – for each bank in selected scenario
  - financial institutions ordered by (1) greatest risk, (2) greatest degradation/change from previous period
  - extraction of systemic risk parameters from the cashflow oriented (ACTUS) regulatory reports
    - Key parameters as identified by DaDFiR3 risk team
  - Identification of effects of DeFi institutions and holdings on system stability
    - Effects identified by DaDFiR3 blockchain/stable coin team
    - Are shared blockchain contracts (modelled in ACTUS) and DeFi institutions changing the behavior of the system ?
    - Are block chain based stable coins improving the stability of the system ?

Containerized ACTUS Demo (CAD) Architecture

Demo requests

- Enter simple contract
- Input contracts file
- Enter simple scenario
- Input scenario file
- Generate reports –
  - profit, liquidity, value
  - risk, system order etc
- Systemic (tbd)
- Blockchain (tbd)

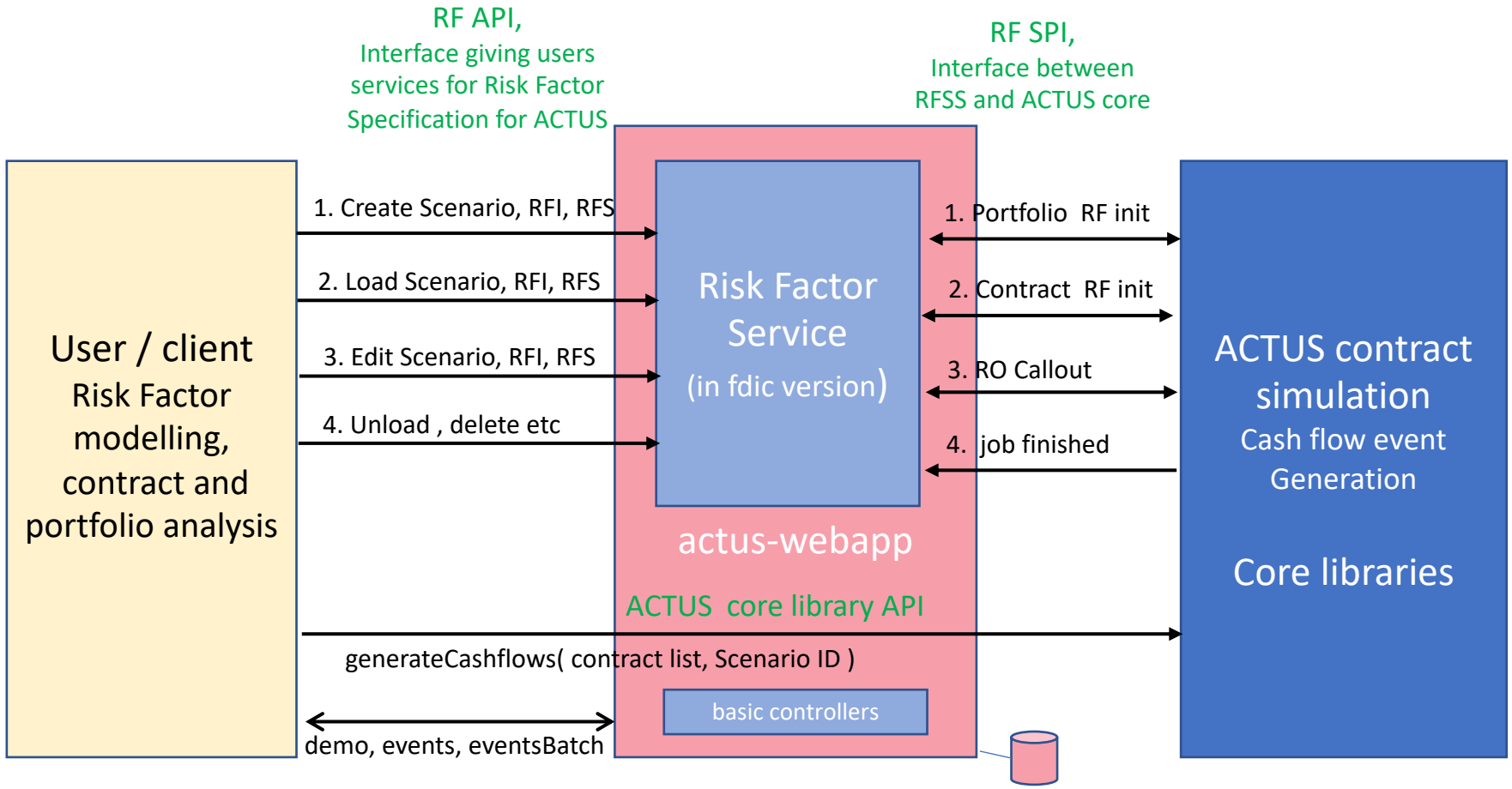


… think Russian dolls !
Dockerfile allows one to specify a list docker component containers making up the (master) containerized ACTUS demo; components are automatically download and installed in setup
- Kubernetes application with 3 components ?

CAD enables *incremental* adoption of cashflow based banking/regulation

- If a financial institution/regulator executive sees potential value in the function of the basic containerized ACTUS demo … they can subsequently initiate:
  - python/sql expertise to review the FDIC Pbank portfolio import code and explore importing data from their existing portfolio files into the demo
  - python/SQL_reportgen expertise to review the FDIC report code and generate summary and aggregated reports relevant to their operational practices
  - Java/JSON/riskmodel expertise to review the actus-webapp V1.1 code and augment with risk model scenario extensions relevant to operational practice
- Potential to do an updated demo of cashflow (ACTUS) based analysis *customized* for the institution
  - Also developing and executing on their institution owned systems so not using Actusfrf processing resources for commercial purposes – avoids questions of data security, resource misuse etc
- => this is a recursive application of the API/SPI duality approach we exposed in defining the ACTUS Risk API
  - Low entry cost potential users with financial rather than technology background want to see potential use via API
  - Practical exploitation and deployment require expertise and software development patterned on available opensource code as the documentation of underlying SPIs
- A regulator could also assess the potential value of more granular, more timely, net lower cost to reporting institution regulatory reports
  - Would regulator provide a downloadable “conversion” container for mapping standard contract formats into ACTUS ?
- Additional containers or for systemic risk and blockchain/stable coin may be helpful

Risk Factor Service – Architecture and Interfaces



Old/current actus-webapp supports:

- generateCashflows (contractList, in-line-marketObjectRFs) at https://demo.actusfrf.org/eventsBatch
- The contract demo at https://demo.actusfrf.org/demo

FDIC version demo *also* (partially) supported: createSave, edit, delete Scenario + generateCashflows(contractList, scenarioId) at https://XXXXX/scenarios/save etc and https://XXXXX/simulations/runScenario

Different categories of ACTUS user

- Risk units supporting banks, finance houses etc.
  - Must be able to use their custom (sophisticated) risk models with actus-core library cash flow contract simulation
  - Has the skill to (1) write java code into a custom-webapp module (2) integrate this with existing/provided actus-core jar
  - => These users *need* and a well-documented actus-core SPI - to build *custom* actus-webapp servers

- Exploratory users - can I use ACTUS for my project, research etc?
  - Want to experiment with using custom data for simple risk models, demonstrate ACTUS feasibility
    - marketObject reference data
    - Prepayment, creditDefault, depositAccountTransactions
  - Expect to use standard provided actus-webapp server
    - No motivation – and possibly limited skill – in java coding building a custom webapp
  - => These users need a well documented published (data only) RFS API and a provided RFS-enabled actus-webapp server

- Recommendations:
  - => Build an actus-webapp server enabled with simple RFS example models for deposit account transactions, prepayment and credit events; post a publicly accessible version of this server at https://demo.actusfrf.org:8080
  - => Publicize and support with “user” documentation the open-source code of this “example” actus-webapp as *the specification* of how to use the actus-core SPI
  - This is the best (only?) way to support both categories of ACTUS user
    - Preserving the focus of actus-core on deterministic cashflows
    - While enabling use of this with a variety of risk Factor models
    - Some discussion of how many examples of risk factor model needed …. compatible with – open source project

CAD (Containerized ACTUS Demo) is Minimal Viable ACTUS Product

- Minimal
  - Must be able to deliver interesting analytic projections on portfolios of contracts using actus-core library as central component

- Viable
  - Instantly installable “understandable” demo with simple example data and output reports
  - Demo software open and structured for easy incremental enhancement / integration into existing enterprise owned systems

- Achieved by
  - Adding Risk Factor API, portfolio import, aggregation and report generation + an interactive browser demo application around actus-core contract simulation
    - ACTUS has value because it can simulate the future cash flows for any contract portfolio with any risk scenario
    - All analytic / risk reporting based on these cash flows
  - Component architecture with open source ‘example’ software to enable incremental enhancement
    - Modify individual components for each different area of enhancement
    - Select best technology for each component – R Java, Python, SQL, Mongo, Shiny, Spring
  - Docker Container architecture for easy demonstration install –
    - Builds on the Risk Factor API initiative

Why only now – are we – ACTUS - considering this?

- Adoption of ACTUS is painfully slow .. and not accelerating
- User oriented documentation did not exist
  - Market risk factor API was unpublished
  - No Risk factor API (except in unpublished RiskFactor)
- Review of initial user guide prompted (thank you Guy Rackham):
  - “we need something that will allow a banking exec – with no specialized programming skill to see potential benefit – less than a day of effort”
- DaDFir3 Project has no industrial partner – so must propose and prototype an MVP - for Next Generation Regulatory Technology
  - - i.e an easily installable demonstration of *cashflow oriented* analytics and regulation
- Integrating: actus-core, actus-webapp, FDIC /ADO python portfolio import and report gen, with FEMSdev R Shiny demo in *containers* can be installable and demonstrate real potential value.
  - There was code missing, as well as documentation and no demo focused on end value
- Our technology orientation and concern not to *dilute* the actus-core cashflow gem with multiple-risk models, import utilities, reports, discouraged us from creating an installable demo of value