Technical Architecture & Vision

Dr. Jay Kola
Clinical Architect
Information Standards Division
MOH Holdings Pte Ltd
Singapore

Date : 28/06/2011

Outline

- What is Snofyre?
- How to use Snofyre?
- Snofyre Technical Architecture
- How to use Snofyre API?
- Relevance to Implementers
- Snofyre Community

Snofyre – Vision

What is Snofyre?

Snofyre – Background

- Snofyre: SNOmed Framework for querY and REtrieval
- By-product of work carried out at NHS CFH, UK on querying of SNOMED CT coded data.
- How do we demonstrate end to end ‘querying’?
  - SNOMED CT in anger!
- SNOMED CT related documentation: 1000+ pages
- Translate this ‘something that works’!

What is Snofyre?

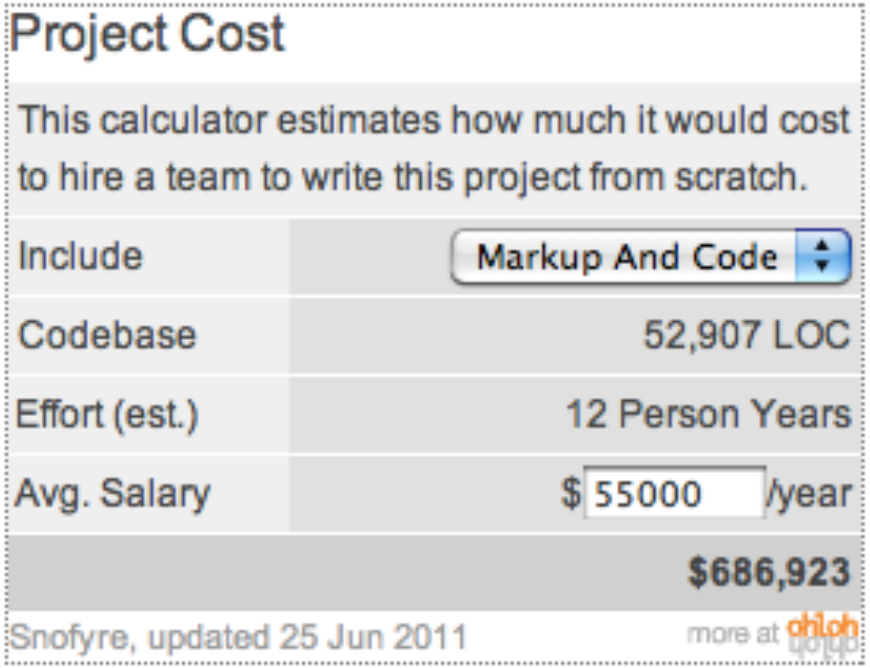
- Obvious Answer: A demonstrator application for querying SNOMED CT encoded data
- Snofyre Application: Query authoring and execution interface for
  - Authoring queries using SNOMED CT concepts (or expressions)
    - Refine queries using subsumption, exclusion, etc
  - Executing queries defined SNOMED CT concepts against pre or post-coordinated SNOMED CT encoded data.
- Snofyre API – Collection of generic services that implement
  - SNOMED CT content access services
  - Search and User interface related services
  - Query execution and resolution services

What is Snofyre? - 2

- Educational artefact
  - Example implementation of SNOMED CT related application services.
- A rapid prototyping application
  - Generates data for a given clinical example; for example you can easily generate patient records that contain Diabetes
- Snofyre is NOT:
  - A reference implementation for SNOMED CT
  - A production ready application

What is Snofyre? - 3

- Reflects an understanding of:
  - Terminology (SNOMED CT)
  - Software development
  - Clinical data analytics
- Some stats…



Lines of Code By Language

Language	Code Lines
Java	33,714
XML	18,787
HTML	364
DOS batch script	20
Groovy	16
shell script	5
MetaFont	1

Snofyre – Hands on

How to use Snofyre

Snofyre App Tasks

- Launch Snofyre
- Simple Query Creation & Execution
  - Create new query for patients who have Myocardial Infarction
    - Execute this query (Select Query Menu → Execute Query)
- Simple Query Creation, Save and Load
  - Create a new query for patients who have Diabetes
  - Save this query to a file called '*diabetes.xml*' on your Desktop. (Select File Menu → Save Query)
  - Clear this query, by clicking on the 'New Query' button.
  - Now Load the stored query from the file on your Desktop. (Select File Menu → Load Query from file)
- Complex Query Creation and Execution
  - Create new query for patients who have Myocardial Infarction with 'Severity' attribute value 'Mild'
    - Execute this query

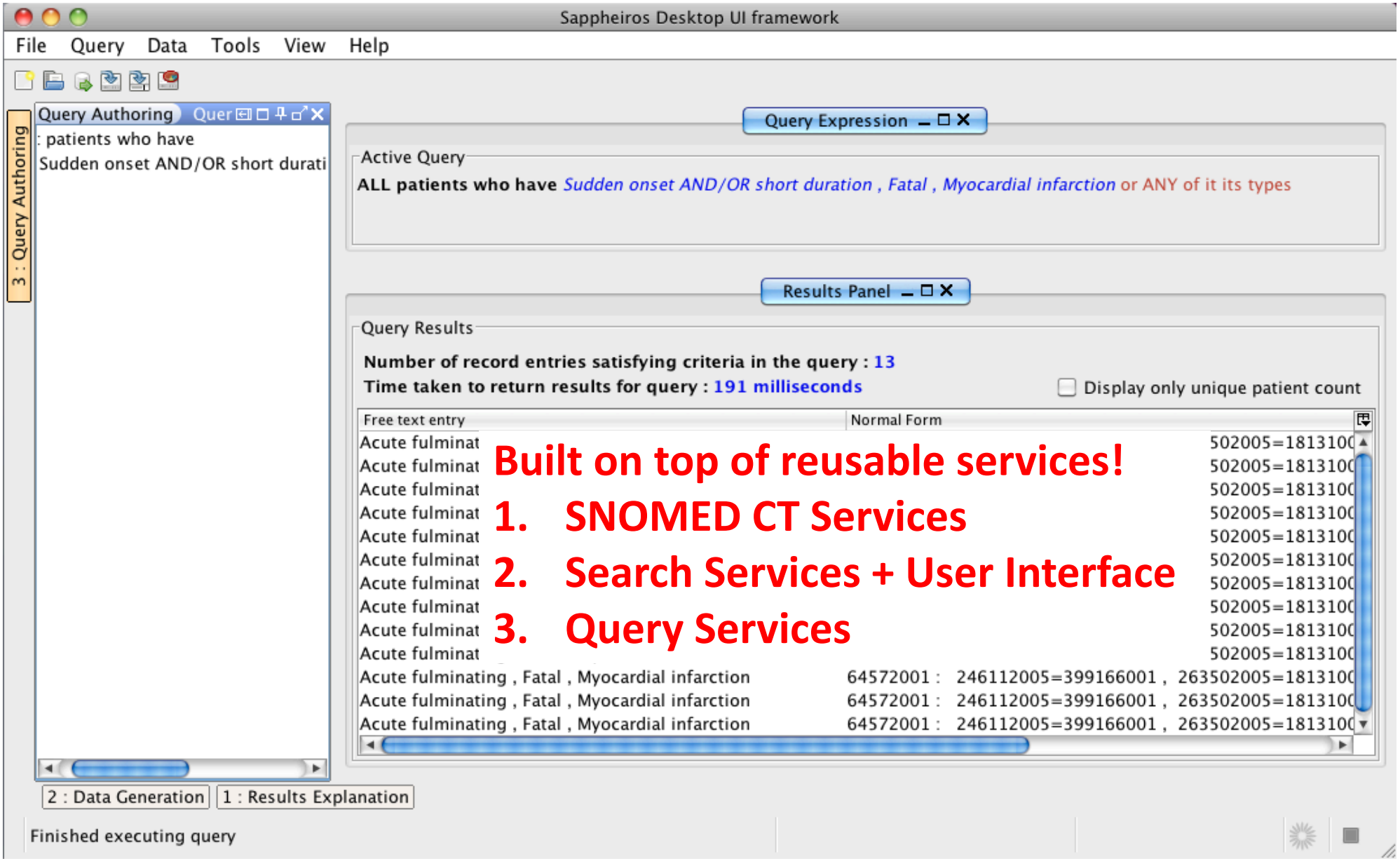
Snofyre App Tasks - 2

- Logical Query Creation and Execution
  - Create new query for patients who have Myocardial Infarction **AND** Asthma
    - Execute this query
  - Load the existing query for patients who have Myocardial Infarction **OR** Asthma (MI-or-Asthma.xml file in *'queries'* folder)
    - Execute this query
- Data Generation
  - Create a query for patients with 'Myocardial Infarction'
  - Generate 10 records that contain MI or any of its types
  - Store this query to your Desktop
- Exporting Query Execution Results
  - Load the previously stored 'Myocardial Infarction' query.
  - Execute the 'Myocardial Infarction' query from previous example; verify results returned
  - Export the results to a file on your Desktop

Snofyre – Technical Architecture

What's under the hood?

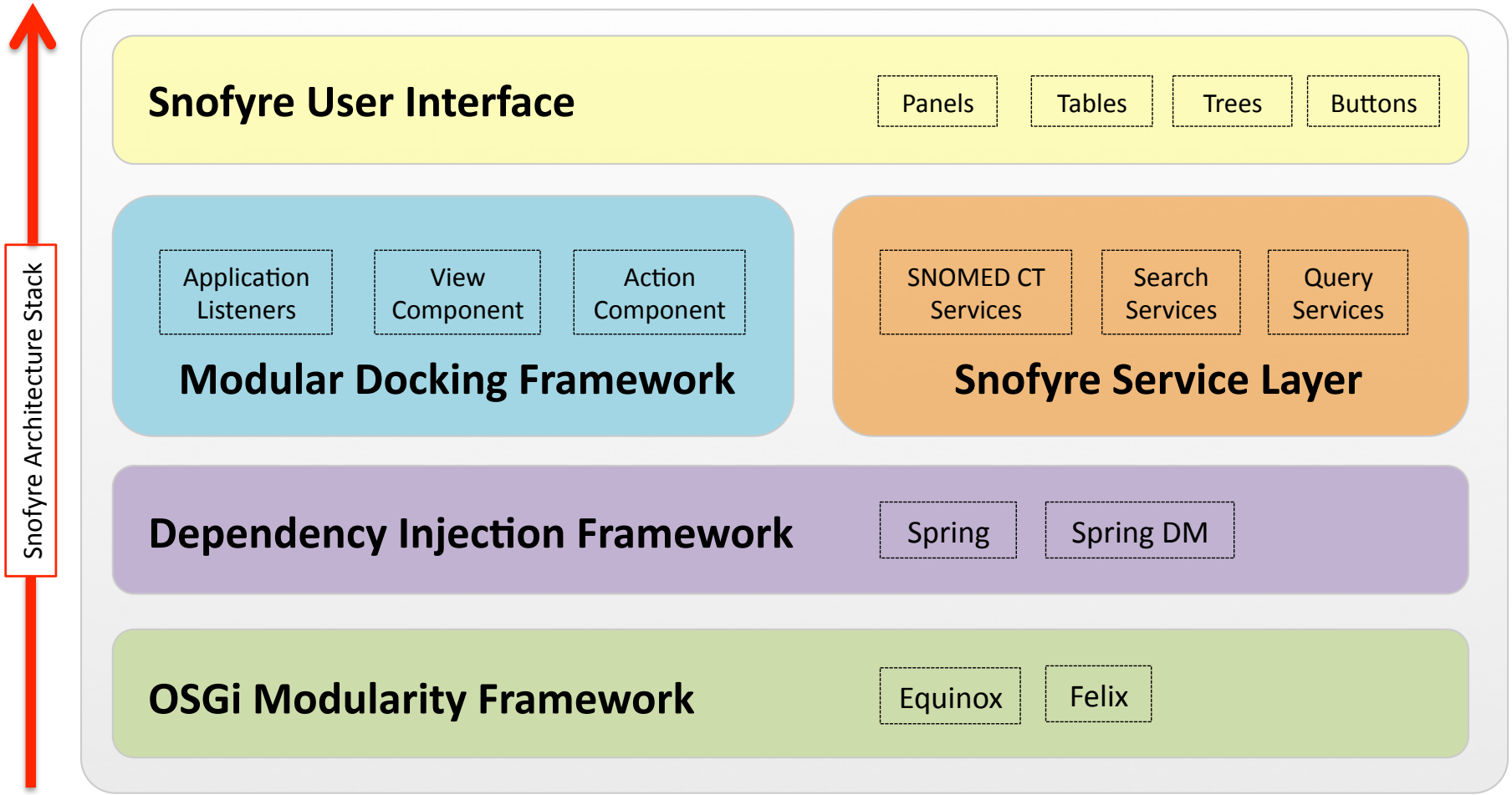
Snofyre - Overview



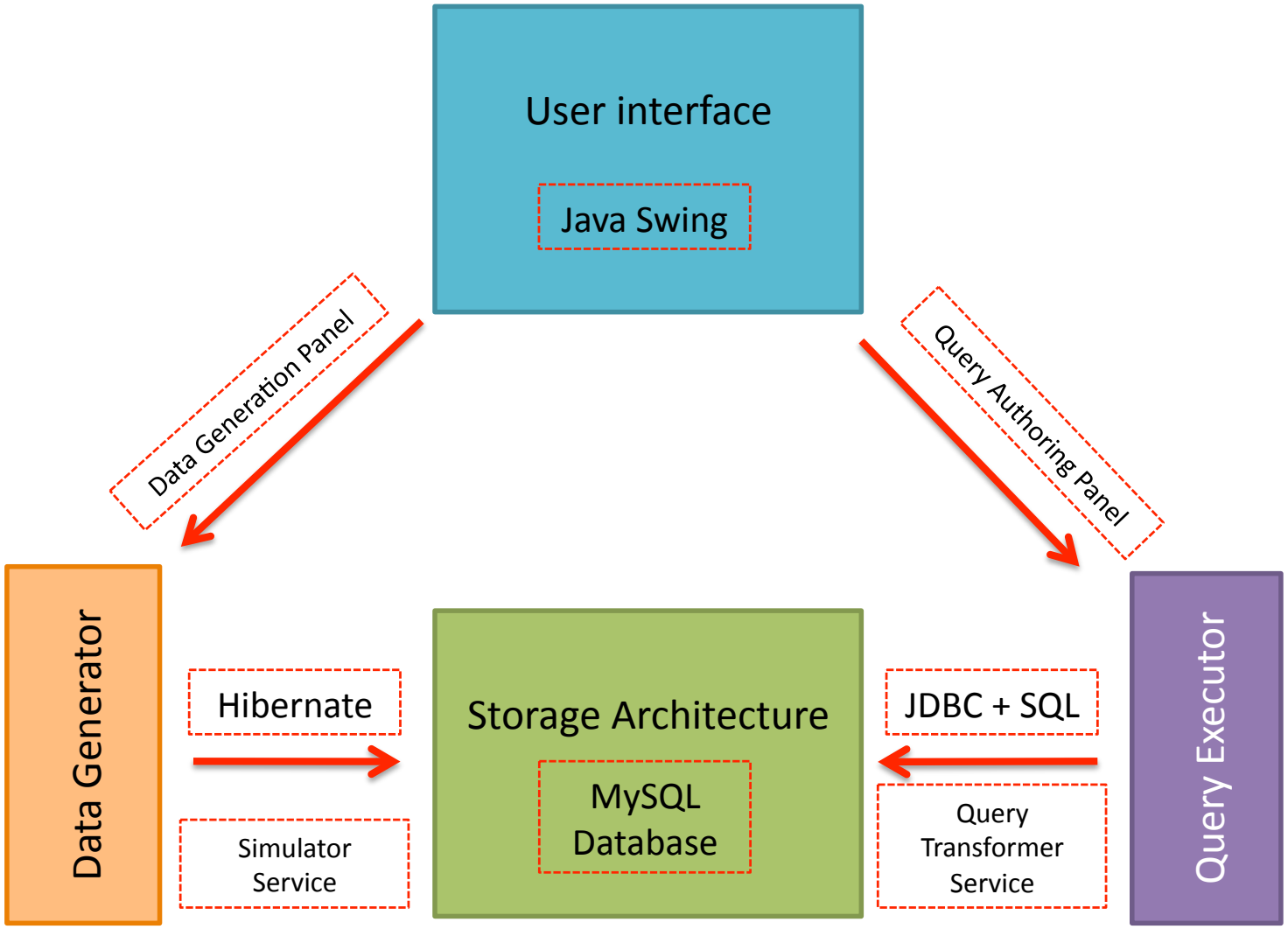
Snofyre - Overview

- When you refer to Snofyre; you could refer to
  - Snofyre Application
  - Snofyre API
- Application - Modular, OSGi Java application
  - Eclipse's framework
  - Easy to integrate into an Eclipse RCP
- API - Service oriented architecture
  - Services are functional units
  - OSGi services : Eclipse equinox services
  - Can be redeployed as a web services (web app)
- Dynamic behaviour
  - OSGi + Spring Dynamic Modules
- Frameworks used
  - Hibernate, Spring, Spring DM, Apache libraries…

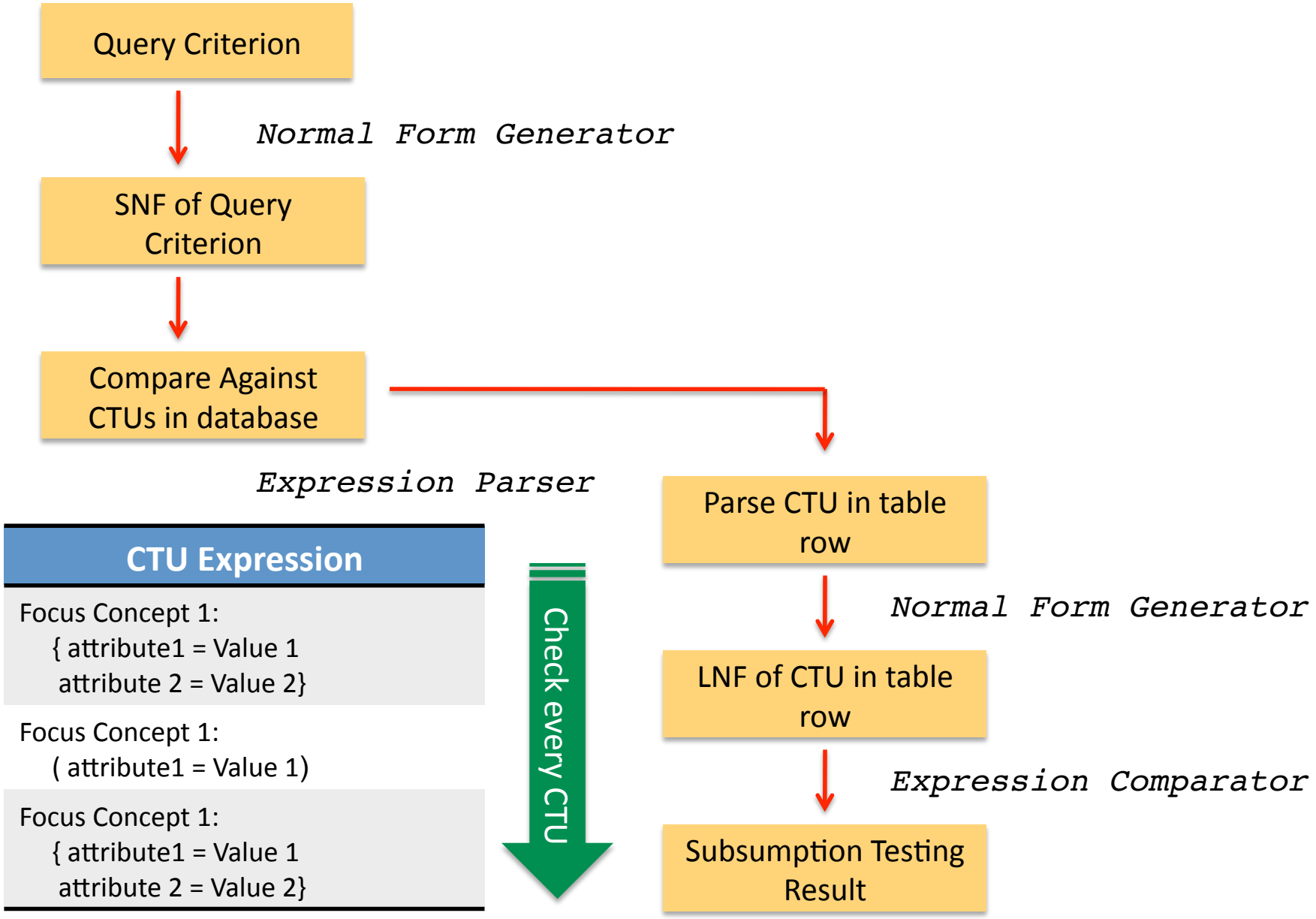
Application Stack Overview



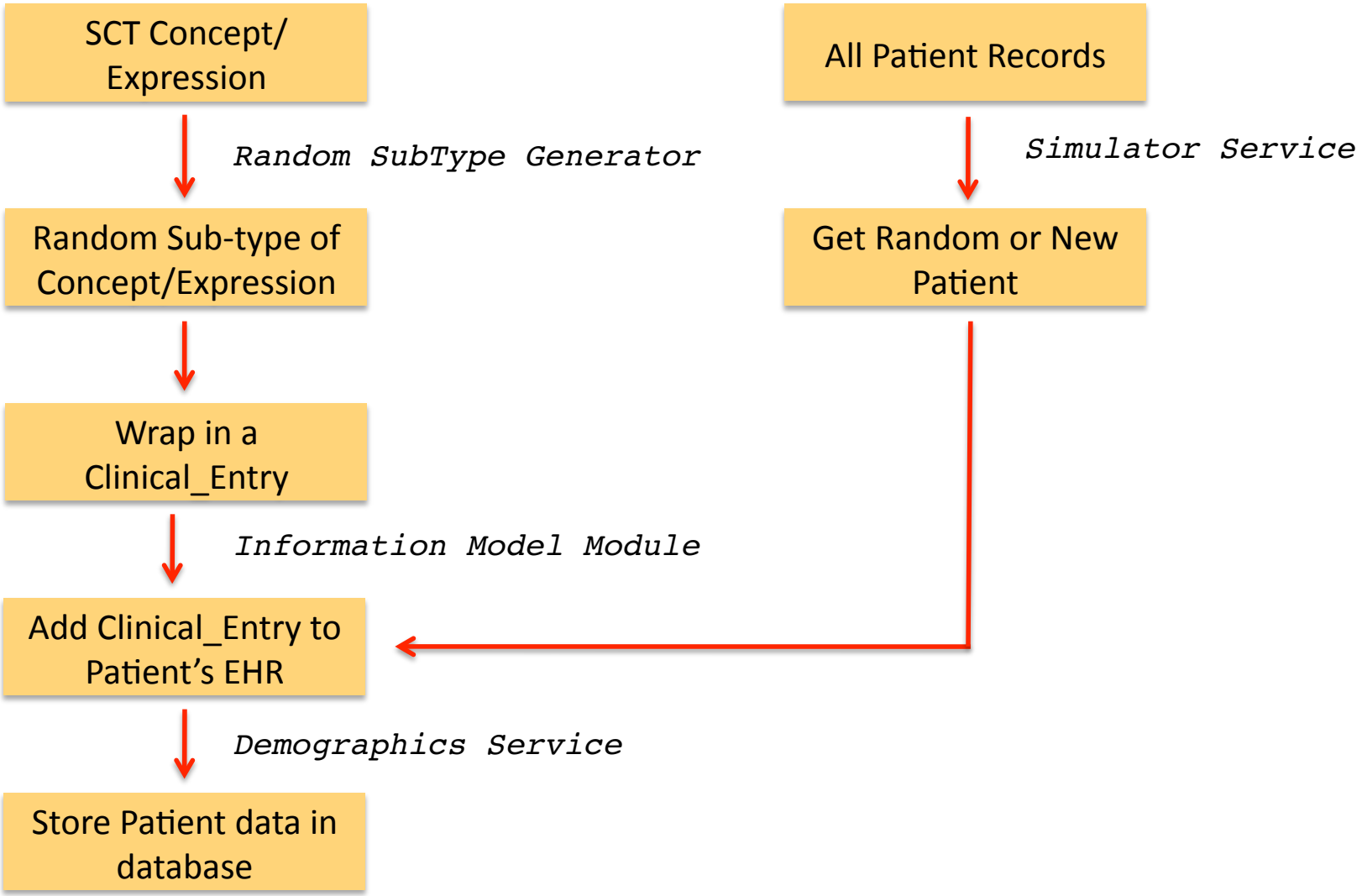
Data Flow Overview



Query Execution Process



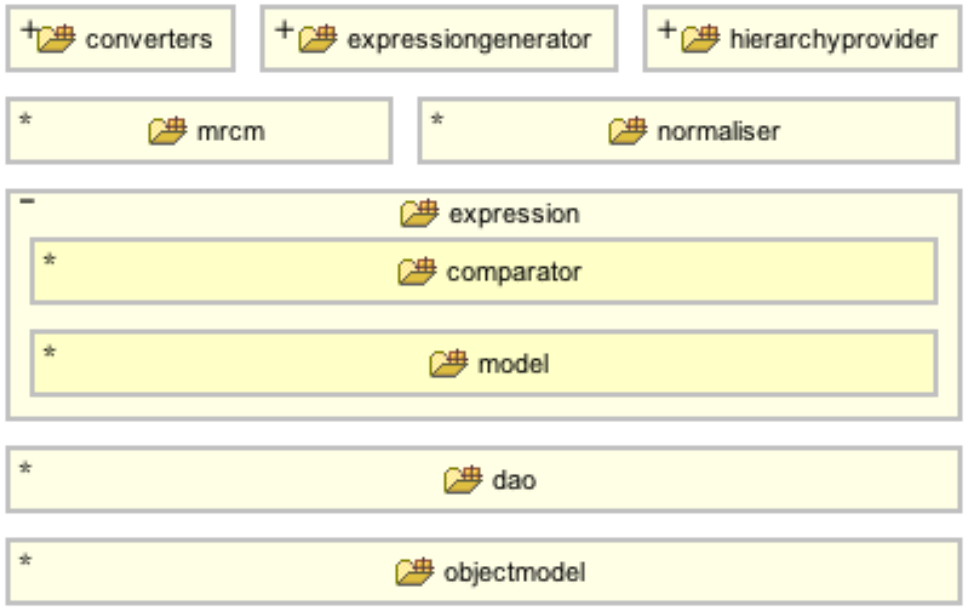
Data Generation Process



Snofyre- Functional Overview

SNOMED CT Service Layer

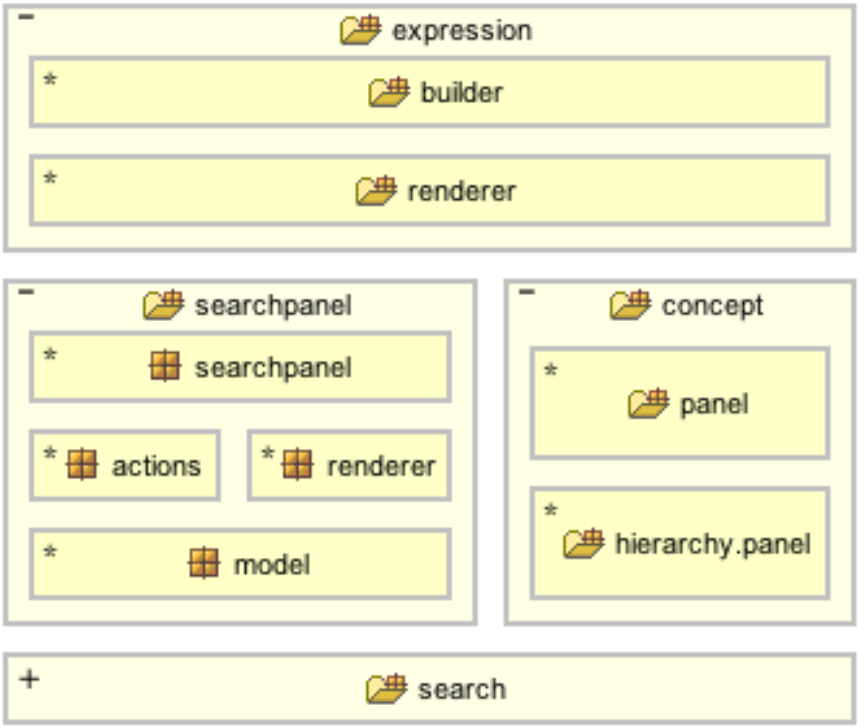
- Represent SNOMED CT
- Access to SNOMED CT
- Represent SNOMED CT post-coordinated expressions
- Generate Normal Forms
- Compositional Grammar Converter (+ parser)
- Compare post-coordinated expression
- Sanctioned attributes - MRCM



Snofyre - Functional Overview 2

Search and User Interface Layer

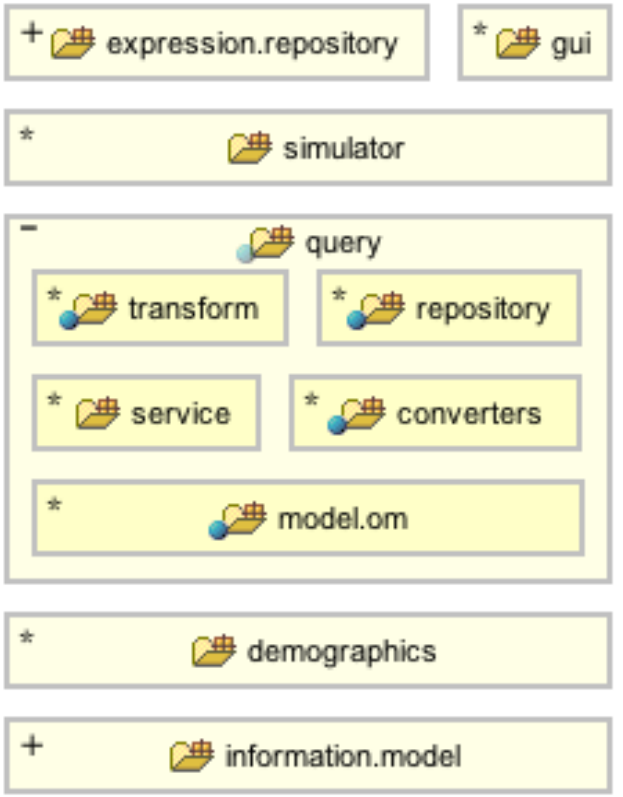
- Search Service for SNOMED CT terms
- Search interface (with filters) + results
- Render terms and concept hierarchies
- Render and build post-coordinated expressions



Snofyre - Functional Overview 3

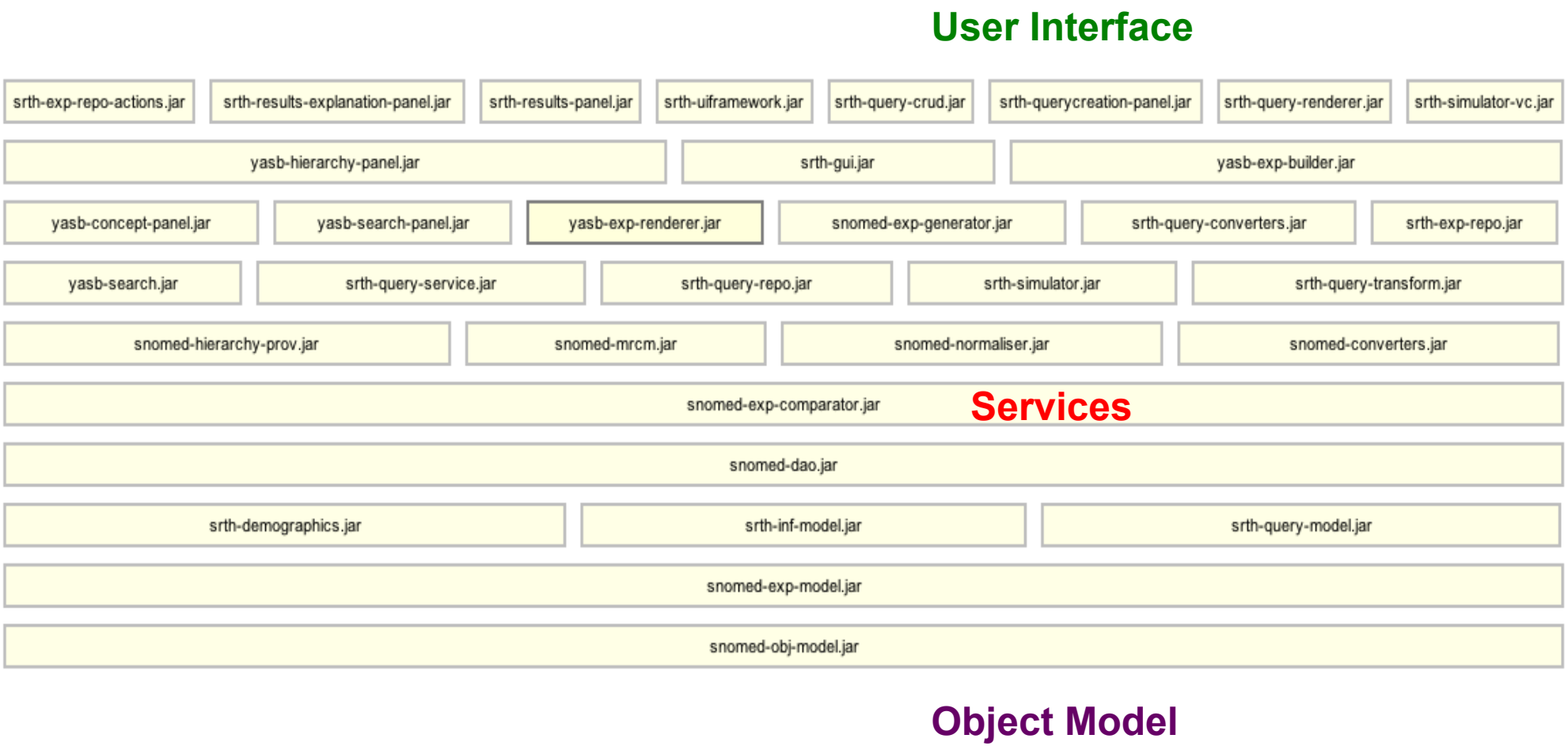
Application Layer

- Represent queries (? Metadata)
- Author queries
- Human friendly rendering of queries
- Store and access queries
- Information Model (+ demographics)
- Query Transform (to SQL) and Execution
- Data generation



Snofyre - Functional Overview 4

Entire Application Stack



Snofyre API – Hands on

How to use Snofyre API

Snofyre API Tasks

- To do this task, you'll need to be 'SNOMED CT Application Developers'!!
- Install Maven
- Install Subversion
- Check out Snofyre Code from the Snofyre Repo
- Compile using mvn clean install
- Now create a class 'Hello Services'
- Import uk.nhs.cfh.dsp.snomed.dao.TerminologyService….
- So and so forth…

Snofyre API Tasks - 2

- To do this task, you'll need to be 'SNOMED CT Application Developers'!!
- ~~Install Maven~~
- ~~Install Subversion~~
- ~~Check out~~ **Don't need all this for this task!** ~~Repo at~~
- ~~Compile us~~
  1. **We can use the bundled Scripting console!**
  2. **Easy peasy!**
- ~~Now creat~~
- ~~Import uk.nhs.cfh.dsp.snomed.dao.TerminologyService….~~
- ~~…. So and so forth…~~

Snofyre API Tasks - 3

- To do this task, you'll need the Snofyre Scripting Console
- Tools → Scripting Console
- Wait…
- Minimise Snofyre

Snofyre API Tasks - 4

- In the console, type
  - Println ‘Hello Snofyre’;
  - Press the Enter Key
- Type
  - def concept = terminologyDAO.getTerminologyConcept (“22298006”);
  - println concept.getFullySpecifiedName();
- Now some more advanced examples…
  - Get the normal form of Myocardial Infarction

Snofyre API Tasks - 5

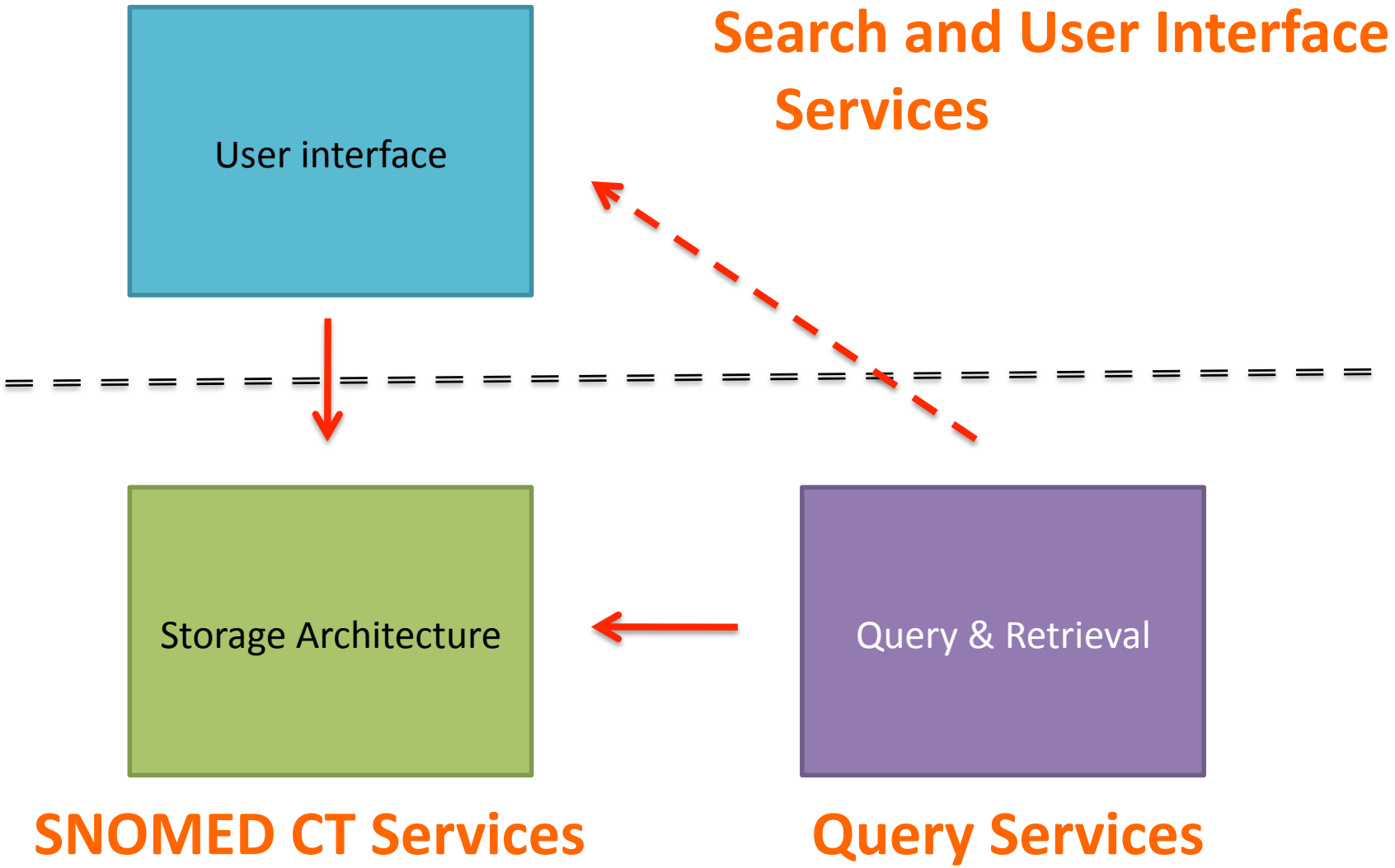
- Services look up table…
- Refer to Snofyre Functional Services Manual
- Who'll be the first one to get all children and descendants of Myocardial Infarction (22298006)?

Service Name	Service Description
terminologyDAO	Access to SNOMED CT content
normalFormGenerator	Generates Normal Forms for Concepts/Expressions
hierarchyProvider	Provides hierarchy look up for SNOMED CT concepts
cgfConverter	Converts a concept/expression into its compositional grammar form
situationWrapper	Wraps concept/expression in a Situation With Explicit Context wrapper
randomSubtypeGenerator	Returns a random subtype for a given concept or expression

Snofyre – Relevance to Implementers

What can SNOMED CT implementers expect from Snofyre

How can I use Snofyre?



How can I use Snofyre? - 2

- Educational artefact
  - Understand how SNOMED CT content is organised…
  - Example implementation of SNOMED CT related application services.
  - Can explore SNOMED CT search functionality
  - Rendering of expressions + concepts
  - Understand downstream effects of authoring or modeling
- A rapid prototyping application
  - User engagement; end to end demonstration
  - Generates data for a given clinical example; for example you can easily generate patient records that contain Diabetes (or any of its types) + Ulcer of lower limb (including any of its types)

How can I use Snofyre? - 3

- Is Snofyre a terminology server?
  - Yes an incomplete implementation, if Terminology Server = Set of functional services to manipulate SNOMED CT
  - No, if you want a ready to use 'out of the box' product
- How does Snofyre relate to CTS 2?
  - CTS 2 is a draft specification
  - Snofyre only covers terminology related services – no messaging services
  - Focused on SNOMED CT – not part of CTS 2?
- Can I use Snofyre to create subsets/refsets?
  - No, Snofyre was not created to be an 'authoring' tool
  - Yes, Snofyre does create subsets/refsets but does not have a way of exporting them out of the box
- Does Snofyre support my favourite feature?
  - Please ask us on the community site

Snofyre community

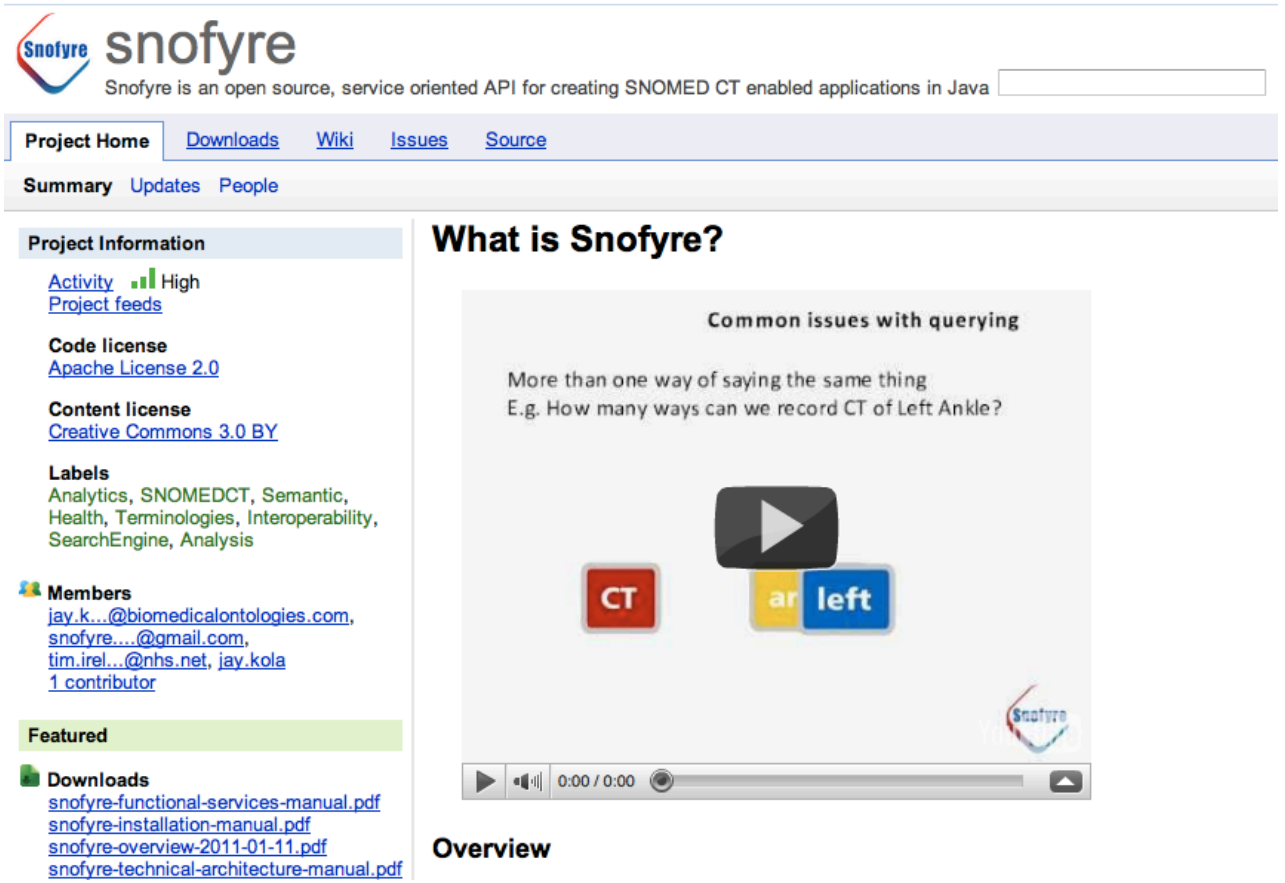
What to expect and how to contribute

Where to find the goodies?

- All project output available:
  - http://code.google.com/p/snofyre
  - Licensing
    - Code: Apache 2 License
    - Documentation: Open Government License
- Resources available
  - Source code
    - Snofyre Application
    - Snofyre API
  - Documentation
    - Snofyre Functional Services Manual
      - http://snofyre.googlecode.com/files/snofyre-functional-service-manual.pdf
    - Snofyre Installation Manual
      - http://snofyre.googlecode.com/files/snofyre-installation-manual.pdf
    - Snofyre User Manual
      - http://snofyre.googlecode.com/files/snofyre-user-manual.pdf
    - Snofyre Developer Manual (not yet available…)

Where to find the goodies? - 2

- And here are the actual goodies… ☺
  - http://code.google.com/p/snofyre



How to contribute?

- User Discussions
  - http://groups.google.com/group/snofyre-users
- Developer Discussions
  - http://groups.google.com/group/snofyre-dev
- Reporting Issues
  - http://code.google.com/p/snofyre/issues/list
- Please join the community… ☺
  - Open for all to join!

- Credits & Acknowledgements
  - Department of Health Informatics Directorate, UK
  - MOH Holdings Pte Ltd, Singapore

Email: jay.kola@biomedicalontologies.com

Thank you!

Questions?