



SOA at GSK

Working in a mixed technology
environment

Brett Hiemenz, R&D IT

Agenda

- History of **Service-Oriented Architecture (SOA)** in GSK Chemistry IT
 - What we've learned
 - Current status
- Web service case studies: **property calculation, structure search and structure format translation.**
 - Issues each raised by mixed technology environment (hardware, platform and structure representation)
 - Examples of problems (Formats, batching, cross databases etc)
- Mixed database technology federated under web services
 - Examples of problems and current status.
- What's next for SOA?

Services Oriented Architecture at GSK

Historical Perspective 1

- Web Services in Chemistry began at GSK at merger between Glaxo and SmithKline Beecham.
- Around 2000, a proof of concept ran to show it was possible to retrieve data from the two companies legacy systems using web services.
- **Chemistry Information Service** was born, then rapidly and iteratively improved.
 - New methods added for **structure search** and **structure formatting**
 - Data returned in the format the user needed: **SMILES**, **mol** or **chime** string.
 - **CIS2** introduced formal SOAP and improved performance.
 - At its peak CIS2 provided approx **500,000** id conversions per day.
 - **CIS3** moved bespoke C++ code to Java web services stack

Services Oriented Architecture at GSK

Historical Perspective 2

- CIS has served numerous clients for nearly 10 years
 - **Chemfido** was among the first clients to fetch data from CIS1
 - Chemfido evolved into **Chemretriever** (finally retired in 2009).
 - **Chemically Aware Spreadsheet** uses all the current CIS3 methods and is still on the current desktop.
- GSK saw many benefits from the CIS web service experiment...
 - CIS **federated structure searches** across Thor and MDL ISIS/Host databases on VMS and Unix
 - Structure normalization rules differed between legacy companies, so services delivered a **common structure format** to the desktop.
- As well as the growing pains of web services
 - XML standard was still evolving, GSK's own standard was used
 - SOAP standard not yet available, used **custom XML**

Services Oriented Architecture at GSK

Historical Perspective 3

- CIS1 moved to CIS2 which was SOAP compliant. Also redirected CIS1 to CIS2 to avoid remediation of legacy clients.
 - Re-pointing services to maintain backwards compatibility has caused problems in retiring legacy services
- **Property Information Service (PIS)** also started around 2000 for simple property calculation from structure.
 - **Modularity** was introduced to add new properties from any vendor (ACD labs properties, CLogP/CMR from Biobyte, various toolkits to calculate SMARTS based properties.
 - Decision to put all responses for a property (e.g. CMR) in a single XML tag based on original limitations still needs to be remedied.
 - Many clients are still using this service.

Services Oriented Architecture at GSK

Lessons from history 1

- **Simple responses** most successful at first.
 - Id to structure and substructure search features were enabled early but not heavily used until better standards emerged.
- Reliable SMARTS input was tricky
 - No good **SMARTS to molfile converters** when we began.
 - Conversion from a SMARTS to a molfile query was written in-house to give the expected behaviour. (in Fortran!).
 - Rendering of SMARTS is an issue to this day.
- **Granularity** of services needs careful thought.
 - Domain level granular pieces of workflow are best.
 - Not atoms-bonds and not 'big picture' ideas.

Services Oriented Architecture at GSK

Lessons from history 2

- ‘Overselling’ of web services as ‘the solution’ based on examples that worked well led to inappropriate use. Over time the boundaries of what works well and what does not have moved.
- **Complex input** and **iterative workflows** did not lend themselves to Web services. User groups chose to use ‘faster’ bespoke code.
 - **Library enumeration** and **library design workflow** web services were deemed too complex.
 - A **multi-objective optimisation service** has very complex input but is still in use.
 - Very large data response for **HTS data** too slow. XML results were **too bulky**, and new methods that pass by reference were not adopted.

The SOA situation in 2009?

- Multiple web services exist, some deprecated, with **multiple vendor technologies** under them.
- Reliance on web services means they had to be robust. Issues with single point of failure and **load-balancing** across servers were resolved via **F5 hardware**.
- Use of IBM for **Service-Oriented Architecture Registry (SOAR)**, allowing services to be 'discoverable'.
- Web service infrastructure now serves a variety of client technologies.
 - **Workflow** (InforSense and PipelinePilot)
 - **Thin Clients** (ASP pages, JSP pages, CGI pages)
 - **Thick Clients** (Excel macros, Java and .NET apps, COM components)
 - **Scripting Clients** (Perl, Python, Jython, Groovy.....)

Some current key chemistry web services at GSK

Service	Function	Hits / Day
Chemistry Lookup	any id to a structure in specified format	8000
Structure Search	searching both against databases and in defined lists	100
Structure Format Translation	convert anything to anything	11,000
Inventory Query	how much of this is available	1300
Simple Property Calculation	1:1 property responses	10,500
Predictive Model Calculations	complex calculation response	150
Compound and Lot Registration	standardized access to compound and sample registry submission	10
CIS2	Lookup, search, translation. Slated for retirement next month	5,000

These services deal with multiple vendor technologies behind the scenes which still leads to issues with standardizing responses

A closer look at three examples

How multiple technology and multiple vendor issues have been resolved with web services in...

- Structure Format Translation
- Property Calculation
- Structure Search

Structure Conversion via Services

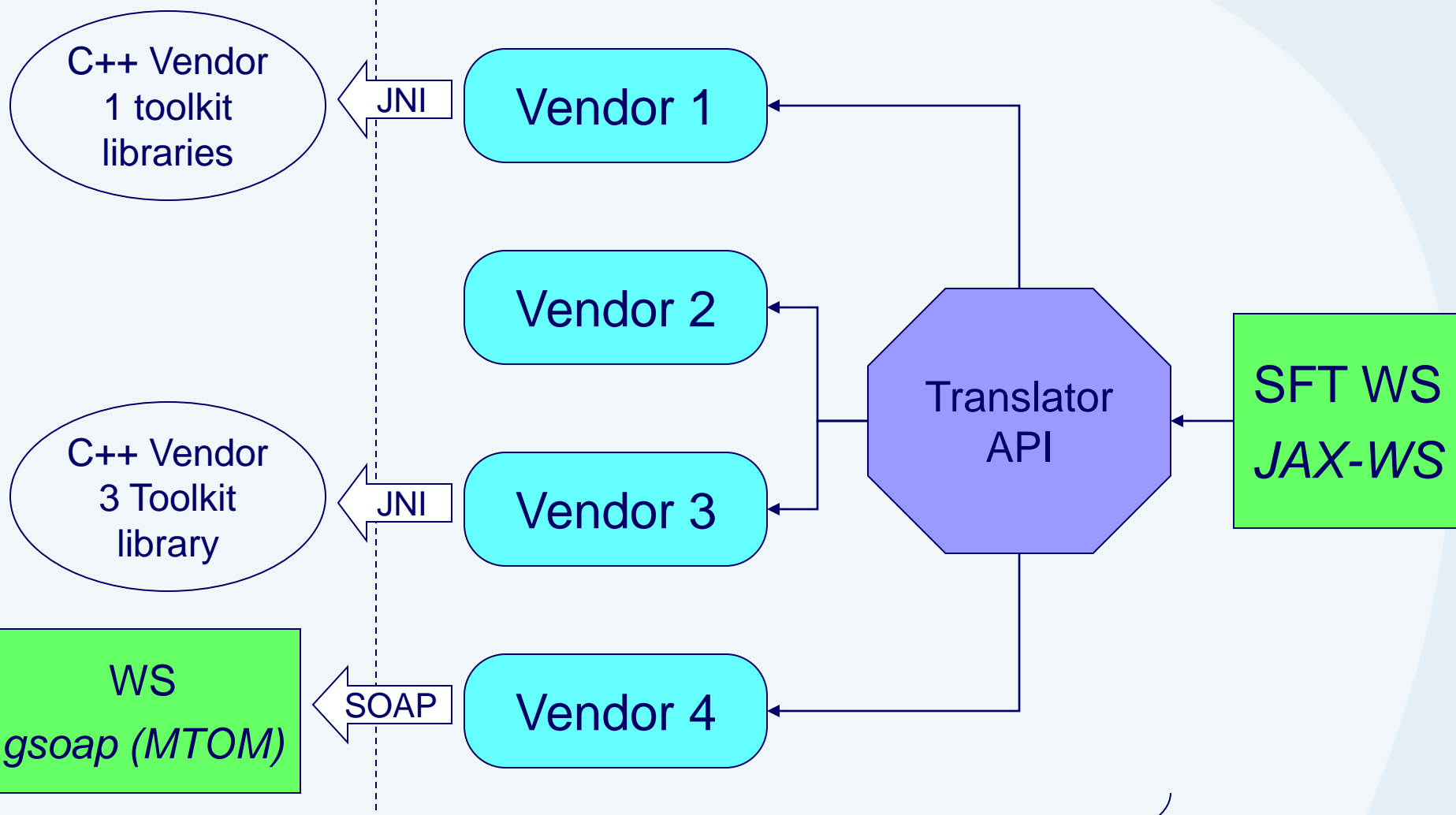
- Originally, chemistry format conversions were performed by a **collection of services**.
- Often the same conversion could be carried out by **multiple vendor algorithms**.
- Many common conversions actually required **multiple service calls**
 - Chime → SMILES = Chime → molfile → SMILES
- Conversions only produced **text-based structure** formats, not binary

Structure Format Translation Service (SFTS)

Rationalize the collection of services and vendor solutions

- Implement a **single service** for all structure translations.
- Set a **preferred default** vendor when multiple vendors can do the same conversion, e.g.
 - SMILES → Molfile V2000
 - SMILES → Molfile V3000
 - Molfile → SMARTS
- Allow clients to optionally specify a different vendor from the default.
- Provide capability for **multi-step conversions**, which can be automatically solved based on the preferred vendor profile.
 - Chime → SMILES
- Support binary **image output formats** in addition to strings

SFTS Architecture



Reused in Chemistry Lookup and Structure Search

SFTS Today

Resulting service is both **backward compatible** and **upgradeable**

- Conversion algorithms get upgraded without requiring client changes
- Clients can maintain **vendor continuity** if needed
- New vendor conversions can be **added as plug-ins** detected at server startup
- Errors are handled cleanly and can be passed through multi-step conversions

Property and Model Calculation via Services

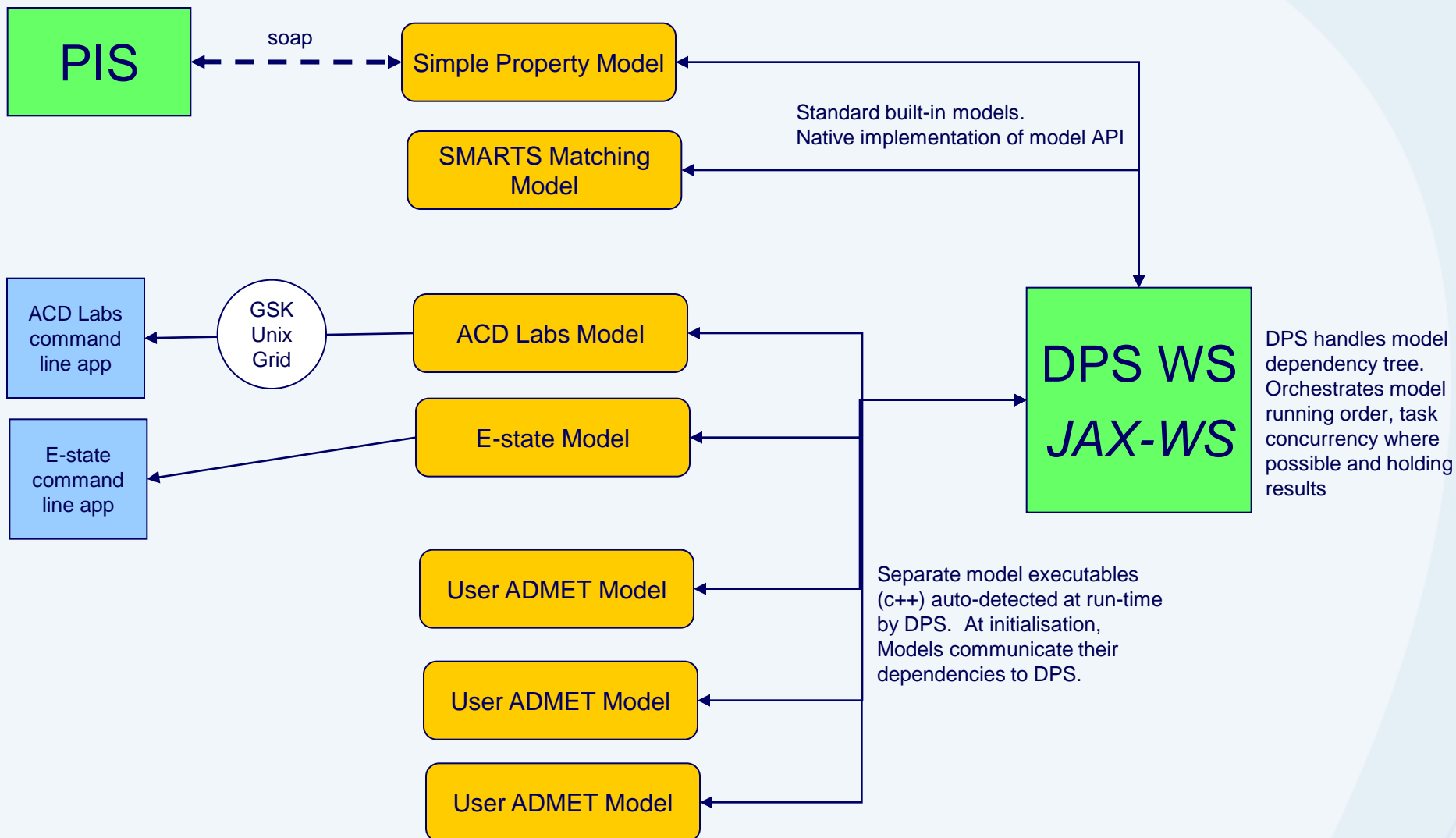
- Scientific groups use a multitude of products to generate ADMET models that need to be delivered to the scientists' desktops.
- Many models use properties from more than one vendor/technology in the algorithm.
- Scientists need a simple single interface to **all** the properties vendors can produce and the models scientists design.
- Some models are computationally intensive, so the service interface must support time-consuming calculations.

Derived Property Service (DPS)

Remove complexity of delivering vendor based properties by wrapping them all under a single service.

- **Science groups own the content** created using a range of vendor products.
 - Service framework is in Java, but models are in many languages
 - Many vendor algorithms are also embedded in the models.
- A **model deployment tool** allows scientists to control content of the web service without any need for IT intervention.
 - Scientists can access all the vendor properties needed for their model without any need to understand the vendor interface.
 - Option to version properties used or default to latest, *but* leads to some lifecycle management issues
- Model governance and lifecycle issues are in the hands of science groups.

DPS Architecture



DPS Today

DPS developed into an **asynchronous** web service.

- Asynchronous methods solve issues with **timeouts** from long-running calculations
- **Load-balancing** across multiple hardware servers helps reliability and performance, but causes issues retrieving data from a particular server
- **Cookie-based persistence** is used, but that requires client to conform to standard and reduced performance gain from load balancing
- Latest services use **database to persist job state** to allow any server to retrieve results or get status (particularly important in substructure)

Substructure Search via Services - Goals

- Searches should have a single interface, regardless of underlying database technology.
 - Searches should accept queries in SMARTS or molfile format.
 - Searches should allow users to specify the result format
- Response should be quick, no more than 3-4 seconds.
- Result matches must be consistent with expectations.

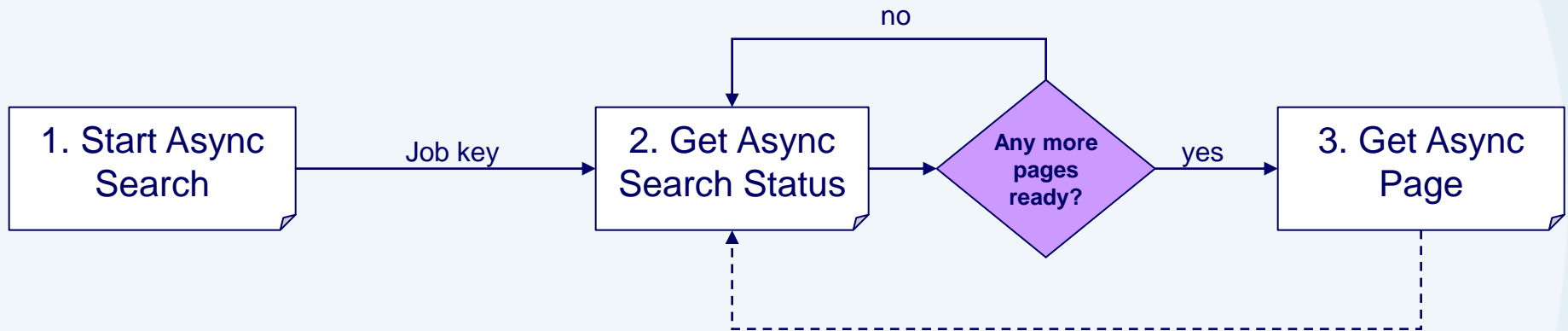
Substructure Search via Services - Obstacles

- Large result sets took too long to return, leading to HTTP timeouts.
 - Service did not respond until **all** results were obtained.
 - Searches across multiple databases were **sequential**.
- Search queries and result formats needed to be adaptable to users' needs, and not dependent upon underlying search technology.
 - Expected search results differ depending on what drawing tool (molfile, SMILES, SMARTS) is used to create the query
 - Users may prefer the results in a format different from what is native to a given cartridge technology.
 - Users expect chemically equivalent drawings to return consistent data (tautomerism, aromatic/non-aromatic, charge-separated, etc.)
 - Molfile → SMARTS conversion not easy. SMARTS → molfile still not viable.

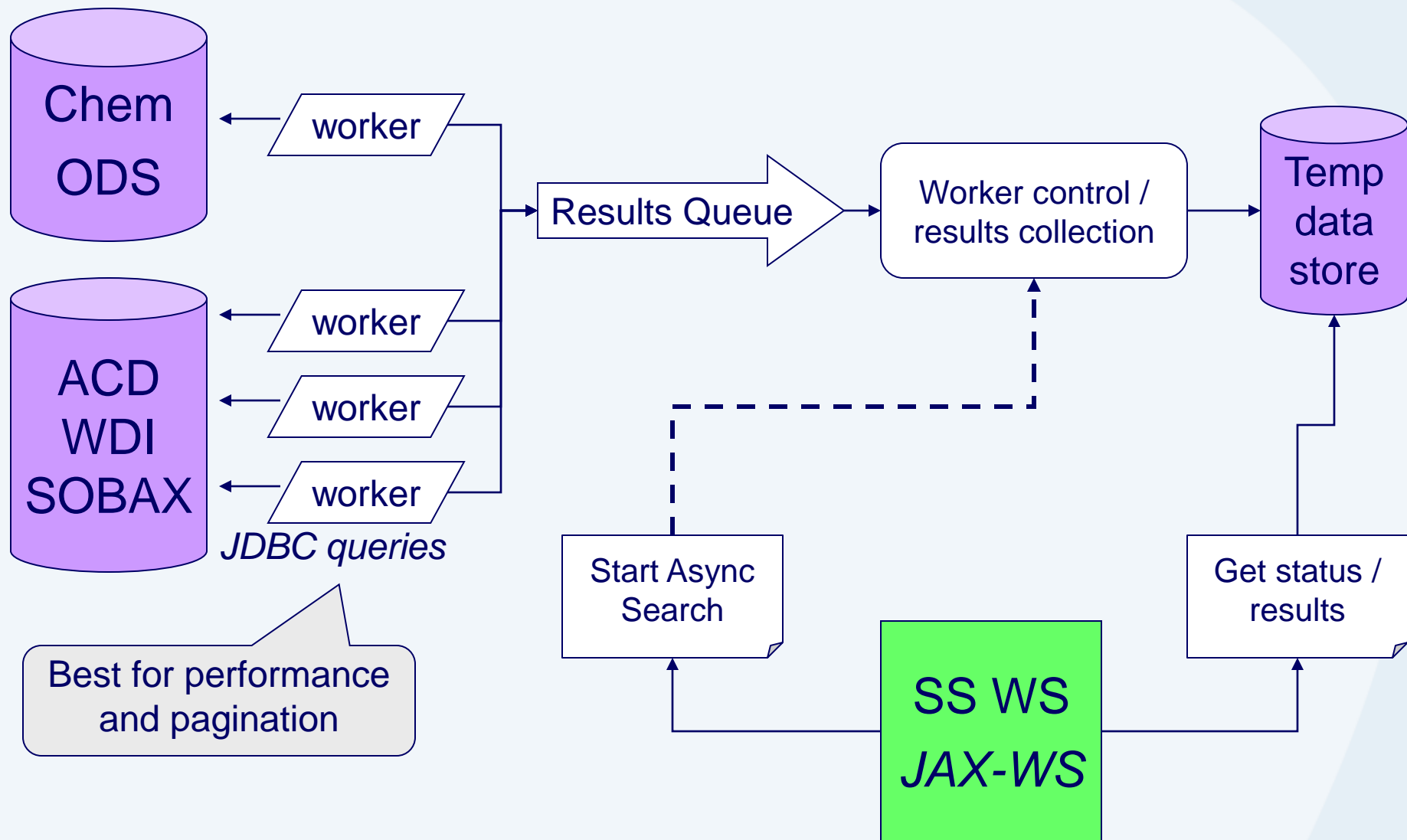
Structure Search Service (SSS)

- Users agreed on a set of set of search behaviors
 - All data sources can return molfiles **or** SMILES.
 - But queries are performed with SMARTS, not SMILES or query molfiles.
 - SMARTS queries are adapted (by conditionally applying terminal atoms, implicit/explicit hydrogen atoms, etc.) so results more closely match those from the database vendor technology, e.g. MDL.
- SSS employed several strategies to improve service responsiveness
 - Make all similarity and substructure search calls **asynchronous**, preventing timeouts.
 - **Return partial results** before the search is complete.
 - **Paginate results** to keep XML response size manageably small.
 - Provide estimated time and number of hits for full results.
 - Federate search across multiple databases in **parallel**.
 - **Persist search results** for 24 hours, saving time on repeat queries.

SSS Asynchronous Search and Pagination



SSS Architecture



Cross Technology Database Issues

Multiple technology exists in the underlying databases at GSK. There still doesn't appear to be a single solution which solves all issues.

Performance

- Vendor specific technology tends to perform faster.
- Oracle data cartridges require fine tuning to get good performance.
- Different vendor data cartridges have different performance profiles.
- Multiple data cartridges may be needed on the same data set.

Flexibility

- Federation across multiple databases leads to user issues when different technology handles the same query differently.
 - Results sets come back from different technologies and are merged, which can lead to inconsistency (especially for heterocyclic aromaticity interpretation)

What is next for SOA?

SOA is now fully embedded in Chemistry and GSK architecture

- More hosted services/algorithms/data to plug in.
- Need for better middle layer data federation and data cleansing tools.
- Security and externalization standards need to be applied across services

Some final issues

- Despite fully conforming with WS-I standards, not all web service clients can cope with GSK web service WSDLs
- Necessary updates to WSDLs have prevented GSK from realizing the full benefit of uplifting the services without changing the client.
- SMARTS are the standard for queries, but there is no good drawing tool for them.
- As the SOAR gets bigger it is harder to understand.
 - Need a service to orchestrate web services corporate-wide
 - Need to 'type' data better

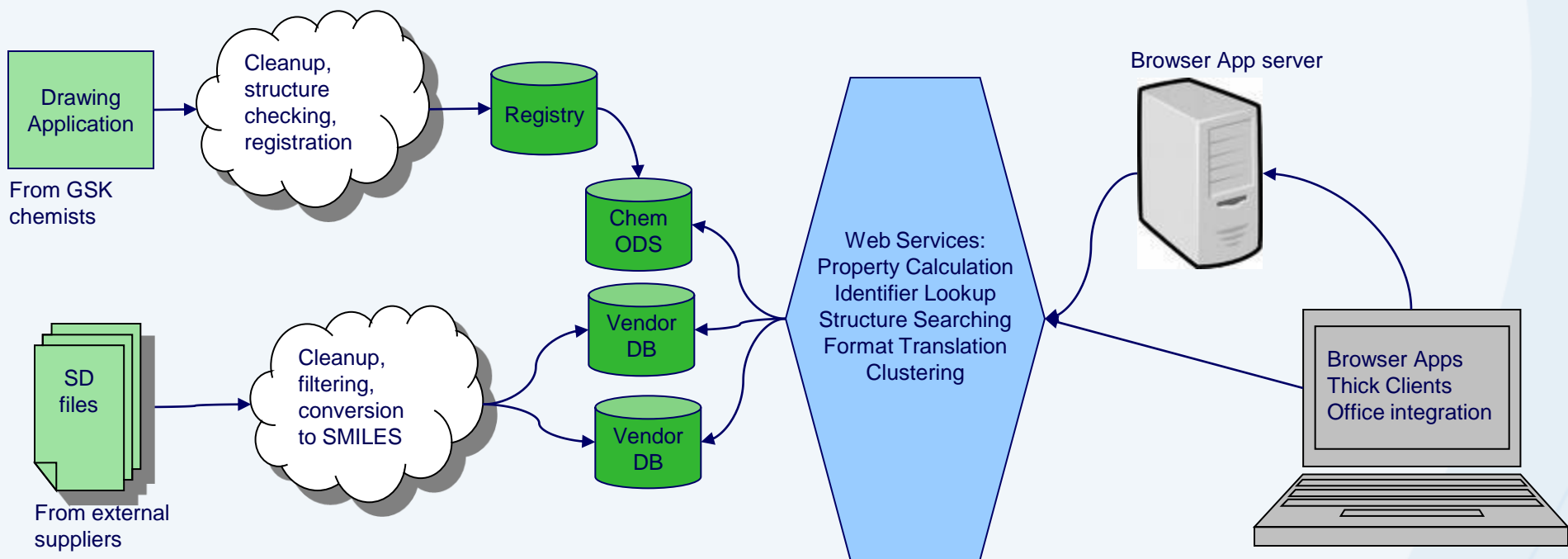


GlaxoSmithKline

Slides cut from presentation

Intro to GSK's SOA Architecture

- Generic picture showing apps middle layer databases. Bubbles for data cleaning and data view creation. Maybe a few contentious middle layer things to get q's unrelated to CXN
- This is just a rough sketch I can make nice later:

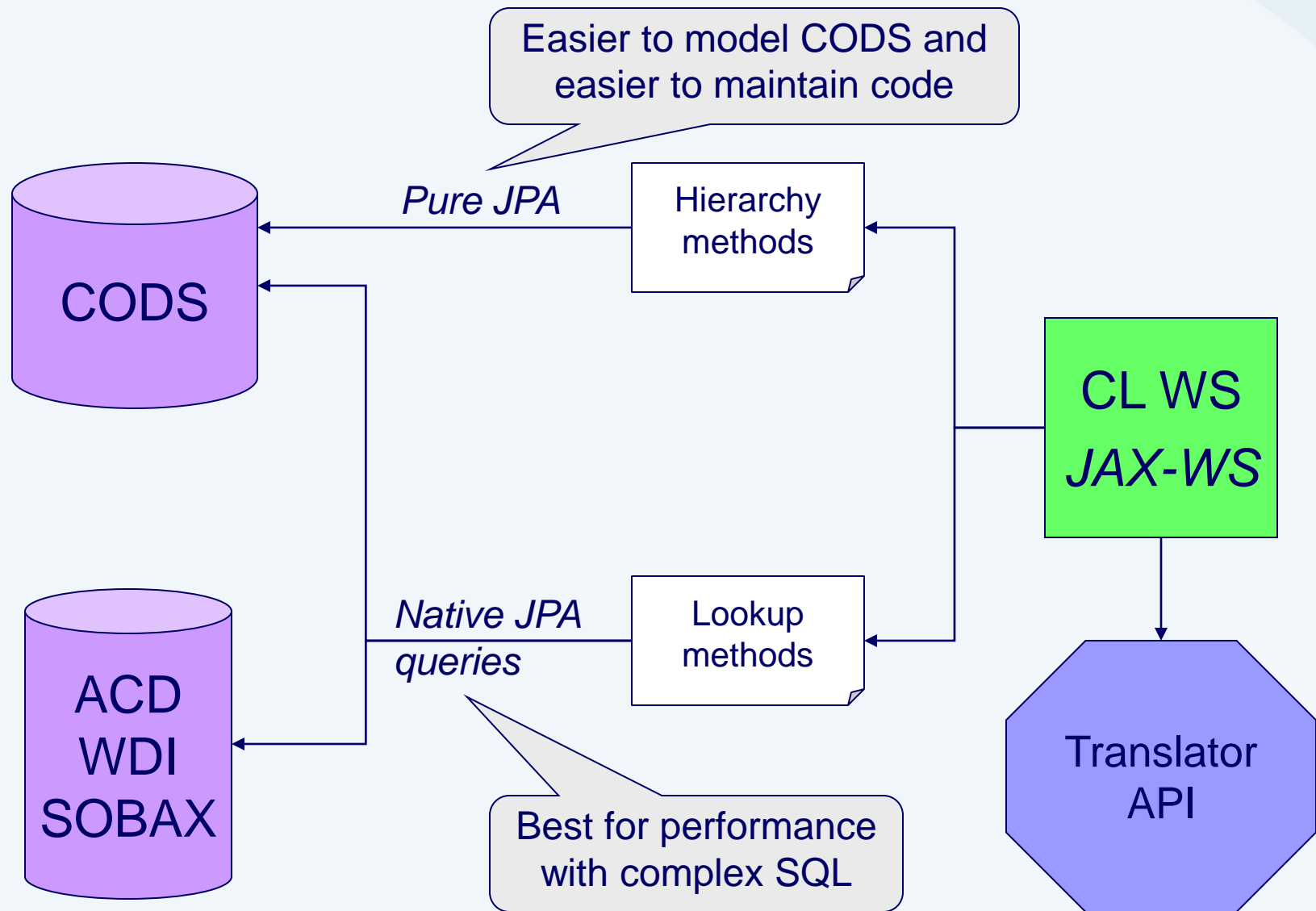


Chemistry Lookup Service

Identifier relationship methods

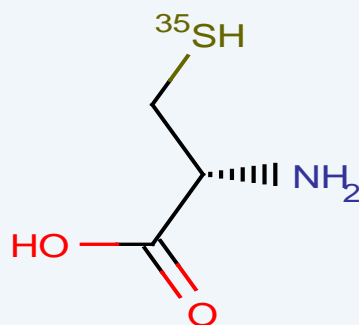
- GetRegistryHierarchy / GetRegistryVersionHierarchy
 - Searching cods database only
 - Return results in a tree format
 - Response governed by a set of predefined behaviours:
 - NORMAL – return the entire tree that the supplied identifier belongs to
 - MINIMAL_TREE – return the tree but omit the siblings at the supplied identifier level
 - SYNONYM_TREES – return all the related “synonym” trees
 - CURRENT_TREE – return the tree (in MINIMAL_TREE form) that all new compounds will be registered under
 - Cross-registration identifiers supported (but will not be returned in response)
 - GetRegistryVersionHierarchy method for those users not interested in parent structures

Chemistry Lookup Architecture



MF Generation. Which of these is correct?

HCl



Smiles: Cl.N[C@@H](C[35SH])C(=O)O

Q. Molecular Formula as calculated by some data cartridges

C3H8ClNO2S

C3H8ClNO2S

C3H8ClNO2[35S]

C3H7NO2S.ClH

C3H7NO2[35S].ClH

A. C3H7NO2[35S].HCl Was what the scientist wanted to see. So none of them.

Chemical Drawing in a Mixed Environment

- Difficult to standardize on a single drawing package due to differing database formats. One approach is to converse with web services in molfile format and allow the services to do necessary translations.
- Most tools can not generate SMARTS queries for Daylight. Might need to use specific tool such as Marvin
- Enhanced stereochemistry such as atom-centered relative and absolute flags not universally implemented among tools, but see pretty good compatibility between ISIS/Symyx Draw and ChemDraw
- Data sgroups usage to tag data directly to structure still tends to be a problem
 - Difficult to search and interpret, layout on screen problematic, sometimes an image is better
 - Vendor specific but compatibility between ISIS/Symyx Draw and ChemDraw is improving. When is it a good idea to do it, when not, use of an image instead to show markups etc.

Common Standards used across Web Services

- Common Agile Development standards used across all Web Services
- Development Test driven, tools include:
 - JUnit (e.g. for Chemistry Lookup 535 unit tests), JMock, FindBugs, CheckStyle, SoapUI (e.g. for Chemistry Lookup 212 soapui tests), TcpMon
- Continuous Integration using Hudson
 - Code coverage and test reports
 - Includes both a unit test build and a regression test build. Regression test build deploys the service and runs soapui test suite.
- Load testing
 - SoapUI offers an inbuilt load test function. However, it does not truly emulate multiple client scenario.
 - SoapUI deployed to the GSK windows grid (like seti@home). Services tested at varying levels of usage to check load and response time.
- Monitoring
 - Support group use SiteScope, monitored offshore.