API Framework for local digital services
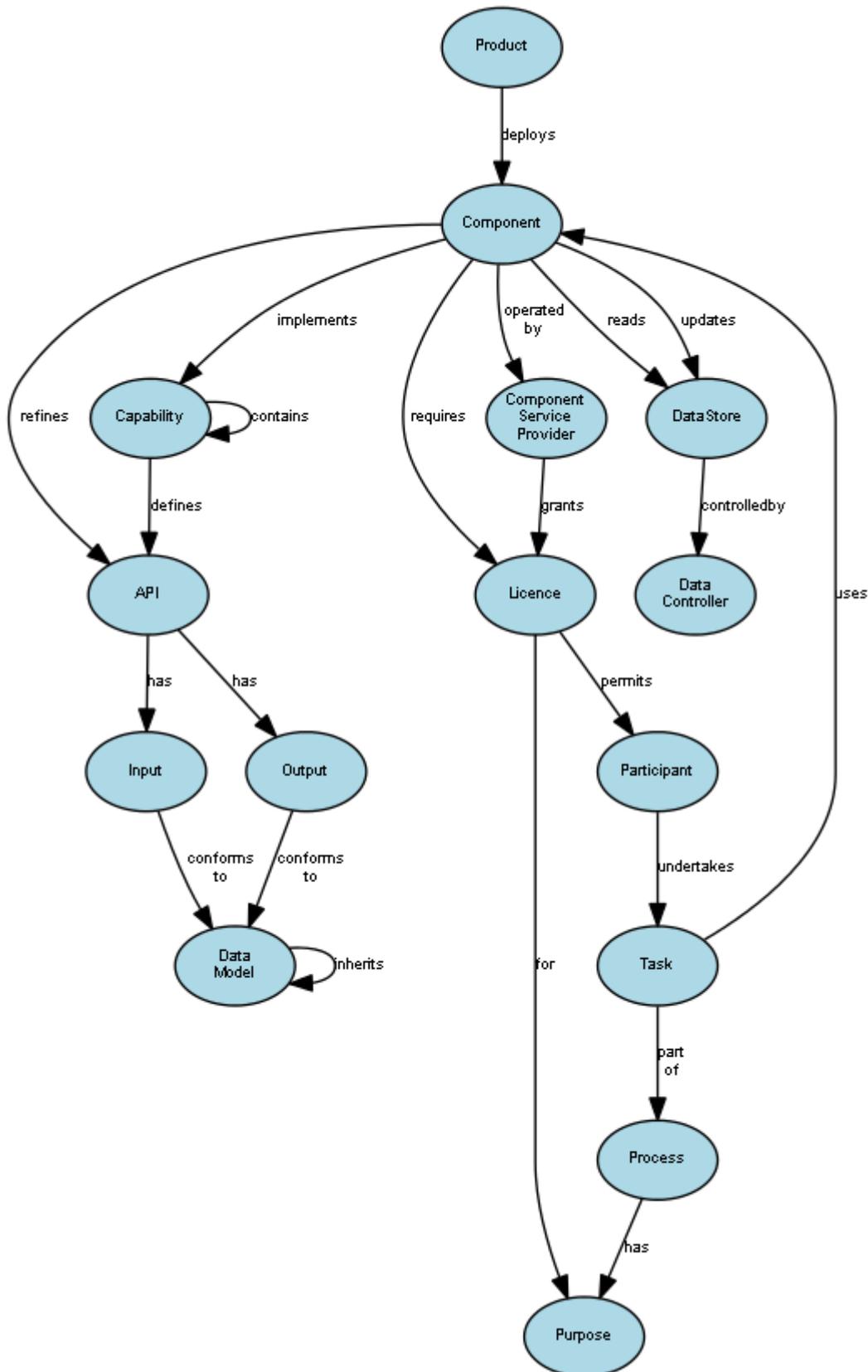
iStandUK (formerly LeGSB)

Version 0.01, Paul Davidson, 9th September 2015

Capability

A 'capability' is a definition of a re-useable functionality that could be consistently adopted across digital partners. The definition includes the 'APIs' that define how the functionality can be accessed. A single 'capability' may have many 'APIs', for example to add information, to query information.

A 'capability' can be implemented as one or more 'components'.

'capabilities' can be grouped together by a parent 'capability' for example

- A 'missed be re-scheduler' may be defined in a 'booking and scheduling' group.

'Capabilities' might include

The Digital Capabilities that could be consistently adopted across local service providers include:

Capability	Description
Citizen identity	A means of identifying users of local digital services that is also trusted by other digital partners
Attribute exchange	A generic mechanism for the online, real time exchange of attributes, with the data subject's consent, to deliver a specific service at a specific point in time, under the governance of a trust framework.
Personal data store	A service to let an individual store, manage and deploy their key personal data in a highly secure and structured way.
Reliable open data	Real-time access to key 'master data' with national coverage, as a foundation for digital participation.
Trusted access to protected data	Near real-time access to protected data where a person is empowered by their organisation to act in a role that has a right to a set of data items for a purpose, and agrees to the terms by which the data is to be used and handled.
Sensors and Actuators	Responding to 'Internet of Things' (IoT) style data services to target or automate intervention.
Content Management	To manage web content and serve it to audiences and devices.
Payments engine	A facility to collect, verify, allocate and clear payment transactions.
Booking and scheduling	A facility to book the use of an asset and associate it with a case.
Texts and alerts	a facility to contact a customer to give information about a case that they are 'subscribed' to.
Transaction log	a history of digital transactions for a case
Forms handling	define, populate, verify, process information submitted by a

	user
Data matching	comparing sets of data to find high probabilities of matching entities, particularly matching on PERSON.

API

An 'API' is a definition of an interface to a 'capability'. An API will have

- inputs
- outputs

APIs will typically be accessed over the web as

- RESTful
- SOAP
- javascript library

An API may require certain tokens to be provided for authentication.

Inputs and Outputs

'Inputs' may include

- query parameters
- sequence control such as pagination
- form data

'Outputs' may include

- query results
- data documents
- result status

'Inputs' and 'Outputs' will conform to a strict syntax and may allow a number of renditions e.g.

- xml
- json
- csv

Data Model

Inputs and outputs will re-use semantic definitions from a Data Model which is applicable to an application, and therefore to many APIs.

- class models defining a type of entity
- properties of a class
- permitted values of a property
- relationships between classes

Some of those definitions will be specific to an application whereas others will define patterns that are found in many applications.

A consistent implementation of semantic models for repeating patterns will encourage APIs to form an ecosystem. A single concept model can be defined in which patterns that are found in many settings can be defined, and then used as the basis for data models for an application.

The data model for an application should be published as a separate asset under version control.

Component

A 'component' is an instance of a deployed 'capability' that is operated by an organisation.

For a some 'capabilities', a single national component may be most appropriate, whereas for others, each local agencies may operate their own 'component'. Where 'components' may be accessed by agencies other than the component operator, they are typically deployed over the web as a 'platform'.

Localised 'components' may refine the 'capability' by offering a reduced scope, or coverage etc.

A 'component' may be operated on behalf of a partnership of agencies, perhaps with coverage for a particular place, community, or audience group.

Product

A 'product' is a facility that can be provided as goods or a service, from a supplier which is the basis for a 'component'. A supplier may claim that a 'product' meets the specification of one or more 'capabilities', and can be accessed via APIs.

Where a product meets a specification for a 'capability', it can be plugged in and out with minimal configuration.

A single product may be supplied to many agencies, as the basis for separate 'components'.

Where a 'product' is a service, it may be accessed, managed and configured over the web, as SaaS (Software as a Service)

Component Service Provider

A 'component service provider' is an organisation that operates a 'component' from which it offers a service to other agencies.

The 'component service provider' sets service levels and quality characteristics for the service.

The 'component service provider' may also offer value added services such as forums, training, integration etc.

Licence

A 'licence' grants a permission from the 'component service provider' to another agency, to access a 'component', and sets conditions on use.

Some licences may be entirely open, that is, free for anyone to use for any purpose, without limitation, whereas other licences may have restrictions such as

- requiring a payment
- undertakings on purpose
- fair use

It would be useful if agreement, and conformance, to a 'licence' was a part of the digital handshake that takes place as an API to a component is accessed.

Purpose

Traditional data-sharing agreements typically cover the 'purpose' for which data can be re-used, which is then often backed up within a legislative framework and specific legal gateways.

Similarly, some 'components' will also provide access to real-time information, which will require the consumer to assert the purpose to which they will put the results to.

Many agencies will share a common 'purpose'.

Process

Define 'process' – link to METHOD

Agencies will have 'processes' to support their local services, which are designed to fulfil a 'purpose'. A 'process' will contain 'tasks', some of which may be common across otherwise unrelated processes.

A digital process may be orchestrated over many digital 'participants' so that, to the end user, the process appears to be end to end, and fulfilled by the organisation that they have engaged with.

Where a process requires participation from more than once agency, each 'participant' will need to provide a consistent API onto their 'componants'.

Task

Define 'task from 'BPMN'

A 'task' is a step in a 'process' that can be fulfilled via a 'component' either operated by the process owner, or, where they have a 'licence' from a separate 'component service provider'.

A 'task' may approach in many processes for an organisation.

Participant

A 'participant' is an agency which is ready to provide its information and services via an API so that they can support 'tasks' defined with a single end-to-end process.

Data Store

Components typically create and/or consume data which is then stored by the 'component service provider'. For example, a single 'capability' may have an API to allow a customer transaction to be recorded, and another API to query transactions for a customer.

Access to the data via the component is controlled by the 'licence'. However, the data is also valuable as a set, for insight and service improvement.

Data Controller

It is possible that a 'component' may manage data on behalf of other organisations such that decisions and responsibility for information governance and security may be taken on subsets of a 'data store', across many 'data controllers'.

Using the framework

The local sector should collaborate to

- define a high level set of 'capabilities' which, if implemented as re-usable platform 'components', will bring benefits;
- prioritise a set of specific 'capabilities';
- consider which 'capabilities' should be implemented as a single national 'component';
- create a marketplace for products which are accredited as meeting the 'API' specification of a 'capability';
- define how APIs should consistently implement control and status 'inputs' and 'outputs';
- devise a common concept model from which application data models are linked.
- accreditation of products against API and 'capability' specifications
- template for component licences
- building a master set of 'purpose' collaboratively, linked to the legislative framework

Component Service Provider

A 'component service provider' operates a 'component'