BRIEFINGS IN BIOINFORMATICS. VOL 9. NO 6. 506–517
Advance Access publication December 6, 2008
doi:10.1093/bib/bbn034

Data curation + process curation = data integration + science

Carole Goble, Robert Stevens, Duncan Hull, Katy Wolstencroft and Rodrigo Lopez

Submitted: 16th May 2008; Received (in revised form): 25th July 2008

Abstract

In bioinformatics, we are familiar with the idea of curated data as a prerequisite for data integration. We neglect, often to our cost, the curation and cataloguing of the *processes* that we use to integrate and analyse our data. Programmatic access to services, for data and processes, means that compositions of services can be made that represent the *in silico* experiments or processes that bioinformaticians perform. Data integration through workflows depends on being able to know what services exist and where to find those services. The large number of services and the operations they perform, their arbitrary naming and lack of documentation, however, mean that they can be difficult to use. The workflows themselves are composite processes that could be pooled and reused but only if they too can be found and understood. Thus appropriate curation, including semantic mark-up, would enable processes to be found, maintained and consequently used more easily. This broader view on semantic annotation is vital for full data integration that is necessary for the modern scientific analyses in biology. This article will brief the community on the current state of the art and the current challenges for process curation, both within and without the Life Sciences.

Keywords: *curation; semantic annotation; processes; services; workflow; ontology; metadata*

INTRODUCTION: WHY

This briefing presents the need for the curation, including the semantic annotation, of the *processes* that filter or transform data as part of a bioinformatics analysis and the vital part this will play in data integration. Integration is a central activity in bioinformatics; it is a perennial problem that has had many proposed solutions [1]. The bioinformatics landscape is one of distributed and heterogeneous data and tools—a landscape a bioinformatician needs to navigate in order to perform the analyses so necessary to modern biology. Today there are a bewildering array of resources available to the modern bioinformatician or molecular biologist—what Stein calls a 'Bioinformatics Nation' [2]. For example, *Nucleic Acids Research* describes 1037 databases [3] and 166 web servers [4]; numbers beyond *ad hoc* reliance on human memory for management and use.

Bioinformatics analyses are a mixture of data and processes. These combinations are often complex. Whilst such analyses are data orientated, it is the services representing the tools that provide, filter or transform these data [5] and form the data pipelines that are common throughout bioinformatics.

The days of a scientist having to cut and paste between different web interfaces are gone—this is simply not a scalable or reproducible process in an era of high-throughput analysis and large-scale data generation [6]. Manual queries through web forms are increasingly being replaced by automated queries through web services [7].

Web services provide a well-defined programming interface to integrate tools into applications over the internet or other network connections. Software applications written in various programming languages and running on various

Corresponding author. Carole Goble, School of Computer Science, University of Manchester, Oxford Road, Manchester, M13 9PL, UK. Tel: +44 161 275 6195; Fax: +44 161 275 6236; E-mail: robert.stevens@manchester.ac.uk

Carole Goble is a Professor of Computer Science at the University of Manchester.
Robert Stevens is a senior lecturer in Computer Science at the University of Manchester.
Duncan Hull is a Postdoctoral Research Associate in the School of Chemistry at the University of Manchester.
Katy Wolstencroft is a Postdoctoral Research Associate in the School of Computer Science at the University of Manchester.
Rodrigo Lopez is head of the external services group at the European Bioinformatics Institute (EBI), Hinxton, Cambridgeshire.

platforms can use web services to exchange data over the Internet.

Using web services to build complex networked tool chains is now a widely accepted solution in bioinformatics for the everyday work of the biologist; for tools and applications [7, 8]. Systems such as Life Science Grid [9], ONDEX [10], GMOD [11], UTOPIA [12] and Vl-e [13] use web services behind the scenes to plug-in services—data sets and tools—into their integration systems, as do warehouses like ATLAS [14] and integration frameworks like GAGGLE [15], and DAS [16]. Commercial systems such as Medicel Integrator[17] do the same. Alternatively, tools can expose web service interfaces to enable scientists to build pipelines (or workflows) of data sources and analyses. For example, scientific workflow management systems automatically orchestrate the execution of services, coordinating processes (process flow) and managing the flow of data between them (dataflow). Workflow management tools such as Taverna, Triana, Kepler, Wildfire, InforSense, Pipeline Pilot and Pegasys [8], provide a mechanism to orchestrate third party and in-house Life Science services. The workflows themselves (Figure 1) are explicit and precise descriptions of a scientific process and, in turn, these workflows can become services within other workflows and applications.

In an effort to manage, analyse and integrate the data deluge we have now created a service deluge. For example, the Taverna Workflow Workbench has access to over 3500 different tools and data resources, over hundreds of third party services. The data that these workflows and services process are often curated, but the processes themselves are poorly curated if they are curated at all. By process curation we mean the cataloguing and annotation of services and workflows and not the content they deliver.

Web services, as a prime example, tend to be poorly described, often with documentation that is insufficient or inappropriate. Their interfaces are commonly (but not always) accompanied by a file that gives the names of the operations performed by the web service, as well as their inputs and outputs; for most this is described in the Web Service Description Language (WSDL) [18]. Unfortunately,

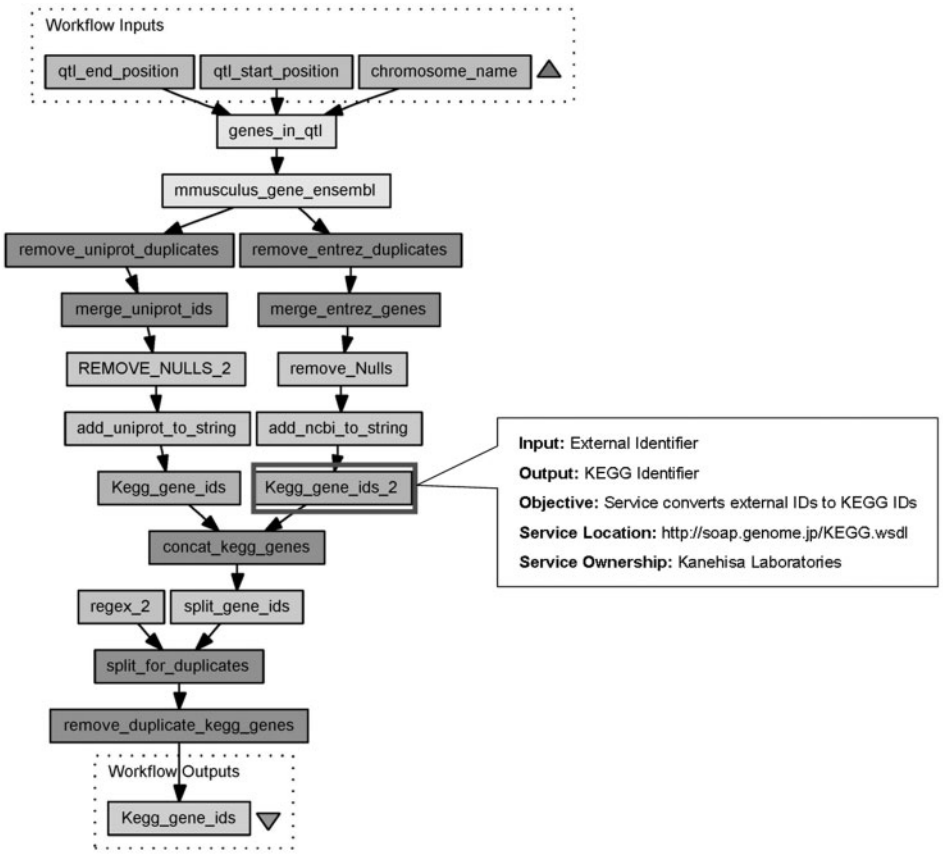

Figure 1: A workflow from the Taverna workflow management system [35], highlighting a KEGG [66] web service operation.

WSDL files are often uninformative [19] and lack information about how they should be used or the kinds of data they produce and consume. For example, one PathPort (pathogen portal) [20] WSDL file [21] describes an operation called 'getRfam' with parameters (all of syntactic type xsd:string) called in0, in1, in2, . . . , up to in7; such descriptions are not uncommon. There is no semantic description to tell a prospective user what these parameters mean, how they should be used or the nature of their data formats. Workflows that can be thought of as compound services, are defined using system-specific files (although BPEL [22] is a proposed standard language, it is not widely adopted in Science) without standards for their documentation. Standardisation of type system, data format and API across the discipline is part of the solution. This is unlikely to happen soon, though there are islands of community standardisation, and such standards would not solve all the needs of a service user.

Both services and workflows need accurate and flexible descriptions, understandable by both people and by software applications. There is no central place that comprehensively catalogues services or workflows world-wide to the Life Science community. This leaves community members to repeatedly gather and use knowledge about resources in an *ad hoc* manner, wastefully duplicating effort. Poor curation of services will make it difficult for their users to navigate the service landscape—yet data integration is the combination of the data and the processes that enable access to the data and the processes that integrate the data. The lack of adequate and standard metadata describing individual services like InterProScan [23] or BLAST often prevent their discovery unless users already know that these services exist, what they do and how to use them. Often the only way to find out what a service really does is to try it.

An absence of curated processes leads to:

- Ignorance of availability: scientists simply do not know what is available and hence reinvent.
- Inadequate information to choose or operate: the lack of sufficient and accurate description is a major obstacle to their potential users and thus the community as a whole, leading to reinvention.
- Poor adoption: services are unknown, unused, poorly used or mis-used as a result.

Active curation of these resources with accurate and flexible descriptions to check their availability, reliability and general quality of service is required. A well-curated resource would potentially enable: *reuse* by improving knowledge of and about processes and hence avoid wasteful reinvention; improve *reliability* by pooling operational histories and reputations; and *validation* by promoting best practice, verified procedures and popular processes.

CURATING PROCESSES: WHO, WHICH, WHERE, WHAT, WHEN AND HOW

Curation of data in the Life Sciences is invaluable for its reuse and comprehension. A similar curation effort to that seen in data [24] is needed to address the problems of discovery and compatibility in the processes that handle bioinformatics data.

Who the curated processes are for

We need to be clear for whom the curated resources are intended. There are many and various types, each with different demands.

Casual users typically access tools and databases using web browser software or perhaps a command line. Their requirements are broad, and their discovery behaviour is *ad hoc* and exploratory. Each use of the service is effectively in isolation, often involving the use of many single applications. The scientists are usually based in an independent laboratory specialising in a particular area where knowledge of the domain is high but not necessarily that of the specifics of the service. Examples include: a bench scientist with a sequence to analyse or a database curator with an entry to annotate.

Expert users, such as *bioinformaticians*, set up pipelines or workflows for automated, systematic bulk analysis of data. Expert users tie together databases and applications in a much broader area of activity, often creating bespoke analysis pipelines requiring many different services and reuse of service compositions. The user responsible for the processes may not be the workflow designer or process developer, so domain and service knowledge can be variable. Service behaviour such as reliability, stability, consistency and currency are important.

Developers are tool builders developing client software or applications that use one or more services and/or workflows. They are involved in the development of methodologies and algorithms. Their domain knowledge is often low but this is counterbalanced by the use of domain experts in

testing and feature requests. Their service knowledge is limited to that provided in the documentation so service usage guidelines and examples are essential.

Providers are the developers and suppliers of the services and workflows and are mostly expert users or tool developers. The catalogue is a means of advertising. They are inherently experts in the process, but this does not necessarily make them well behaved in respect of maintenance, stability and user or application-oriented documentation.

Automatons, such as software that seek processes to automatically plug-in to applications, for example, to seek an alternative service in a workflow when the prescribed service repeatedly fails.

Knowing for whom curation is intended is a guide for what curation to provide; where it should be done and by whom. The exploration of the issues surrounding service curation below is in the context of all these user types.

Which kinds of process

In this briefing we concentrate on two specific classes of processes: web services and workflows. Workflows, however, are not necessarily composed only of web services. For example, only about 60% of the services available to Taverna are web services. Additionally, Taverna can access BioMart queries [25], Beanshell scripts [26], BioMoby Services [27] and many more. All of these also require curation and can be catalogued, so the issues described here are broadly applicable.

A key distinction for the curation process is between *registering* a process and referring to it and that of *depositing* a process and storing it. Both require rich semantic annotation to capture metadata but the curation life cycles and management responsibilities differ. Service catalogues (such as the ^{my}Grid Service Registry) typically register off-site services hosted remotely; workflow or script catalogues (such as myExperiment [28]) are typically storage repositories, where workflows are uploaded and potentially delegated to the care of the resource and its curators (though they could also refer to remotely held XML documents).

Where processes can be found

Currently the scattered workflows and services are most likely to be located by word of mouth or Google searches. Groups or individuals gather them on their web sites or portals. There are, however, a number of efforts to systematically catalogue them.

Biology specific web service catalogues. Several independent initiatives have been established to collect and curate web services related to a specific Life Science field. The Stargate Portal [29] specialises in Glycoproteomics and is aimed at developers rather than expert or casual users and is tightly bound to WSDL-only services. Sswap (Simple Semantic Web Architecture and Protocol) [30] caters for proteomics researchers and has rich descriptions that exploit reasoning over an OWL ontology, but is hampered by an interface that requires the user to have a high degree of a priori knowledge of the underlying model to be effectively used. The DAS Registry [31, 32] logs over 500 DAS biology servers that use a unique http protocol that predates standardised web services. BioMoby Central [27, 33] is a registry of only BioMoby web services [34] and is a potentially rich source of well-annotated services, though in practice the annotations are often rather general. The ^{my}Grid Service Registry [35] contains 700+ web service descriptions, expertly annotated by a full time curator; however, though publicly available and used by clients such as Find-O-Matic [12], its discovery tool, Feta [36] is only available as a plug-in from the Taverna Workflow Workbench. Other efforts include uncurated personal lists, group portals and curated lists of links to web sites, such as the Bioinformatics Links Directory [4] but do not describe them in a way that they can be run.

General web service catalogues and search engines Web Services List [37], XMethods [38], Salesforce [39] and Wsoogle [40] are commercial online directories with a wide range of services. They are general, tend to lack rich metadata and tend to be aimed at service providers and tool developers, rather than expert or casual users, but they have huge coverage. All have some sort of community rating to record experience and the 'quality' aspects of services. Woogle [41], was an early research attempt at a web service search engine with no commercial implementation. SeekDa [42] is an active commercial search engine site for any kind of web service with facilities for reviewing, testing and tagging services. An associated research project called Service Finder—part of the SOA4 All initiative—promises a platform for service discovery embedded in a Web 2.0 environment and emphasises the use of semantics to describe and discriminate between services. There are plans for automated and semantic annotation, but other curation policies are unclear.

Workflow repositories are less mature and tend to be project specific portals, for example, the INB MOWserver and BioWep [43] or product specific, e.g. InforSense's Customer Hub [42], the Pipeline Pilot [44] and the Kepler component repository [45]. An exception is the myExperiment [46] social networking and scientific workflow repository [28]. This is a Web 2.0 style repository that emphasises community participation for workflow developers to upload their workflows for the common good of the community. Although developed for the Taverna workflow community, it is intended as a resource for all workflow systems, recently incorporating Triana workflows. myExperiment has particularly addressed the issues of sharing and credit, with an underlying model to support author attribution, versioning and cross-workflow reuse analysis. A custom Google search [47] finds workflows from a range of systems.

Present efforts to produce catalogues tend to be biased toward the developer community or experts (Table 1). Many have web-browser-based interfaces and a few provide a client API to the catalogue itself such that it can be called from an application rather than just browsed from a web site. WSMO Studio, SAWSDL [37] and BioMoby have Eclipse plug-ins such that services can be found from the development environments used in their creation.

WHAT METADATA TO CURATE

A well-curated resource will support reuse, reliability and validation for a range of users, both human and machine. Metadata can be used to: *index* processes; *cluster* them with respect to usage, user or properties; *match* and *rank* them, with respect to each other and with respect to a search request, for *similarity*, *relatedness* and *complementarity*; and *operate* the process correctly and safely, catering for a range of technologies for delivering services (not just WSDL). Thus the metadata must be sufficient to support selection from a large collection and discriminate between processes that are functionally equivalent.

Describing web services is a challenging activity [48], in part because the providers of the services are not the same as the range of consumers, who have their own differing needs. The WSDL document that accompanies many web services is designed for machine processing and service invocation rather than human comprehension; workflow languages are designed to plan the execution of a series of processes, but again they carry a fraction of the metadata required to describe the processes used in bioinformatics analyses. A fully curated registry of services and workflows would ideally contain the following:

Functional capability: the task that the service performs; the data resources and/or any underlying algorithms it may use; the operations it supports; the type and representation of the inputs and outputs of those operations; and the relationships and constraints between the inputs and the outputs; validation rules on input parameters; examples of use with data and/or clients. Workflows are often complex, and web services are often 'black boxes' that do not reveal their workings; consequently good explanations are essential to enable them to be used appropriately. This is, however, the most difficult and effort-intensive metadata to obtain.

Operational capability: invocation and configuration—where and how to execute a service or workflow and configure it correctly, with examples, defaults and error handling guidelines; circumstances—when to use it, what technical set-up is needed to use it; performance prediction profiles; reliability—failure rates and quality of service; stability—volatility of the interface to the service; availability—licensing and pricing; usage policy—limitations on service usage such as parallel processes or number of submissions per day; access and security policies; mirrors—replicas of identical services; consistency—will results of successive runs be safely comparable; versioning—availability of past versions, version history and difference tracking between versions; and maintenance including depreciation—how often is the service itself, or the process it runs, is updated; and miscellaneous contract obligations.

Provenance: the origins of the service and its host/authors; and the sustainability policy, maintenance profile and service support. A service that has run out of funding will decay and should not be relied on; certain service providers or workflow authors will be trusted over others and a good provider will give notice of downtimes, problems and new versions.

Curation provenance: who has provided the service annotations and how does this person/software relate to the service. If a service has multiple, conflicting annotations, which is correct? Expert service curators or service providers may be more reliable than casual users.

Usage: the emergent reputation and popularity of the service or workflow in the community are important selection and ranking criteria. Properties include: anonymous and attributed ratings

Table 1: A summary of a selection of process catalogues

	DAS	myGrid	BioMoby	Stargate	Sswap	SeekDa!	Wsoogle	XMethods	Saleforce	myExperiment	Pipeline pilot repository	InforSense customer hub
Process type	DAS Service	Web services, Beanshell scripts, BioMART queries	BioMoby Web services	Web services	Web services	Web services	Web services	Web services	Web services	Workflows	Components and protocols	Workflows
Coverage	500 servers	3000 + operations Life Sciences 700+ semantically curated	1200 + Life Sciences	10 Proteomics	1500+ Bioinformatics	27000+ Wide range 203+ Life Science relevant	400+ Wide range ??? Life Science relevant	500+ Wide range ??? Life Science relevant	800 Wide range 10 Life Science relevant	150+ Taverna WfMS Triana WfMS ??? Life Science relevant	700 components 350 protocols Chemistry and Life Sciences	Known only to registered customers
Target users	Genomics experts Software developers Providers	Bioinformatics Experts	Bioinformatics Experts Developers Providers	Proteomics Experts Developers Providers	Developers	Developers Providers	Developers Providers	Developers Providers	Developers Providers	Scientist Casual and Expert Developers Providers	Scientist Casual and Expert Developers Providers	InforSense customers Experts, Developers
Access	DAS Any client	Taverna Feta client plug-in	Web browser BioMoby client	Any client	Sswap client Web browser	Web browser	Web browser	Web browser	Web browser	Web browser Any REST client	Web browser	Web browser
Metadata	Unknown	RDF and OWL	MySQL database and RDF	SAWSDL	OWL	Metadata in WSDL, tagging voting	Tagging Schema unknown	Tagging Schema unknown		Metadata in files, tagging, voting,	Simple schema Free text	Schema unknown
Functional	?	+++	++	+++	+	+	+	+		+	–	?
Operational	+	–	++	?	?	++	+			–	–	?
Provenance	?	+	+	?	?	+	+			++	+	?
Curation	?	–	–	?	?	–	–			+	–	?
Usage	?	–	?	?	?	++	+	+		++	–	?
Curators	DAS experts	myGrid experts, robot web scavenger	BioMoby community	Stargate	Sswap developers	Robot web crawler Community tagging	Wsoogle curators	XMethods curators		Community tagging	Pipeline pilot curators	InforSense curators

+ means relatively more; − means relatively less.

and recommendations on quality of a service and usefulness; popularity—who uses it, what for and download/execution statistics; validation—certified by authorities or part of 'best practice' packs; and clusters and co-use configurations—services that are commonly grouped and used together; *de facto* or prescribed inter-service dependencies; alternate service substitution sets; and workflows/services patterns identified by usage (people who used workflow A also used workflows B and C and services X,Y,Z). Popularity is not only a discriminator for users but a key sustainability metric for service providers and their funding agencies.

They present a view of an ideal world of total curation. Operational, provenance and usage metadata can be largely automated or gathered incidentally through observations and analysis of logs. Functional metadata typically cannot. In this regard current catalogues broadly fall into two categories: (i) Richly annotated and curated, but tightly controlled catalogues using specialist ontologies and software that are product or area specific. These often have hard to use or non-intuitive interfaces for both curating and discovery. These also tend to have low but specialist coverage. (ii) Catalogues with wiki-style or simple tagging with a broad coverage and simple to use search-based discovery, but limited curation, poor standardisation on tags, documentation or quality control. These tend to have high coverage and better operational and usage metadata (Table 1). Human curators are accurate but have low coverage. Robotic web crawlers are fast, have high coverage but generate poor and inaccurate metadata, as there is insufficient metadata for them to harvest. For example, Sswap has a rich metadata model and a large coverage but on investigation the descriptions are too vague to be useful. On the other hand, the [my]Grid Registry curator carefully annotated 98 services from 18 workflows but it took 2 months of effort.

The spectrum of metadata [49], the effort and expertise of the user and curation base; the tools available for curation; and automated instrumentation in order to gather operation and usage analytical data also depends on: the representation of the metadata models. Tags, text and keywords are ideal for humans to interpret but poor for machine computability. Consequently, efforts to use formal ontologies using machine computational languages such as RDFS, OWL and F-Logic are a popular research area [50, 51].

The cost of functional metadata curation must be weighed against the use to which it is put (Figure 2). We observe two major uses [52]:

Decision support, typically discovery by people. Metadata are used to index and group processes classified on key properties drawn from all the categories described

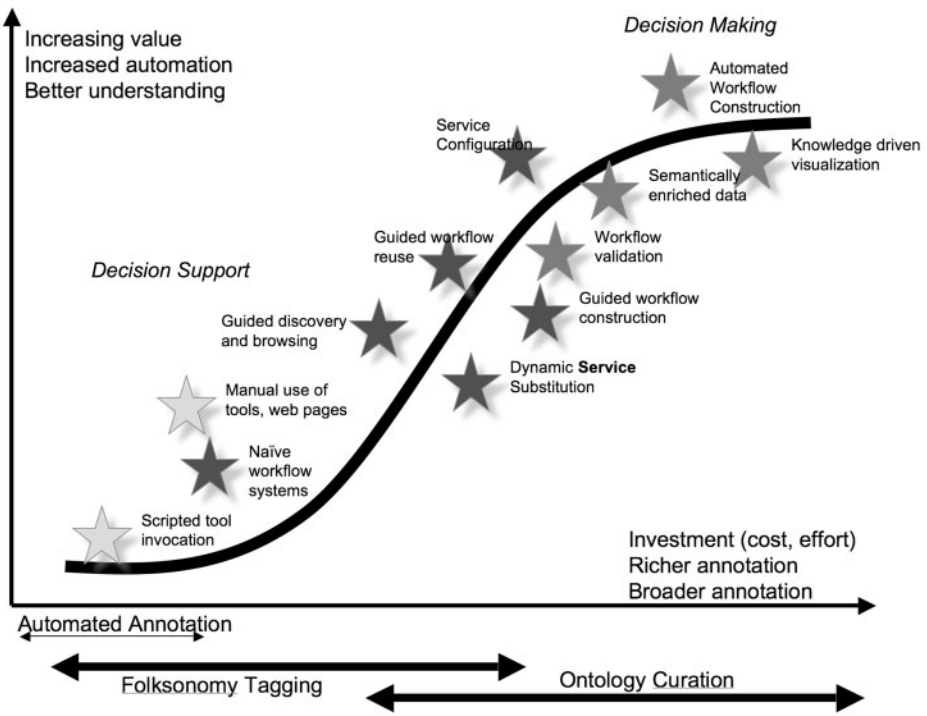

Figure 2: A curation continuum for functional metadata.

above. Simple taxonomies, thesauri languages such as the Simple Knowledge Organisation System [53], and keyword tags are thought adequate and obtainable. Examples include: XMethods, Seekda, Wsoogle, Salesforce and myExperiment. The functionality goes into the search, navigation and browsing tools designed to answer questions such as 'find a workflow like this one' or 'find a service that takes this data as input and produces this kind of data as a result'. The myGrid and BioMoby Registries are presented in this way—despite being underpinned by richer ontologies, they use simple taxonomies to drive their discovery and assembly interfaces.

Decision making, typically automated validation and execution. Metadata are used for automatic or assisted service configuration, substitution and composition—for example, chaining the inputs and outputs of services together and driving workflow assembly tools [54]. It is also used for workflow debugging and repair, for example, finding services that mediate between mismatched services by format transformations (these services are often referred to as 'shims') [55]. This kind of metadata is typically much richer in describing the functional capabilities, using reasoning of some form over feature-based descriptions. The semantic web service community has focused on auto-service composition and invocation, rather than manual discovery, e.g. OWL-S [51] and WSMO [56]. The drawbacks are the high cost of annotation, poor tooling and unclear benefits for the majority of human centred applications and users. An expert using WSMO Studio takes half a day to annotate a single service. BioMoby has made an effort to make services interoperate by using semantic annotations and has gathered an active community; the approach has been to keep the models as simple as possible. The myGrid registry has a curation track where rich annotations are used to spot service mismatches, but this is only used by specialist Taverna plug-ins.

Standardisation efforts for web service metadata are still in their infancy. Registry standards UDDI [33] and ebXML [57] are XML-based specifications and protocols that underpin the more commercial registries in Table 1, but are rigid and impoverished with respect to their metadata coverage, and in particular the functional semantics. Implementations of UDDI such as GRIMOIRES [58] end up having serious and clumsy extensions. The W3C proposes SAWSDL as a recommendation for semantic annotation of services [37], largely focusing on a subset of the functional capabilities. It is not widely adopted as yet, and is only applicable to WSDL documents. The more semantically rich OWL-S and WSMO efforts have practically no adoption in the field.

In biology, the minimum information about the operation of a web service (MIAOWS) has been proposed in a submission to the minimum information for biological and biomedical investigations (MIBBI) [59, 60]. This effort is nascent and does not seem to be tied to any specific or public service cataloguing initiative. The principle of a lightweight, minimum model with the capability for progressively enriching services on a 'pay as you go' basis is, however, laudable.

The tendency of the cataloguing world has been to adopt a particular model and then make that transparently obvious to the community. A much more flexible approach is to internally develop a model and externally present various APIs that *behave* as standard models and *map* to the internal model. For example, myExperiment takes a flexible view drawn from standards in the digital library and open repository world. It has adopted the Open Archives Initiative [61] standardised object information and interoperable protocols for describing aggregations of URLs. The message is that services and workflows, and their accompanying documents and annotations of any form, can be accrued incrementally into scientific research objects (SRO) and regrouped into packs, which are nested. An example might be the workflows, services and tutorials for Bioinformatics 101. SROs are themselves subject to curation.

How and when to curate

Processes are, by their nature, dynamic. Services are not deposited and preserved—they are remotely referred to and are volatile, subject to continual change, often changing with little warning. They change their location, their capabilities and their signatures; new ones appear and existing ones disappear; they decay and become outdated and so forth (Figure 3). Consequently, the services need constantly monitoring and their metadata needs constant tending. In particular, operational and usage metadata can quickly become stale.

Although workflows are deposited, they are also subject to a life cycle of curation. They spawn versions. They are broken up and combined to create 'workflow remixes' whose provenance and attribution must still be tracked and whose metadata must be propagated, and they are composites of

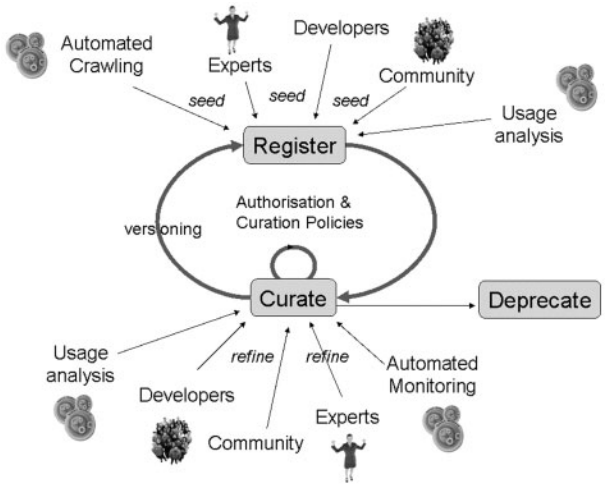

Figure 3: Curation life cycle and its participating curators.

services that are volatile. In myExperiment this is supported by a *versioning and contributions* system.

Active monitoring and metadata revision are important. SeekDa [51] and DAS support active service monitoring, and BioMoby services have a simple Boolean 'Is Alive' feature. Recently, the myGrid project has developed a monitoring service called WorkflowMonitor as part of a suite of tools for handling and notifying users of workflow decay through the myExperiment system. The notion of a life cycle for services and their metadata has important implications for what metadata are provided for whom and by whom. Curation process management also needs explicit and computationally processable *policies* for: authorisation (who can curate/view what and when and who decides); monitoring (what changes when, how to know and what to do when it does), versioning and usage.

Who curates and how

Curating is a non-trivial undertaking that needs to balance content coverage against content quality, and give those who use and contribute to the catalogue a sense of ownership and control. By contributions we mean the processes themselves, their metadata descriptions and the analysis arising from their everyday use.

We group curators into four categories:

Expert curators: bioinformaticians who understand the services and workflows whose job is to annotate and set up the curation pipelines, for services and workflows that are not of their own making.

Self-curation: some registries are closed—the myGrid registry is only curated by experts from the myGrid project itself. Others encourage service developers to self-curate, emphasising the use of plug-ins to service development environments such as Eclipse; examples include BioMoby's jMoby plugin and SAWSDL4J, Lumina and Radiant toolkits for SAWSDL and WSMO Studio (21). Workflow repositories such as myExperiment rely on self-curation by the workflow developers and community curation by their users. Challenges include (i) the enforcement of controlled vocabularies by self-curators, particularly if the vocabularies are also managed by the developers as they can quickly become unruly and (ii) incentivising people to contribute their services and workflows for the good of the community.

Community curators: the trend is to follow in the footsteps of popular Web 2.0 social computing sites and encourage community curation through user feedback, blogging, e-tracking, recommendations and folksonomy-based tagging. Community approach to services development and use being tried by Seekda and BioMoby and for workflows by myExperiment. Community and self-curation requires built-in incentive models for people to contribute such as credit and attribution, but can be made to work, for example, iCapture successfully pioneered community curation of ontologies (Wilkinson PSB).

Automated curators: automated scavengers and crawlers identify candidates for submission and extract as much metadata as possible. Functional metadata is hard to auto-curate, requiring: specialist metadata extraction tools [54]; software plug-ins that incidentally gather metadata from services as they are used in applications; or smart reasoning over seeded service descriptions and workflows [54]. Operational and usage metadata is ripe for automation, generated from monitoring services, application diagnostics, customer reports and social network analysis. Workflow analytics is the term used for processing workflow collections to identify, for example, service co-use patterns and service popularity. Automated curation needs excellent infrastructure.

In reality, a catalogue should support all the above in not so much a curation pipeline as a curation swirl (Figure 3) where curated content—services, workflows and metadata—is seeded, validated and refined

continually by the different players. The desire to creatively free-wheel descriptions is tensioned against the need to organise and structure. Consequently, catalogues need to support 'metadata creep' overtime in multiple forms, and of varying depth and quality. Expert curators become moderators, policing and enhancing the catalogue rather than the primary source of content.

Attention to versioning, privacy, intellectual property protection and security is still rudimentary or neglected in most cases. The functional, and some operational, capabilities are described using controlled vocabulary terms drawn from folksonomies and tagging [62], some using the W3C Web Ontology Language OWL (Sswap, Stargate) and Resource Description Framework Schema (BioMoby). myGrid has a full time curator for enriching web service descriptions with a Service Ontology [52, 63]. This ontology is also used by Feta [36], a service discovery plugin for Taverna, which describes the biological *in silico* research domain and the ways in which a service can be characterised from the perspective of the scientist.

DISCUSSION

Integration within bioinformatics necessarily involves the processes that filter and transform the data being integrated. Just as bioinformatics data are annotated in order to make them available for integration, so must processes—the services used by bioinformatics—be annotated. The community needs a well-curated catalogue of all services and workflows in the Life Sciences with progressively enrichable metadata that serves a range of uses and users, but is simple to use. There are, however, many difficulties: [64, 65], for example, identify several obstacles.

As touched upon in the 'Introduction: why' section, this annotation of services is not the only solution to the problem. Much of the envisaged annotation is a substitute for a type system and a means by which the API may be 'understood'. This annotation could be obviated by the use of a common type system throughout bioinformatics supported by standardisation throughout the discipline. Many projects have developed typed systems (e.g. MOBY or CaGrid) and there are many community data standards. The adoption of these across the discipline is, however, a social rather than a technical problem and it is not likely that such a global standardisation will happen in the near future. An annotation effort is therefore a lower cost effort that can bring immediate benefit and can accommodate the issues of legacy that are bound to arise. In addition, annotation will remain an element of any standardisation effort: Elements such as versions; the organisation developing the service; quality metrics; the relationships between data inputs and outputs; etc., will remain best undertaken by annotation. As standards arise and are adopted, then annotation can adapt to the changing circumstances.

As a consequence, the authors are initiating the BioCatalogue (http://biocatalogue.org) project, an offshoot of myExperiment, but incorporating the experiences of the myGrid Registry. bioCatalogue will provide a means by which a bioinformatician, for instance, could upload a web service file such as WSDL and have it annotated and made available via the catalogue. On upload, as much information as possible such as service operations, inputs, outputs will automatically be extracted from the WSDL and displayed within an annotation user interface. The bioinformatician can then add more text to describe the web service and map the web service elements to terms in the myGrid service and domain ontology. Missing terms can be substituted with tags that can be subsequently mined for new ontology terms. Annotation dates and histories can also be part of the markup, as could the last test from WorkflowMonitor. The latter will 'ping' the service to see if it is alive. The 'lapsing' of annotations of unmaintained resources can thus have some element of management. BioCatalogue aims to support all the users described in the 'Who the curated processes are for' section.

The goal of BioCatalogue is to improve the process: reuse, reliability and validation by encouraging self and community curation alongside automated and expert curation pipelines. By generating content that can be indexed by third party information providers, such as Google, it should be easy to find. By presenting programmable APIs we aim to make the catalogue easy to mashup and incorporate in third party applications. As a consequence, the Bio-Catalogue aims to provide the missing part of the equation that yields science from data integration.

Key Points

- The processes that are part of delivering and analysing biological data are a vital part of integrating those data.
- There are a range of metadata that can be used for describing these processes, starting with descriptions of inputs, outputs, tasks, etc., all the way to those describing reliability and performance.
- Catalogues of curated processes are needed so that processes, both individual services and their compositions in workflows, can be found and reused.
- Services, like the data they process, are dynamic entities with a life cycle that needs sustained input.

Acknowledgements

The authors would like to acknowledge the following members of the myExperiment, myGrid Registry and BioCatalogue teams for their work and for their contributions to this briefing, in particular: Katy Wolstencroft, Marco Roos, Jiten Bhagat, Mark Borkum, David De Roure, Don Cruickshank, Antoon Goderis, Thomas Laurent, Danius Michaelides, Eric Nzuobontane, Kaixuan Wang and Hamish McWilliam.

FUNDING

Microsoft Corporation; Joint Information Systems Committee; Biotechnology and Biological Sciences Research Council (BB/F01046X/1, BB/F010540/1); Engineering and Physical Sciences Research Council (EP/D044324/1, EP/C536444/1).

References

1. Goble C, Stevens R. State of the nation in data integration for bioinformatics. *J Biomed Inform* 2008, http://dx.doi.org/10.1016/j.jbi.2008.01.008.
2. Stein L. Creating a bioinformatics nation. *Nature* 2002;**417**:119–20.
3. Galperin M. The molecular biology database collection 2008 update. *Nucleic Acids Res* 2007, http://dx.doi.org/10.1093/nar/gkm1037.
4. Fox JA, Butland SL, McMillan S, *et al*. The bioinformatics links directory: a compilation of molecular biology web servers. *Nucleic Acids Res* 2005;**33**:W3–24.
5. Stevens R, Goble C, Baker P, *et al*. A classification of tasks in bioinformatics. *Bioinformatics* 2001;**17**:180–8.
6. Stevens RD, Tipney HJ, Wroe CJ, *et al*. Exploring Williams – Beuren syndrome using myGrid. *Bioinformatics* 2004;**20**:303–10.
7. Romano P, Marra D, Milanesi L. Web services and workflow management for biological resources. *BMC Bioinformatics* 2005;**6**:S24.
8. Taylor IJ, Deelman E, Gannon DB, *et al*. Workflows for e-Science, Scientific Workflows for Grids. Springer, 2006. http://www.amazon.co.uk/exec/obidos/ASIN/1846285194/.
9. Lilly E. Lilly's science grid goes open source. http://www.bio-itworld.com/BioIT_Article.aspx?id=75790 (13 August 2008, date last accessed).
10. Kohler J, Baumbach J, Taubert J, *et al*. Graph-based analysis and visualization of experimental results with ONDEX. *Bioinformatics* 2006;**22**:1383–90.
11. Stein LD, Mungall C, Shu S, *et al*. The generic genome browser: a building block for a model organism system database. *Genome Res* 2002;**12**:1599–610.
12. Pettifer SR, Sinnott JR, Attwood TK. UTOPIA: user friendly tools for operating informatics applications. *Comp Funct Genomics* 2004;**5**:56–60 (13 August 2008, date last accessed).
13. Hertzberger O. e-Science and the VL-e approach. *Trans Comput Syst Biol IV* 2006;**3939**:58–67.
14. Shah S, Huang Y, Xu T, *et al*. Atlas - a data warehouse for integrative bioinformatics. *BMC Bioinformatics* 2005;**6**:34.
15. Shannon P, Reiss D, Bonneau R, *et al*. The gaggle: an open-source software system for integrating bioinformatics software and data sources. *BMC Bioinformatics* 2006;**7**:176.
16. Olason PI. Integrating protein annotation resources through the distributed annotation system. *Nucleic Acids Res* 2005;**33**:W468–70.
17. Medicel: Research Infrastructure for the Biosciences. http://www.medicel.com (13 August 2008, date last accessed).
18. Christensen E, Curbera F, Meredith G, *et al*. *Web Services Description Language (WSDL) 1.1*, 2001, Technical Report. http://www.w3.org/TR/wsdl.
19. Al-Masri E, Mahmoud Q. *Investigating Web Services on the World Wide Web. WWW2008*. Beijing, China.
20. Eckart JD, Sobral BW. A life scientist's gateway to distriputed data management and computing: the path-port/ToolBus framework. *Omics* 2003;**7**:79–88.
21. Wsoogle.com: Search Web Services. http://www.wsoogle.com (13 August 2008, date last accessed).
22. Zhang N, Rector A, Buchan I, *et al*. A linkable identity privacy algorithm for HealthGrid. In: *Proceedings of the HealthGrid* 2005, pp. 234–45, IOS Press, Oxford, 2005.
23. Quevillon E, Silventoinen V, Pillai S, *et al*. InterProScan: protein domains identifier. *Nucleic Acids Res* 2005;**33**:W116–W120.
24. Bodenreider O, Stevens R. Bio-ontologies: current trends and future directions. *Brief Bioinform* 2006;**7**:256–74.
25. Durinck S, Moreau Y, Kasprzyk A, *et al*. BioMart and bioconductor: a powerful link between biological databases and microarray data analysis. *Bioinformatics* 2005;**21**:3439–40.
26. Konishi F, Yagi T, Konagaya A. MolWorks+G: integrated platform for the acceleration of molecular design by grid computing. In: Tan TW, Arzberger P, Konagaya A, (eds). *Grid computing in life science (LSGRID2005)*. Singapore: World Scientific, 2006;134–41.
27. Wilkinson MA. BioMOBY: the MOBY-S platform for interoperable data-service provision. In: *Computational Genomics: Theory and Application. Horizon Bioscience*. Wymondham, UK, 2004.
28. Goble C, De Roure D. myExperiment: social networking for workflow-using e-scientists. In: *WORKS'07: Proceedings of the 2nd workshop on Workflows in support of large-scale science*. New York, USA: ACM Press, 2007;1–2.
29. Stargate Glycomics Web Portal. http://www.dlib.org/dlib/september07/treloar/09treloar.html (13 August 2008, date last accessed).

30. XMethods. http://www.xmethods.net (13 August 2008, date last accessed).
31. Dowell R, Jokerst R, Day A, *et al*. The distributed annotation system. *BMC Bioinformatics* 2001;**2**. http://dx.doi.org/10.1186/1471-2105-2-7.
32. CRM Software On Demand, Online CRM Solutions - salesforce.com. http://www.salesforce.com/(13 August 2008, date last accessed).
33. Curbera F, Duftler M, Khalaf R, *et al*. Unraveling the web services web: an introduction to SOAP, WSDL, and UDDI. *IEEE Internet Comput* 2002;**6**:86–93.
34. Wilkinson M, Schoof H, Ernst R, *et al*. BioMOBY successfully integrates distributed heterogeneous bioinformatics web services. The PlaNet exemplar case. *Plant Physiol* 2005;**138**:5–17.
35. Hull D, Wolstencroft K, Stevens R, *et al*. Taverna a tool for building and running workflows of services. *Nucleic Acids Res* 2006;**34**:W729–W732.
36. Lord P, Alper P, Wroe C, *et al*. Feta: a light-weight architecture for user oriented semantic service discovery. In: *European Semantic Web conference ESWC* 2005, Vol. 3235 *LNCS*, Springer Verlag, Heidelberg, 2005, 17–31.
37. Kopeck J, Vitvar T, Bournez C, *et al*. SAWSDL: semantic annotations for WSDL and XML schema. *IEEE Internet Comput* 2007;**11**:60–7.
38. ebXML: Enabling a Global Electronic Market. http://www.ebxml.org/(13 August 2008, date last accessed).
39. MIBBI: Minimum Information for Biological and Biomedical Investigations. http://mibbi.sf.net (13 August 2008, date last accessed).
40. seekda.com - Helping to Find and Use Web Services. http://www.seekda.com (13 August 2008, date last accessed).
41. Dong X, Halevy A, Madhavan J, et al. Similarity search for web services. In: Nascimento M, Özsu T, Kossmann D. *et al*. (eds). *Proceedings of the 30th VLDB Conference, Toronto, Canada 2004*. Morgan Kaufmann, 2004;372–83.
42. Lausen H, Haselwanter T. Finding web services. In: *European Semantic Technology Conference*, Vienna, Austria, 2007.
43. Romano P, Bartocci E, Bertolini G, *et al*. Biowep: a workflow enactment portal for bioinformatics applications. *BMC Bioinformatics* 2007;**8** (Suppl 1). http:/dx.doi.org/10.1186/1471-2105-8-S1-S19.
44. Accelrys Pipeline Pilot. http://accelrys.org/pipelinepilot (13 August 2008, date last accessed).
45. Kepler Project. http://www.kepler-project.org (13 August 2008, date last accessed).
46. myexperiment.org, find, use and share workflows. http://www.myexperiment.org/(13 August 2008, date last accessed).
47. MyExperiment Custom Google Search Engine. http://www.myexperiment.org/google (13 August 2008, date last accessed).
48. Lord P, Bechhofer S, Wilkinson M, *et al*. (2004), Applying semantic web services to bioinformatics: experiences gained, lessons learnt. In: *International Semantic Web conference (ISWC2004)*, Vol. 3298 *LNCS*, Springer, Heidelberg, pp. 350–64.
49. Treloar A, Groenewegen D, Harboe-Ree C. The data curation continuum: managing data objects in institutional repositories. *D-Lib Mag* 2007;**13**. http:/dx.doi.org/10.1045/september2007-treloar.
50. Dimitrov M, Simov A, Konstantinov M, *et al*. WSMO studio- a semantic web services modelling environment for WSMO. In: *Proceedings of the 4th European Semantic Web Conference*. InnsbruckSpringer, Berlin-Heidelberg, Austria, 2007.
51. Martin D, Burstein M, Hobbs J, *et al*. OWL-S: semantic markup for web services. 2004, http://www.w3.org/Submission/OWL-S (13 August 2008, date last accessed).
52. Wolstencroft K, Alper P, Hull D, *et al*. The ^{my}Grid ontology: bioinformatics service discovery. *Int J Bioinform Res Appl* 2007;**3**:326–40.
53. Simple Knowledge Organization System (SKOS). http://www.w3.org/2004/02/skos/(13 August 2008, date last accessed).
54. Belhajjame K, Embury SM, Paton NW, *et al*. Automatic annotation of web services based on workflow definitions. In: *International Semantic Web Conference*, Vol. 4273 *LNCS*, Springer, Korea, 2006, pp. 116–29.
55. Hull D, Stevens R, Lord P, *et al*. Treating shimantic web syndrome with ontologies. In: Domingue J, Cabral L, Motta E, (eds). *Advanced Knowledge Technologies Semantic Web Services (AKTSWS)*. http//CEUR-WS.org/Vol-122/(13 August 2008, date last accessed).
56. Roman D, Keller U, Lausen H, *et al*. Web service modeling ontology. *Appl Ontol* 2005;**1**:77–106.
57. Seitz L, Montagnat J, Pierson J, *et al*. Authentication and authorization prototype on the micro-grid for medical data management. In: *Proceedings of the HealthGrid 2005*, IOS Press, Oxford.
58. The GRIMOIRES registry. http://code.google.com/p/grimoires/(13 August 2008, date last accessed).
59. Li W, Byrnes R, Hayes J, *et al*. The encyclopedia of life project: grid software and deployment. *New Generation Comput* 2004;**22**:127–36.
60. Taylor C, Field D, Sansone S-A, *et al*. Promoting coherent minimum reporting requirements for biological and biomedical investigations. The MIBBI project. *Nat Biotechnol* 2008;**26**:889–96.
61. The Open Archives Initiative. http://www.openarchives.org/(13 August 2008, date last accessed).
62. Furnas G, Fake C, von Ahn L, *et al*. Why do tagging systems work? In: Olson G, Jeffries R, (eds). *CHI Extended Abstracts*. New York, NY, USA: ACM, 2006;36–9.
63. Wroe C, Stevens R, Goble CA, *et al*. A suite of DAML+OIL ontologies to describe bioinformatics web services and data. *Int J Coop Info Syst* 2003;**12**:597–624.
64. Goderis A, Sattler U, Lord P, *et al*. Seven bottlenecks to workflow reuse and repurposing. In: *International Semantic Web Conference (ISWC) 2005*, Springer Berlin, Heidelberg, pp. 323–37.
65. Goble C, Wolstencroft K, Goderis A, *et al*. Knowledge discovery for biology with taverna: producing and consuming semantics in the web of science. In: Baker CJO, Cheung K-H, (eds). *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*. USA: Springer, 2007;355–95.
66. Kanehisa M, Goto S, Hattori M, *et al*. From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res* 2006;**34**(Database issue):D354–D357.