DAL/API MVP - Scope, Capabilities, Job Stories

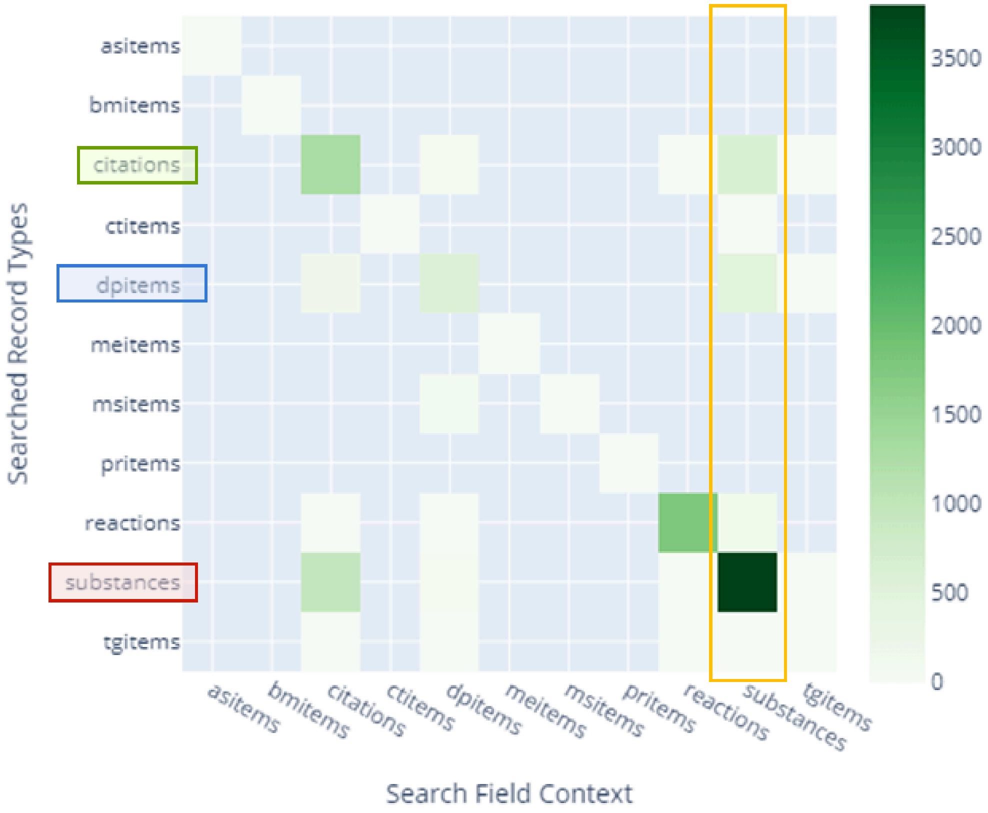
Scope API MVP

The MVP for the new Reaxys search API shall demonstrate:

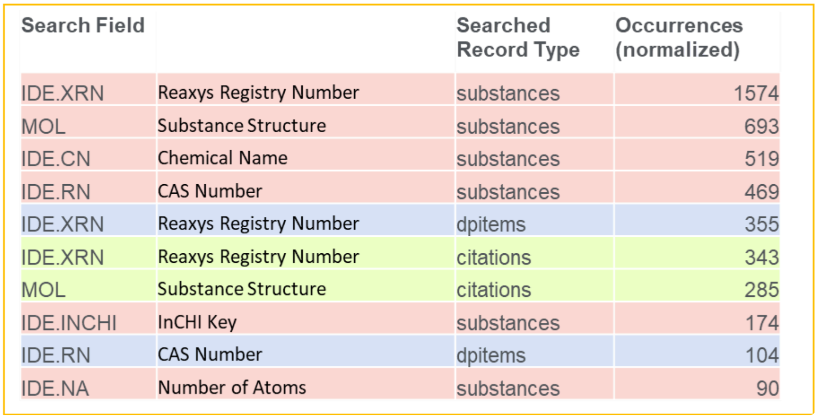
- basic search functionalities (fact and structure search)
- simplification of cross context search and retrieval workflows
- RESTful API

Basic Search Functionality

Basic search functionalities such searching by properties and chemical structures are commonly used as exemplified by the existing API usage behavior of using substance properties.



Substances fields most commonly used in queries

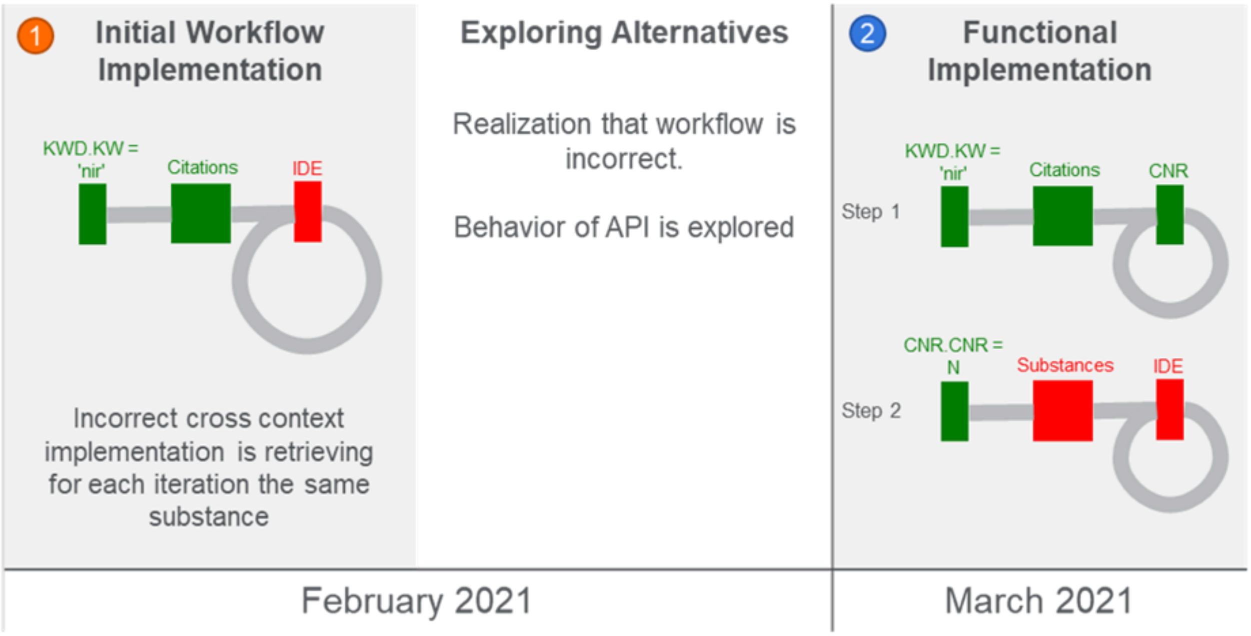


Search Field		Searched Record Type	Occurrences (normalized)
IDE.XRN	Reaxys Registry Number	substances	1574
MOL	Substance Structure	substances	693
IDE.CN	Chemical Name	substances	519
IDE.RN	CAS Number	substances	469
IDE.XRN	Reaxys Registry Number	dpitems	355
IDE.XRN	Reaxys Registry Number	citations	343
MOL	Substance Structure	citations	285
IDE.INCHI	InCHI Key	substances	174
IDE.RN	CAS Number	dpitems	104
IDE.NA	Number of Atoms	substances	90

The analysis of the substance fields that are used most commonly (based on the usage of the existing API) exemplifies that basic queries my database fields (e.g. `IDE.CN`, `IDE.RN`) and by substance structure need to be supported by the new API.

Simplification of Cross Context Search and Retrieval

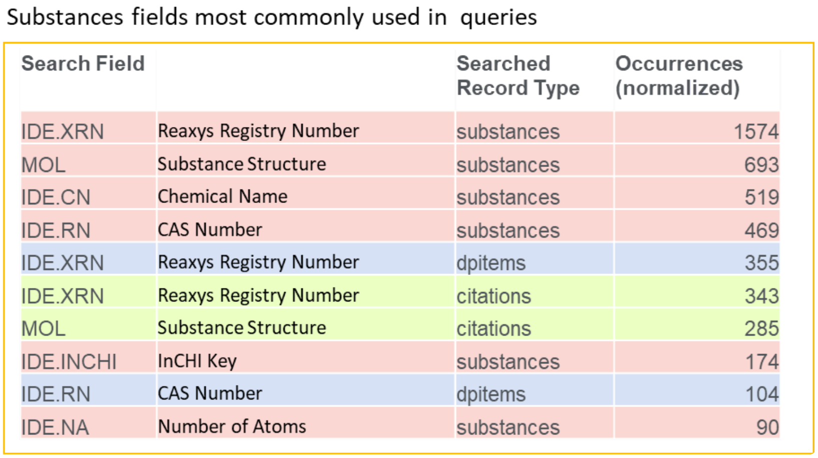
The new API shall reduce the complexity of retrieving information from multiple contexts (such as documents and substances, or substances and bioactivities). How to proper execute “cross context“ retrievals is not easily understood by API users and can lead to incorrect implementations.


1
Initial Workflow Implementation
KWD.KW = 'nir'
Citations
IDE
Incorrect cross context implementation is retrieving for each iteration the same substance
Exploring Alternatives
Realization that workflow is incorrect.
Behavior of API is explored
2
Functional Implementation
Step 1
KWD.KW = 'nir'
Citations
CNR
Step 2
CNR.CNR = N
Substances
IDE
February 2021
March 2021

Pain Points
API implementation concepts are not intuitive and onboarding without guidance can require too much experimentation

The example above is based on the experience of the API customer JSR. The current API was used initially incorrectly to retrieve substances from a citation subset by requesting the IDE fact from a citations hitset. After some time customer realized the problem and switched to a multi stage workflow where (1) the citations where searched (2) the citation numbers where captured (3) the substances where identified for the captured citation numbers (4) desired properties where collected.

The MVP shall improve the experience by removing the intermediate step of collecting and searching by citation numbers.



Substances fields most commonly used in queries

Search Field		Searched Record Type	Occurrences (normalized)
IDE.XRN	Reaxys Registry Number	substances	1574
MOL	Substance Structure	substances	693
IDE.CN	Chemical Name	substances	519
IDE.RN	CAS Number	substances	469
IDE.XRN	Reaxys Registry Number	dpitems	355
IDE.XRN	Reaxys Registry Number	citations	343
MOL	Substance Structure	citations	285
IDE.INCHI	InCHI Key	substances	174
IDE.RN	CAS Number	dpitems	104
IDE.NA	Number of Atoms	substances	90

The analysis of the substance fields that are used most commonly (based on the usage of the existing API) shows that the substance registry number is highly overrepresented and used to search various record types. This is also and indication that many customer use the intermediate step of capturing identifiers to search/retrieve records from over contexts. This is another indication that the users will benefit from an optimized retrieval strategy.

Requirements (not limited to MVP)

API Capabilities

The capabilities and activities represent the tasks a API user wants to perform using the API. For the Scope of the MVP all capabilities above the red line shall be implemented.

Capabilities	Activities	Description	Associated Job Stories	Comments
Access comprehensive online			APIJS-001	

documentation				
Use SDK as tooling for development			APIJS-002	Descope for MVP
	Integrate Reaxys API in python application			
	Integrate Reaxys API in Java application			
Retrieve fields including those without value		User commonly retrieve entire facts, e.g. DAT for bioactivities. When retrieving a fact then the response data structure shall include also field names and None values for fields that are not populated.	APIJS-017	
Retrieve data in JSON format		JSON is a commonly used data format by API users and needs to be supported as	APIJS-022	

		default response format		
Execute a stateless session-free workflow		Reaxys API session creation is not required	APIJS-025	
Retrieve substances, associated bioactivities and targets with one request			APIJS-020	
	Search substances by exact structure search			
	Retrieve substance structure, identifications, and associated bioactivities and targets in one request			
Retrieve documents and associated substances		Note: Needs investigation if expanding join record as subnodes or a flat	APIJS-027	

with one request		representation is the better choice		
	Search documents by keywords (e.g. “nir” for near infrared)			
	Retrieve Abstract, Keywords, Substance Identifications, Structure, UV/VIS and IR Spectroscopy in one request			
Define the authentication and authorization configuration			APIJS-024	
Receive standard error codes and a descriptive error message			APIJS-026	

Search substances by structure using SMILES			APIJS-006	
Search reactions by structure using reaction SMILES			APIJS-007	
Support KNIME / Pipeline Pilot		This capability shall ensure that the API can be used with standard KNIME/PLP components for HTML, JSON processing	APIJS-003	
End Of MVP Requirements				
Search all Reaxys substance databases simultaneously			APIJS-014	
Retrieve incremental updates for reactions			APIJS-028	
Retrieve incremental			APIJS-029	

updates for bioactivities				
Search target names using taxonomy			APIJS-004	
Identify number of substances with solubility data and total number of measurements			APIJS-023	
Request data in tabular format			APIJS-005	
Retrieve substances as SMILES			APIJS-009	
Retrieve reactions as SMILES			APIJS-010	
Grant access to selected RCS supplier data			APIJS-013	

Limit data retrieval to specific record type (substances, reactions, citations, bioactivities, targets,etc.)			APIJS-015	
Limit data retrieval to this specific subset (e.g. bioactivities for a given target)			APIJS-016	
Retrieve substance properties aggregated by chemical uniqueness			APIJS-018	
Reference a specific reaction variation			APIJS-021	
Download search result as archive			APIJS-011	

Acess quick search			APIJS-030	
Understand the required SMILES format			APIJS-008	
performance batch download async			APIJS-012	
Expand with frequencies			APIJS-019	

Job Stories

The Job stories define when (Situation) a user wants to use the API to accomplish a specific goal (Expected Outcome).

The Motivation describes how the API shall be used. This translates into the required capabilities that need to be supported by the API.

The list of job stories will be extended until we reach feature completeness.

	Situation		Motivation		Expected Outcome	Capability
When	I get started using the Reaxys API or come back to	I want to	access comprehensive online API documentation	So I can	quickly onboard myself without the need of length	Access comprehensive online documentation

	using the API after not using it in a longer time and I am under time pressure to deliver results		ion and tooling		discussion with the Elsevier support	
When	I get started building applications using the Reaxys API	I want to	use SDK as tooling for my preferred programming platform	So I can	quickly prototype a solution	Use SDK as tooling for development
When	I retrieve results for further processing	I want to	to choose JSON as the format	So I can	avoid having to transform the clunky XML to JSON, which we use internally as a standard format	Retrieve data in JSON format
When	I implement an API workflow	I want to	execute a stateless session-free workflow	So I can	quickly focus on the important features of searching and retrieving data and do	Execute a stateless session-free workflow

					not have to deal with session resets when executing long-running workflows	
When	I retrieve bioactivities	I want to	retrieve all bioactivity fields including those that have no value	So I can	easily transform the response in a tabular format without having to know which fields do exist	Retrieve fields including those without value
When	I compile a bioactivity profile for a set of substances	I want to	retrieve associated bioactivities and targets in one request	So I can	avoid having to run multiple requests and do not have to know which fields to use for joining these different contexts	Retrieve substances, associated bioactivities and targets with one request
When	I need to identify substances and associated	I want to	be able to retrieve the documents and substances	So I can	simplify the workflow by reducing the number of calls and	Retrieve documents and associated substances

	properties for my literture search		in one request		I do not need to understand the concept of a "context switch"	with one request
When	I entitle a customer for API access	I want to	define the authentication and authorization configuration indepentently from the .com settings	So I can	simplify the entitlement setup and do not have to workaround with the \.com setup to avoid conflicts such as the choice screen	Define the authentication and authorization configuration
When	the API is running into an issue	I want to	receive standard error codes and a descriptive error message	So I can	understand what the root cause is and I can take appropriate actions if the issue is cause by myself	Receive standard error codes and a descriptive error message
When	I build API workflows in KNIME or Pipeline Pilot	I want to	have access to the same functionalities as available	So I can	fully leverage the API and do not have to bother myself	Support KNIME / Pipeline Pilot

			through the direct API access		thinking about workarounds	
When	I need to search Reaxys for chemical structures	I want to	use SMILES as input format	So I can	use our preferred format without having to convert our substance library using a tool I am not familiar with	Search substances by structure using SMILES
When	I need to search Reaxys for reactions using a chemical structure representation	I want to	use reaction SMILES as input format	So I can	use our preferred format without having to convert our reactions library using a tool I am not familiar with	Search reactions by structure using reaction SMILES
When	I need to identify if a substance structure is already known	I want to	search all Reaxys substance databases I have access to simultaneously by default	So I can	be sure I did not miss my substance and I do not need to know about the available databases	Search all Reaxys substance databases simultaneously

When	I need to update my reaction based model with the latest additions and updates	I want to	retrieve all reaction variations that have been released/updated after a given date and identify which part of the reaction has been modified for updates	So I can	create an increment for my currently available local data set and do not have to download the entire variations data set all over again	Retrieve incremental updates for reactions
When	I need to update my bioactivities based model with the latest additions and updates	I want to	retrieve all bioactivities that have been released/updated after a given date and identify which part of the measurment has been modified for updates	So I can	create an increment for my currently available local data set and do not have to download the entire bioactivity data set all over again	Retrieve incremental updates for bioactivities
When	a customer is interested in using Reaxys data for building a solubility predictor	I want to	identify number of substances with solubility data and total	So They	can assess if sufficient data is available to support the use case and it's	Identify number of substances with solubility data and total

			number of measurements		worth to subscribe the data set	number of measurements
When	I search for bioactivities by biological target name	I want to	use any commonly used target name as search term and retrieve results for all synomyns	So I can	avoid having to know about preferred Reaxys names for targets and can minimize the number of false positives	Search target names using taxonomy
When	I need to process larger amounts of Reaxys data	I want to	easily import the API results in tabular data processing tools such as pandas or Excel	So I can	avoid spending time on having to convert the response	Request data in tabular format
When	I request substance structures	I want to	retrieve them in SMILES format	So I can	use the response directly in our ML workflow without having to convert the substance using a tool I am not	Retrieve substances as SMILES

					familiar with	
When	I request chemical represenations for reactions	I want to	retrieve them in SMILES format	So I can	use the response directly in our ML workflow without having to convert the reactions using a tool I am not familiar with	Retrieve reactions as SMILES
When	I need to retrieve properties to profile a list substances	I want to	retrieve the substnace properties aggregated by chemical uniqueness	So I can	easily compile the substance profiles without having to deal with the duplicated molecules in Reaxys and the non-standardized chemical names	Retrieve substance properties aggregated by chemical uniqueness
When	I provide reaction examples to my colleagues	I want to	reference a specific reaction variation	So they	can easily pull the specific example including	Reference a specific reaction variation

	as suggestion how to synthesize a compound				the conditions via the API for further processing	
When	a customer retrieves RCS data	I want to	grant access to selected RCS supplier data only	So I can	open the subset that we are legally allowed to provide for redistribution	Grant access to selected RCS supplier data
When	a customer's subscription allows them to only access a specific Reaxys record type for the approved use case	I want to	limit the data retrieval to this specific record type	So I can	ensure that no additional data is downloaded and we are protected against loss of revenue	Limit data retrieval to specific record type (substances, reactions, citations, bioactivities, targets,etc.)
When	a customer's subscription allows them to only access a subset of a given record type for the	I want to	limit the data retrieval to this specific subset	So I can	ensure that no additional data is downloaded and we are protected against loss of revenue	Limit data retrieval to this specific subset (e.g. bioactivities for a given target)

	approved use case					
When	I identified a large data set I want to utilize locally	I want to	download the search result as archive	So I can	avoid to execute a long-running synschronous workflow	Download search result as archive
When	I run an API query	I want to	get the same results compared to using quick search	So I can	explore the content in the UI and then retrieve a consitent result set via the API	Acess quick search
When	I use SMILES or reaction SMILES as input to search Reaxys	I want to	understand the required SMILES format	So I can	avoid wasting time by having to figure out which SMILES defintiions are understood by the Reaxys structure search engine	Understand the required SMILES format
When	I need to access large amounts of	I want to	download the data in an easy way	So I can	avoid to build complex workflow to	performance batch

	reaction data				fetch the reaction and associated facts	download async
When	I explore the Reaxys content for available values of a field	I want to	easily page through all unique values and the number of occurences	So I can	create statistics that allow me to assess if the content is helpful for my use case	Expand with frequencies