ACTUS External Risk Factor Service API – Specification

April 2024

Draft: Jeff Braswell and Francis Parr

1 Introduction and Overview

1.1 Purpose of this document

1.2 Risk Observer role in the ACTUS Technical Specification: review

1.3 Outline

- Motivation: Why an external risk service
- Concept: External Risk service, ACTUS service, risk observations
- Context for the Risk API interface: User, ACTUS service, Risk Service relationships
    - User -> Risk service interactions: entering risk data, models and scenarios
    - User -> ACTUS service interactions: job orchestration, analysis requests
    - ACTUS service -> Risk service interactions: core Risk API
    - ACTUS service -> user interactions: returning analysis and job results
- Interactions in the Risk API and related ACTUS service interactions
    - initializeJob(…)
    - initializeAnalysis(…)
    - setAnalysisPortfolio(…)
    - startAnalysis(…)
    - endJob(…)
    - initialiseAnalysisRiskService(…)
    - getAnalysisScenarioData(…)
    - startOfContractSimulation(…)
    - riskObserverEvent (…)
    - analysisCompleteForRisk(…)
    - analysisCompleteForUser(…)
    - jobCompleteForUser(…)
- Exceptions returned through the API
- Additional Risk Service Implementation guidance
    - Configuring the ACTUS server to work with a Risk Service
    - State within the Risk Service
    - Risk Factor concepts – and implementation guide
    - i.e. User interactions with Risk service to set up: Scenarios, Reference Indexes, Yield Curves, risk model types, risk model instances , etc Market data
    - Selecting risk models for a specific contract
- Planned updates to the Reference Implementation
- Other Possible External services which would require additional APIs
    - Contract and Portfolio Store
    - Simulation output service – cashflows store
- Relation to earlier ACTUS documentation – migration issues

2 Motivation for an external Risk Factor API

In this section we explain why it is important to define a Risk Factor Interface to an external Risk Factor Service for use by an ACTUS contract simulation service.

- An import feature of ACTUS is that for a financial contract defined using the ACTUS standard terms, it is possible to algorithmically determine the cashflows “expected to occur” for *a given forward-looking Risk Factor Scenario*.

Such a Risk Factor Scenario could include projections for market interest rates, projected stock prices, and would also include risk factor models for such things as option exercise, loan default, prepayment, deposit account behavior etc.

- The ACTUS standard defines the terms for contract specification but imposes no constraints on how risk factors making up a risk scenario could be modelled and applied in a contract simulation.
- The goal is to show that the ACTUS standard for contract specification can be used *with user-specified* risk models, risk modelling products and risk specification technology.
- In contrast to contract cashflows which are be defined by the contract logic and are therefore amenable to standardization, the field of risk modelling is complex, subjective and context dependent. In selecting a risk model, many different trade-offs are made between accuracy, expressiveness, computational efficiency etc. Hence risk modelling is an area for competitive differentiation by financial institutions and not amenable to open-source standards based offerings.
- The ACTUS specification describes in concept a “Risk Observer” mechanism by which the simulation of contract logic can interact with a risk service at different simulation observation times. The risk observer is expected to apply its risk model specifics affecting contract performance at each of these observation points.
- The current open-source reference implementation in Java of the ACTUS standard distributed by ACTUS-FRF, includes some sample risk factor modelling – particularly for projected market rates. Java coding and a rebuild of the actus-core and actus-webapp modules is required to extend or provide alternate risk factor models interacting with ACTUS contract simulation.
- Mapping the abstract Risk Observer callout specification in the technical specification to a well-defined external RiskFactor API, and providing a version of the ACTUS reference implementation which gets its risk scenario input by calling out to this API will show that any user selected risk modelling technology which can be connected to this API can work with ACTUS reference implementation contract simulation without requiring changes to the ACTUS service.
- Having an externalized RiskFactor API demonstrates that risk modelling services interacting with ACTUS cashflow generation can be implemented on any server using any programming language.
- An additional benefit is that modifying any existing risk modelling and contract simulation system to use ACTUS contract cashflow logic becomes a much better-defined task. If risk modelling the existing system can be modified to support the ACTUS

RiskFactor API, then plugging in actus-core logic to use this becomes a component swap-in.

3 Proposed External Risk Service and Risk Factor API Concept

In this section we summarize the flows in the current ACTUS service (with integrated risk modelling) and contrasts it with architecture and interaction flows for a ACTUS service interacting with a fully externalized Risk service

3.1 As-is ACTUS: risk processing integrated into the ACTUS service

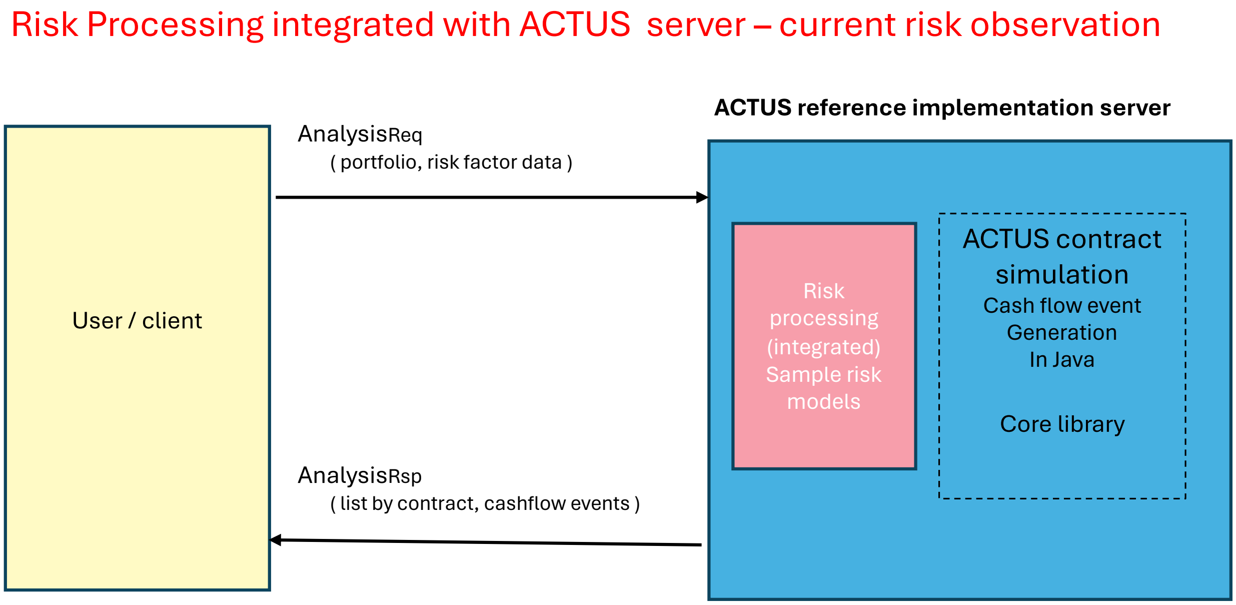

Risk Processing integrated with ACTUS server – current risk observation
ACTUS reference implementation server
AnalysisReq
(portfolio, risk factor data)
User / client
Risk processing (integrated) Sample risk models
ACTUS contract simulation
Cash flow event Generation In Java
Core library
AnalysisRsp
(list by contract, cashflow events)

Figure 1. Current ACTUS reference implementation – integrated sample risk factor service

We start by summarizing the "shape" of an ACTUS simulation request as provided currently with risk processing provided as a component *within* the ACTUS contract simulation service. Then the concept of an External Risk Server can be described as a set of changes to this basic flow.

- The most basic form of an ACTUS request is to send to the ACTUS server as a REST API request, serialized JSON data which represents the terms of the contract to be simulated.
- The response from the ACTUS server – assuming that the input data is well formed and complete is a list of cashflow events returned as serialized a JSON list of events, each with its date/time, event type, payoff, current principle, accrued interest etc.
- But if the simulation is for a contract with variable, market-determined interest rates, it is also necessary to send a time series of projected values for all indexes referred to in the contract as part of the initial request.

- In practice, one needs most frequently to simulate the cashflows of a portfolio - a list of contracts rather than a single contract.
- In a portfolio simulation, with current versions of the ACTUS reference implementation server, the JSON simulation request sent to ACTUS consists of:
  - a list with an ACTUS contract specification for each of the contracts in the portfolio
  - Specification of projected future values for each of the market indexes used in portfolio contracts to reset variable rates and set future market prices.
  - Specification of additional risk models to determine contract behaviors such as prepayment and default which can happen at some specific simulation time in the life of a contract and depend on the state of the contract, terms of the contract and projected market rates at that specific point in simulation time.

The summary above describes the pattern of riskFactor handling for the *current ACTUS demo Risk Factor API example,* where:

- risk factor service is available only for some small, limited set of "toy/sample" risk models.
- risk factor service is integrated into the process of the ACTUS reference implementation service (not externalized)
- all risk factor input data is supplied as input parameter data in the initial request to simulate a portfolio of contracts.

The flow of such a portfolio request is illustrated in Figure 1 above. In this flow the "risk callouts" occur entire inside the ACTUS reference simulation server. They involved exchanges of java objects in memory between the core contract logic and some simple provided example risk models for market data, loan prepayment and loan default.

3.2 Proposed new architecture with all risk modelling external to ACTUS

What we mean by an external Risk service is that all risk modelling and provisioning is provided in a separate service which interacts with the ACTUS service using well defined calls, and no requirement to share any memory with the ACTUS service. The Risk Service and ACTUS service could be deployed on separate physical processors.

When we externalize all risk factor modelling and processing in this way, the architecture and flows to run ACTUS cashflow based analysis changes because there are now two independent service involved in the contract or portfolio simulation. The user, analyst or client now interacts with a Risk Service to enter risk factor data and risk scenarios; but they request the contract or portfolio analysis and receive returned results by interacting with the ACTUS service. The ACTUS service now gets risk model results and data from the new external Risk service during the requested analysis using the new Risk Factor API. Configuration of the ACTUS service to tell it which Risk service to interact with and identification of the risk scenario to use in any analysis must now be specified by the user during user to ACTUS interactions.

Figure 2 below illustrates how in the new flow externalizing risk modelling, there are now separate sets of interactions involving different combinations of parties. The user/analyst interacts with the Risk service to define market data projections, risk models to be use specific contract simulations and identified risk scenarios. The form of the interactions to do this will be specific to each risk service. Since our primary goal is to define an architecture in which ACTUS simulations work with *any* external risk factor modelling and provisioning, the ACTUS RFAPI places no restrictions on this set of interactions. This set of interactions is defined by the risk service which the user chooses to use. These are the step 1 interactions in Figure 2.

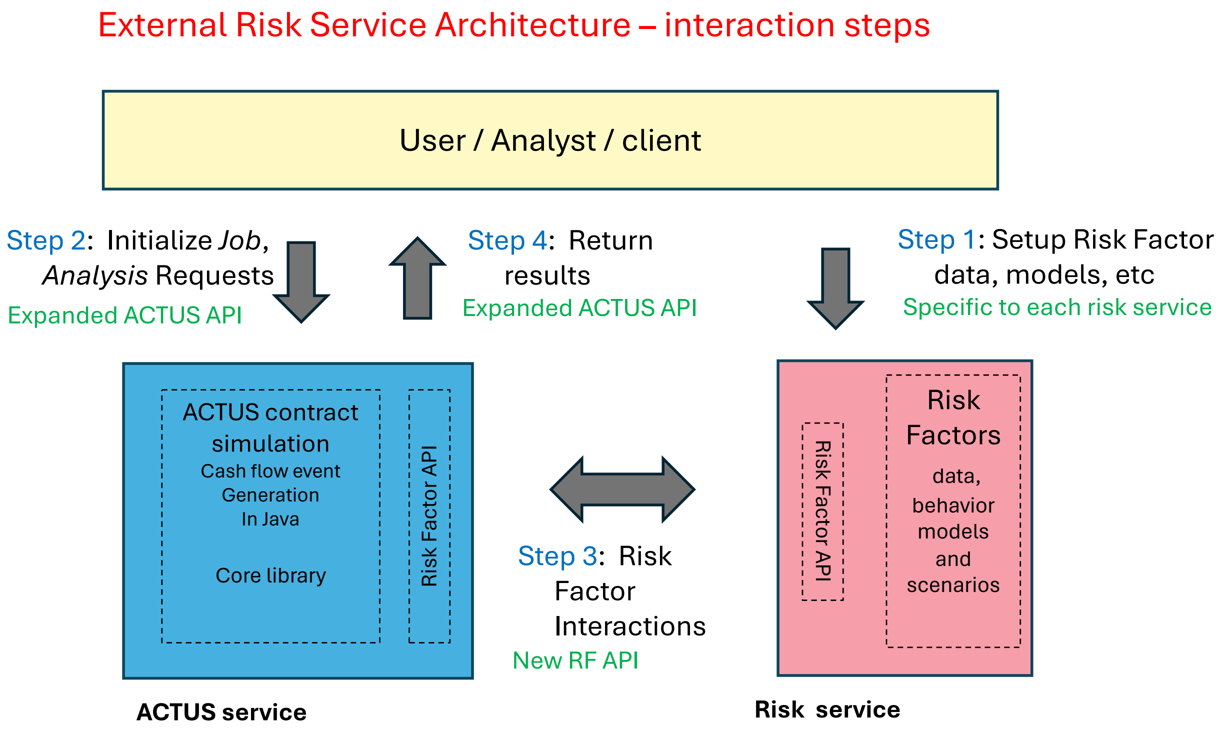

External Risk Service Architecture – interaction steps
User / Analyst / client
Step 2: Initialize *Job*, *Analysis* Requests
Expanded ACTUS API
Step 4: Return results
Expanded ACTUS API
Step 1: Setup Risk Factor data, models, etc
Specific to each risk service
ACTUS contract simulation
Cash flow event Generation In Java
Core library
Risk Factor API
Step 3: Risk Factor Interactions
New RF API
Risk Factor API
Risk Factors
data, behavior models and scenarios
ACTUS service
Risk service

Figure 2 . An external risk factor service (pink) and Risk Factor API interface (green)

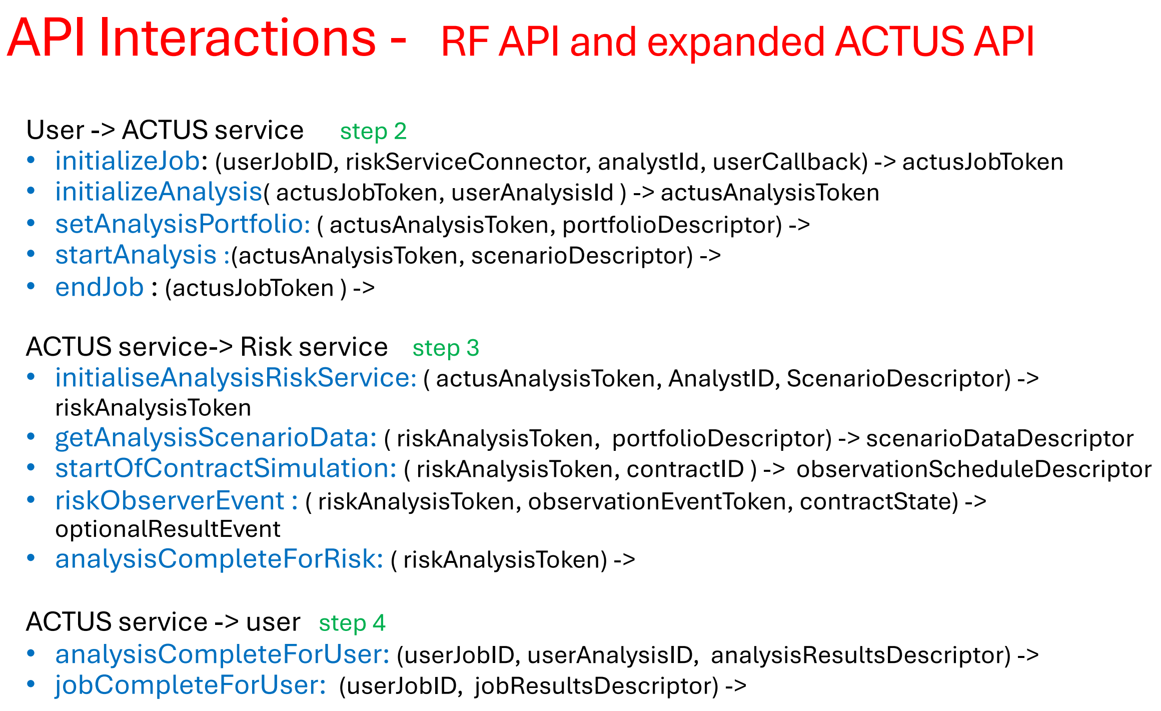
A second set of interactions are those initiated by the user/analyst but directed towards the ACTUS server. This set includes an analysisRequest interaction, where the user asks the ACTUS service to simulate a particular portfolio of ACTUS contracts using a specified risk scenario, but also includes interactions and analysis request parameters:

- to initiate a job consisting of a *series* of analysis requests
- to configure the ACTUS service to use a particular specified external Risk Service for all analysis which are part of this job.
- to have the portfolio of contracts for an analysis requests specified as a “portfolio descriptor” - enabling the portfolio data to be fetched on demand by the ACTUS service rather than passed explicitly as parameters of the analysis requests
- to have the risk scenario of an analysis request specified as a “scenario-identifier” meaningful to the external Risk service – enabling market data for the risk scenario to be

fetched on demand by the ACTUS service rather than passed explicitly in the service request.

Hence the set of interactions initiated by the User and directed to the ACTUS service, includes the analysis requests available in the ACTUS reference implementation today, but *expands* this API to accommodate the needs of the external RFAPI to the Risk Service and the scalability requirements associated with industrial strength use of ACTUS for risk analysis. These are the step 2 interactions in Figure 2.

A third set of interactions occurs between the ACTUS service and the configured external Risk service during the processing of an analysis. These interactions make up the core of the Risk Factor API. Having them formally specified and made part of the ACTUS standard is new and is the key feature enabling a compliant ACTUS service implementation able to work with any risk modelling service.



API Interactions - RF API and expanded ACTUS API

User -> ACTUS service step 2

- initializeJob: (userJobID, riskServiceConnector, analystId, userCallback) -> actusJobToken
- initializeAnalysis(actusJobToken, userAnalysisId) -> actusAnalysisToken
- setAnalysisPortfolio: (actusAnalysisToken, portfolioDescriptor) ->
- startAnalysis :(actusAnalysisToken, scenarioDescriptor) ->
- endJob : (actusJobToken) ->

ACTUS service-> Risk service step 3

- initialiseAnalysisRiskService: (actusAnalysisToken, AnalystID, ScenarioDescriptor) -> riskAnalysisToken
- getAnalysisScenarioData: (riskAnalysisToken, portfolioDescriptor) -> scenarioDataDescriptor
- startOfContractSimulation: (riskAnalysisToken, contractID) -> observationScheduleDescriptor
- riskObserverEvent : (riskAnalysisToken, observationEventToken, contractState) -> optionalResultEvent
- analysisCompleteForRisk: (riskAnalysisToken) ->

ACTUS service -> user step 4

- analysisCompleteForUser: (userJobID, userAnalysisID, analysisResultsDescriptor) ->
- jobCompleteForUser: (userJobID, jobResultsDescriptor) ->

Figure 3. List of Risk Factor API related interactions

The interactions include:

- the ACTUS service initiates Risk service support of a new analysis.
- the ACTUS service passes the portfolio descriptor of the analysis to the Risk service.
- the ACTUS service passes the risk Scenario identifier for this analysis to the Risk service.
- the ACTUS service receives all market data projections (e.g. future interest rate and stock price projections) for the analysis from the Risk service.

- the ACTUS service initiates Risk service support of a new contract simulation within the current analysis.
- the ACTUS service receives a schedule of risk observation events for the contract just starting simulation from the Risk Service – based on risk modelling for that contract.
- the ACTUS service initiates a risk observation at a scheduled observation point, passes the state of the contract and hears back from the Risk service whether some specific risk-modelled behavior (e.g. prepayment, contract default, option exercise, principal change etc.) has occurred at this point in simulation time.

Figure 3 provides a list of these ACTUS service to Risk service interactions. The details of each of these interactions and the rationale for it are described in section 4 below.

These are the step 3 interactions in figure 2.

The final group of interactions in Figure 2 are the ACTUS initiated flows returning results of analyses and results of a completed to the user who initiated the job. A scalability feature of these interactions is that results are passed back using a “results descriptor” which can be a service interface and a key allowing for on demand fetching of the results rather than requiring that they be returned as a direct JSON field in a RESTful request.

These are the step 4 interactions in Figure 2.

Risk Processing external to ACTUS simulation server – the RF API

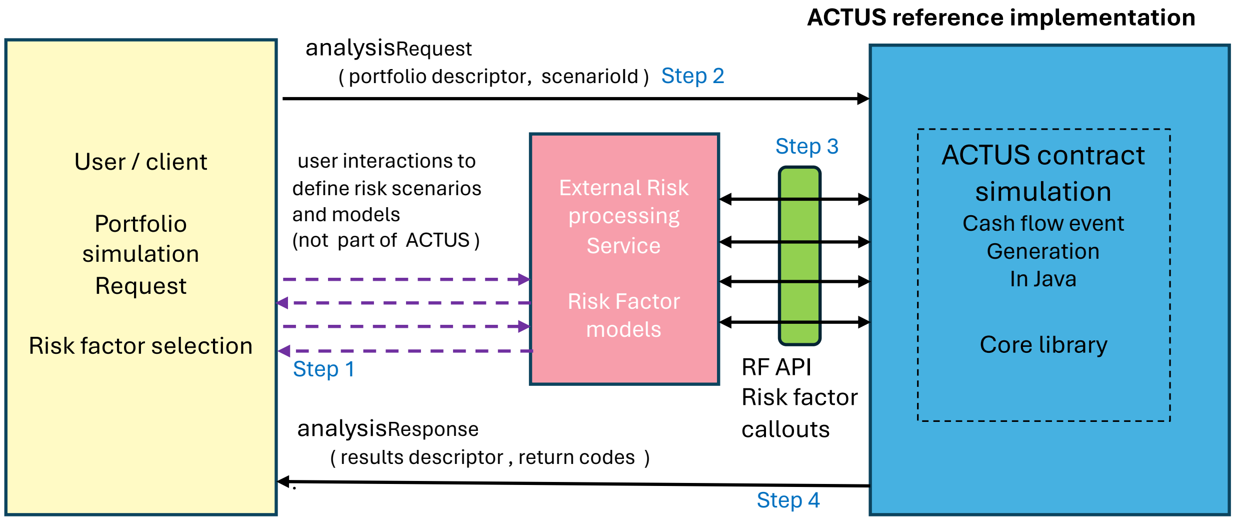

ACTUS reference implementation
analysisRequest
(portfolio descriptor, scenarioId) Step 2
User / client
Portfolio simulation Request
Risk factor selection
user interactions to define risk scenarios and models (not part of ACTUS)
Step 1
External Risk processing Service
Risk Factor models
Step 3
RF API Risk factor callouts
ACTUS contract simulation
Cash flow event Generation In Java
Core library
analysisResponse
(results descriptor , return codes)
Step 4

Figure 4. Interaction steps for an analysis with example flows.

Figure 4 above shows the sequence of interaction steps for a single analysis request. In this diagram we see that for an ACTUS service interacting with an external Risk Service, there are:

- a set of step 1 interactions between the user and the Risk Service to set up risk factor models data and scenarios for use by an ACTUS service; these interactions are specific to each risk Service and are not defined on an ACTUS Risk Factor standard.
- a step 2 interaction for each analysis request between the user and the ACTUS server, specifying the portfolio to be simulated and the risk facto scenario to use for the simulation. This interaction is an extension of the existing ACTUS API in that the risk factor information is made available to the ACTUS service in the form of a risk scenario identifier rather than as risk data passed directly within the analysis request.
- during processing of the analysis requests, there are multiple step 3 interactions between the ACTUS service and the Risk service. There are the core interactions of a new External Risk Factor API enabling risk scenario information, portfolio information, risk observation schedules, and the results of risk observations to be communicated between the ACTUS service and the Risk service.
- step 4 interactions are used by the ACTUS service to communicate results of analyses and jobs back to the user. These extend the current ACTUS in allowing more scalable result descriptors, rather than requiring all results to be passed as data fields in the response to an analysis request. Asynchronous responses are also permitted.

4 Risk Factor API Interface Definitions

The high-level flow of an ACTUS portfolio simulation using an external Risk Service and ACTUS Risk Factor API 2.0 will involve:

- Interactions between the end-user and an External Risk Service to introduce risk factor data and models for contract and portfolio simulations.
- Interactions between the end-user and an ACTUS-service to set up a job and request portfolio simulations identifying a risk "scenario" for each simulation.
- Interactions between the ACTUS-service and a Risk-service during a portfolio simulation - enabling the External Risk service to be the provider of all risk modelling and risk factor data, while the ACTUS-service provides the contract term aware logic generating expected cashflow event sequences.

The ACTUS Risk Factor API 2.0 defines the form of interactions between an ACTUS- service and an External Risk Service - the third category in the list above. However, this definition has dependencies on simulation job setup – as described in the second category of interactions. It also assumes that required risk factor and model data are available in the External Risk-service and have been entered in some way – as described conceptually in the first category above.

4.1 **User -> Risk Service** interactions – entering Risk Factor data and models.

The end-user application interacts in step1 with the External Risk service using user-facing APIs to that service to:

- Define projections for reference data used by contracts in the ACTUS portfolio to set interest rates and prices during a portfolio-scenario simulation.
- Create, manage, and identify risk factor models which will operate as external risk factor observers. These models use values of contract terms, contract state, non-ACTUS user defined contract characteristics and current reference data values, to determine whether events such as prepayment, default or other risk determined behavior occur at specific times in a contract simulation.
- Characterize the complete set of risk assumptions and evaluation environment for a specific portfolio simulation as a Scenario identified by a Scenario descriptor.
- A Scenario descriptor identifies a specific catalogued collection of structures in three classes of data structure types:
    - single-dimensioned, scenario-specific time series (market rates, interest rates, exchange rates, macro-economic indicators, etc.) accessed via Market Object Codes (MOCs)
    - yield curves with scenario-specific term structure values.
    - risk factor models associated with the portfolio of contracts that will be analyzed in conjunction with the scenario assumptions.

These operations are determined by the user interface to the External Risk Service but not defined as part of the ACTUS RiskFactor API.

4.2 User -> ACTUS service interactions: job orchestration, analysis requests

The user interacts with the ACTUS service in step 2 to orchestrate requests for portfolio simulations. In this section we use *analysis* to mean: a request to an ACTUS-service to simulate a particular portfolio of contracts using a specific risk scenario. We use *job* to refer to a series of analyses initiated by one analyst-user and configured to work with a specified risk-service.

- **initializeJob** (userJobID, riskServiceConnector, analystId, userCallback) -> actusJobToken
    - Initializes a job which will consist of a series of analyses.
    - The job is configured to use a specific external Risk service – to be used by the ACTUS-service as its source of risk factor data and models for this job.
    - The user provides some callback information on how to return job results.
    - the output of the interaction is an actusJobToken. This is entered on all following interactions associated with the job. It is created by the ACTUS-service for its convenience in locating processing structures and resources assigned to this job.

- **initializeAnalysis** (actusJobToken, userAnalysisId) -> actusAnalysisToken
    - The actusJobToken identifies the job in which this Analysis will be a part
    - Initializes an analysis, as a step in a defined ACTUS job.
    - The output is an ACTUS-service generated actusAnalysisToken to be included in any following requests related to this analysis.

    - The actusAnalysisToken is linked to the actusJobToken of the owning job
    - The userAnalysisId is the user's unique identifier for this Analysis. If output cashflow event sequences from the simulation are written out to a persistent store, this identifier is used as key metadata identifying the origin of the dataset,

- **setAnalysisPortfolio**(actusAnalysisToken, portfolioDescriptor) ->
    - After this interaction, the portfolio for the Analysis is set; the ACTUS server knows how to access and load portfolio data – primarily the terms for each contract in the portfolio
    - The actusAnalysisToken is the token returned from a preceding initializeAnalysis() interaction and identifies the Analysis for which a portfolio is being set.
    - The portfolioDescriptor could be a service connection and a key enabling the ACTUS-service to fetch portfolio data on demand from some database or portfolio-service.
    - Alternatively, the actual portfolio data could be passed as a data field in the interaction which the ACTUS-server could parse and read into its local memory.
    - There is no output data returned as part of this interaction.

    The RFAPI2.0 standard does not specify the implementation method for communicating access to the portfolio data. It does require that after the interaction, the portfolio data is accessible to the ACTUS -service.

- **startAnalysis**(actusAnalysisToken, scenarioDescriptor) ->
    - the actusAnalysisToken ties this interaction to a previously initialized Analysis for which a portfolio has already been define by a setAnalysisPortfolio( ) interaction.
    - User passes in a riskScenarioDescriptor identifying the risk scenario to be used for this analysis.
    - The Risk service uses this identifier to locate a set of riskfactor data and risk behavior models predefined by this user, which will enable it to interact with the ACTUS-service providing required risk factor interactions.
    - On completion of the startAnalysis interaction, the ACTUS-service begins active processing of the Analysis.
    - there is no output data returned as part of this interaction.

- **endJob**(actusJobToken) ->
    - In this interaction the user informs the ACTUS service that there will be no more analysis requests as part of this job.
    - The actusJobToken and any resources assigned to the job in the ACTUS-service are freed up.
    - There is no output data returned as part of this interaction.

4.3 **ACTUS service -> Risk service** interactions: (the core RF20API)

While processing *Analyses* as part of job, the ACTUS service initiates step 3 interactions with the Risk service. During each *Analysis*, the ACTUS service interacts with the (external) Risk service to get projected future values of market indices and to get risk behavior model results as *risk observations*. Each risk observation is a request for a decision by the risk service whether for this contract at this point in simulation time, some risk behavioral event does or does not occur. There interactions between the ACTUS service and the Risk service during an analysis are the core of the Risk Factor 2.0 API.

- **initialiseAnalysisRiskService** (actusAnalysisToken, AnalystID, ScenarioDescriptor) -> riskAnalysisToken
    - This alerts the external risk service that it will need to process requests from the ACTUS service for this analysis. The Risk service will typically create a processing context or thread to for its handling of the analysis.
    - The actusAnalysisToken is passed – this identifies the analysis being serviced.
    - AnalystID and the scenarioDescriptor enable the Risk Service to locate relevant riskfactor information as requested by the user for this analysis.
    - The interaction returns to ACTUS a token to be used in all subsequent interactions with the risk service associated with this analysis.

- **getAnalysisScenarioData** (riskAnalysisToken, portfolioDescriptor) -> scenarioDataDescriptor
    - Gives the risk service the scenario key from the user and access to all data defining the portfolio.
    - The returned scenarioDataDescriptor gives the ACTUS server access to projected future values for market indexes and prices as needed for the analysis.
    - Making this information available to the ACTUS service may be implemented as a service connection and key, allowing on demand retrievals, or a returned data set of the interaction.
    - The scenarioDataDescriptor returned to the ACTUS service must include a unique timeseries of projected future values for any marketObjectCode used by at least one contract in the portfolio to reset rates or determine cashflows. Hence the scenario data returned may be created specifically to meet the requirements of the analysis portfolio.

- **startOfContractSimulation** (riskAnalysisToken, contractID) -> observationScheduleDescriptor
    - This interaction is initiated by the ACTUS service as it starts the simulation of a contract in the analysis portfolio. The contractID in the interaction parameters identifies the contract being simulated.
    - The riskAnalysisToken locates processing context and resources in the risk service associated with this analysis.

  - The Risk service must determine (1) which risk behavior models should be used with this particular contract (2) ensure that these models are available and loaded into memory and (3) generate a schedule (for each active risk behavior model) of *observations*.
  - Each observation will trigger an *observation callout* from the ACTUS service to the risk service at the scheduled simulation time driving an observation callout to be described below.
  - The output from the startOfContractSimulation interaction is an observationScheduleDescriptor. This give the ACTUS service access to the observation schedules for all the risk behavior models being applied to this contract. The observationScheduleDescriptor may be a link to a service with a key to retrieve the data dynamically, or it may be returned as a complete data set with all scheduled for all attached models in the returned data of the interaction.

The Risk service has access to all the terms and attributes of the contract identified, by contract id because in the preceding getAnalysisScenarioData interaction it was passed the portfolio descriptor for the Analysis. Using this it can retrieve any of the ACTUS terms or non-ACTUS user defined attributes of any of the contracts in the portfolio. It is expected that one of the user defined attributes of each contract will be a list of risk behavior models to be attached to this contract when it is simulated. The Risk service will interpret those model names using a names space associated with the End-user. The AnalystID/UserID was passed to the Risk service in the **initialiseAnalysisRiskService**( ) interaction.

- **riskObserverEvent** (riskAnalysisToken, observationEventToken, contractState) -> optionalResultEvent
  - A riskObserverEvent – as described in the ACTUS Technical Specification – is a call out from the ACTUS service to a risk service to determine whether, at this point in simulation time, a behavior risk model event does or does not occur in this contract simulation.
  - The riskObserverEvents were added in to the simulation schedule by the **startOfContractSimulation**() interaction described above.
  - ACTUS defines a priority for each event type, so that if there are exactly simultaneous events in the simulation schedule, the order in which they are processed is well defined.
  - A riskAnalysisToken provided by the ACTUS service as input, locates processing resources in the risk service associated with this analysis - including the risk behavior models attached to this contract.
  - The observationEventToken provided by the ACTUS service as input, specifies the risk observation event details. This includes the simulation time of the observation and the identifier of the risk behavior model to be observed.
  - ACTUS contract state for this simulation is made available to the interaction. This includes current nominal value, current interest rate, accrued interest etc. along with contract identification and time.

    - The Risk service processes this risk observer event by making the current contract state, all contract terms and current relevant risk scenario values available to the model and determining whether a behavior event does or does not occur.
    - The output of the interaction is an optionalResultEvent. This is a report back to the ACTUS service of whether a behavior event does occur or not with specifics of the event. The event type when this is returned must be an ACTUS contract event (defined in the ACTUS data dictionary).

  Typical examples of risk behavior models are: (1) prepayment of a loan – where this is permitted by the contract, (2) default on a loan and, (3) exercise of an American option. Input relevant to the behavior modelling for each of these cases would at a minimum include:

    - for loan prepayment modelling – the current rate on the loan, expected rate reset values, amount outstanding, current market interest rate for loans in this risk category, the original tenor of the loan and time remaining in the loan contract.
    - for loan default modelling - the risk characteristics of the counterparty, the industry, and risk category of the loan product, the current state of the business cycle and interest rates (most of the above attributes of a contract would be specified as user defined rather than ACTUS attributes of the contract).
    - for option exercise modelling – European options are exercised if they are in the money at expiry, but for American and Bermudan options, exercise can depend on the degree to which they are in the money, the time until expiry, alpha and beta estimates for the underlying stock – possible combined with a Black-Scholes valuation.

  Since the interface is from ACTUS to a completely independent and external risk service, any behavior modelling is possible. Competitive industrial strength risk behavior modelling is likely to use many more parameters than the examples sketched above.

- **analysisCompleteForRisk** (riskAnalysisToken) ->
    - The Risk Service is free to discard this riskAnalysisToken and any resources associated with it.
    - This command is required is there are no more analyses in the ACTUS job.
    - it may be defaulted by a new **initialiseAnalysisRiskService**( ) request from the ACTUS server - if we assume serial processing of analyses with a job.

4.4 **ACTUS service -> User** interactions

These are the step 4 interactions where result data from Analyses within a job are returned to the end user after processing by the ACTUS service.

- **analysisCompleteForUser**(userJobID, userAnalysisID, analysisResultsDescriptor) ->
    - This returns the results of the simulation to the User from the ACTUS server

    - The userResultsDescriptor may consist of a service connection and key or could returned data set or other
    - The user call back information in the job enables these results to be posted to some service which will write them to store.
    - The user may also want statistics and reports on the analyses processing - compute time, o used etc.

- **jobCompleteForUser(**userJobID, jobResultsDescriptor) ->
    - This returns the results of all analyses in a job in batch form.
    - The jobResultsDescriptor may consist of a service connection and key or returned

5 Additional guidance on functions provided in a Risk Service

5.1 Categories of Risk Factor Data

In this section we identify and describe the different categories of Risk Factor data and models which are involved in the Risk Factor API callouts from the ACTUS server.

The primary forms of Risk Factor specification required for ACTUS contract simulation are:

- projected market data
- contract behavioral models.

The projected market data is a specification of what a portfolio simulation should assume about future interest rates, future stock price movements, future dividend payment trends. For example, ACTUS specification of a variable interest rate loan must define: (1) the schedule of interest rate resets, (2) the base market reference indicator to be used as a basis for determining the new rate, and (3) a spread to be added to the current value of the base to set the new rate value. Different risk scenarios enabling this form of interest rate setting therefore consists of "projections" of future values of a named market reference index. The elements of a market date risk factor risk are:

- a market risk factor Identifier – unique to a particular projected set of values for a reference index
- the label for the reference index used as a base for the value calculation – identified in a variable rate contract term.
- the time series of projected future values for reference index.

We have used an example of contract interest rate setting to describe market risk factors, but the same approach can be applied to risk factors relating to stock or commodity price movements affecting cash flows of futures and options contracts, also to dividends on stocks.

Contract behavioral models specify how a specific category of contract is likely to respond to a combination of future market data values, and future contract state. Examples include the modelling of prepayment of loan contracts when this is allowed, modeling of contract default and of likely loss given default. Modeling of American option exercise would be another example of behavioral modelling. Each of these behaviors needs to be specified as a risk factor for successful simulation of contract and portfolio cash flows.

The behavior models to be used with a contract can depend on: (1) the type of the contract - for example loans need prepayment and default modelling, options may need exercise modelling; (2) user defined attributes of the contract such as which industry, credit rating of the borrower; (3) state of the contract - remaining time until loan termination or amortization, current interest rate compared with current market rate; (4) state of the business cycle etc. (5) exercise models on a stock option could involve alpha and beta estimates on the underlying, industry or commodity models etc. User defined attributes of a contract are not ACTUS terms and nit standardized associated with contracts to assist risk modelling.

5.2 Choosing risk factor market data and behavior models for a simulation

The external risk service is responsible for selecting the market data and behavior models for contract simulation by the ACTUS server.

All contracts in a portfolio presented for simulation should be simulated with the same set of market risk factors. The set of market data risk factors provided to the ACTUS server to simulate a portfolio of contracts must provide *complete and unique* coverage. This means that for each marketObjectCode used in the ACTUS terms of at least one of the contracts in the portfolio, the set of market data risk factors provided by the external risk service during portfolio simulation must include *exactly one* time-series of projected future values for that marketObjectCode.

Compatibility of a set of projections for future interest rates or stock or commodity prices is an important consideration. For example, when projecting forward interest rates for different tenors, having an arbitrage free set of market risk factors adds consistency properties to any portfolio simulation using this. Deciding exactly what consistency properties to require is a responsibility of the External Risk Service. The ACTUS server receives a set of market data risk factor values for a portfolio simulation from the external Risk Service but applies no specific consistency rules.

In practice it may be helpful for the External Risk Service to organize names in the market Data Risk factor space to address consistency. A market data interest rate "scenario" may have a name like "yield curve x% up". Specific marketObjectCode projections to be used in portfolio simulations might then be named to reflect the combination of interest rate scenario and tenor separately: i.e. riskFactor< "yield curve x% up", tenor=1year>, riskFactor< "yield curve x% up", tenor=5year> etc. Similar market modelling and consistency rules apply to projection of sets of stock market or commodity prices for distinct but related marketObjectCodes (stock and commodity ticker symbols)

Choosing the set of risk behavior models for contracts in a portfolio, is very different from selecting market data risk factors, because each contract needs to be simulated with a valid set of models appropriate for that contract type, its contract terms and even non-ACTUS, non-cashflow related attributes of the contract.   Prepayment and default models are needed for loan contract types such as PAM, LAM, ANN etc. but make less sense for Options and Future Contracts. Models specifying “exercise” behavior are important for American and Bermudan but not for European Options – which use a simple “exercise if in-the-money” rule.

The External Risk Service is responsible for selecting and initializing the appropriate set of behavior risk factor models.