Designing Personalised Information Access to Structured Information Spaces

George Magoulas and Dionisis Dimakopoulos
gmagoulas@dcs.bbk.ac.uk

Birkbeck UNIVERSITY OF LONDON

Outline

- Structured Information Spaces
- Service-oriented approach for personalisation
- Service-based architectures
- User data models for personalisation
- Conclusions and future work

Structured information spaces

- Popular way to provide e-content: Web directories; Library catalogues; LMS; Musée; LookSmart; yell.com; ultraseek.co.uk; Subject Gateways (SOSIG, SciCentral).
- Main characteristics: facilitate organization, search, and analysis of information resources; static layout; static structure; hierarchical organisation of the content.
- User difficulties in:
  - Locating relevant information;
  - Interpreting search results;
  - Navigating through the content;
  - Creating a mental model of the information structure.

Service-oriented approach

The service-oriented approach **models an information space on the basis of services** that work on data structures/objects and processes, which describe sequences of steps and the services and data involved in each step.

Components can consist of objects, services and processes that are related to each other, e.g. a component can organise the operation of other components.

Challenges

What components are needed

How they should be connected so that they have minimum dependencies in order to be recombined for different purposes.

Identify what services components should offer.

A service can be used to connect one component with another, or as a method applied on an object.

Service-oriented approach –Why?

• Facilitates the integration of commercial, in-house and open source components and applications within organisations and regional federations by agreeing upon common service definitions, behaviours, data and user models, and protocols.

• Enable alignment with business processes

• Offer flexibility to accommodate evolving organisational/user requirements as they provide a flexible and modular technology base that makes information sharing of applications simpler and allow collaborative organisations to deploy applications that meet their common needs.

Service-oriented approach for personalisation

Challenge

• Provide personalisation on the basis of well defined service behaviours and interfaces

• Allow various open specifications, open source toolkits and standards to be used in implementing the services.

Service-oriented approach for personalisation

- **Flexible personalised systems**: components can be added, removed or replaced more easily than in traditional models of adaptive hypermedia systems, and where new applications or systems can be composed from collections of available services.

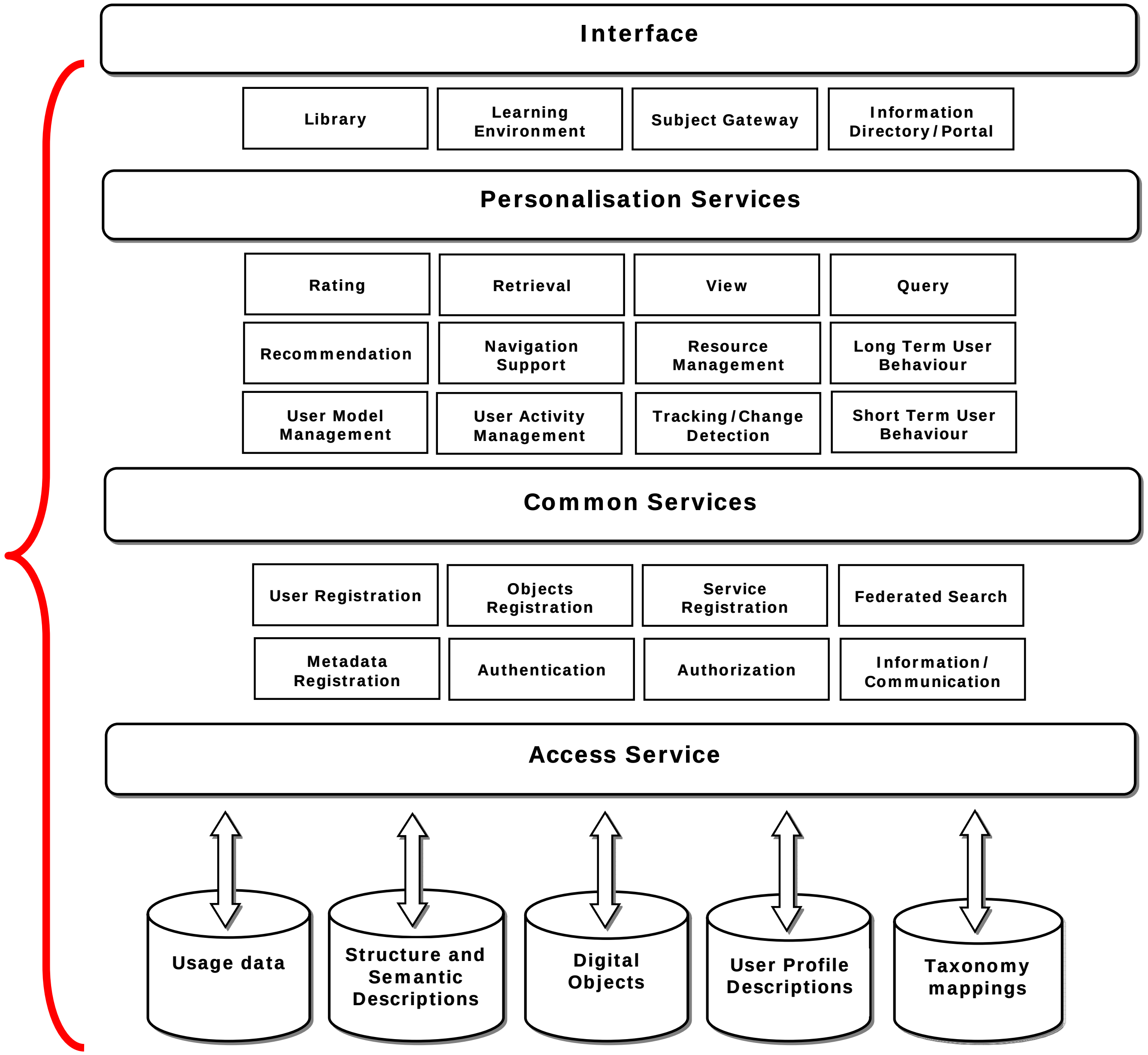
- **Faster deployment of personalisation technologies:** as long as the needs of new components are compatible with the existing component interfaces.

This approach is *different* from integrating directly at the user interface level (e.g. by using portals) or at the data level (e.g. by creating large datasets or data warehouses).

Service-based architecture-1/3

Set of identified services

The services are organised into logical groups but no explicit association among service functional definitions is implied.



Service-based architecture-2/3

Service	Scope
Rating	Creates and manages annotations for objects. Support for the use of secondary metadata (user ratings and text annotations) for resources.
Retrieval	Personalised information seeking. Browsing though the digital objects based on taxonomies descriptions. Augmented keyword-based by exploiting metadata properties of the objects according to user goals, user preferences, cognitive style etc. .
Recommendation	Allows applications that use this service to recommend information content based on application-specific user history and behaviour, and metadata descriptions.
Query	Provides query facilities over structured and semantic descriptions. Narrowing down or broadening the queries depending on interests and previous search history, information seeking strategies of users and user behaviour.
User model management	Support the management of individual and group user models. Includes policies for updating and registering user models. Initialises services with user preferences information.

Sample of services

Service-based architecture-3/3

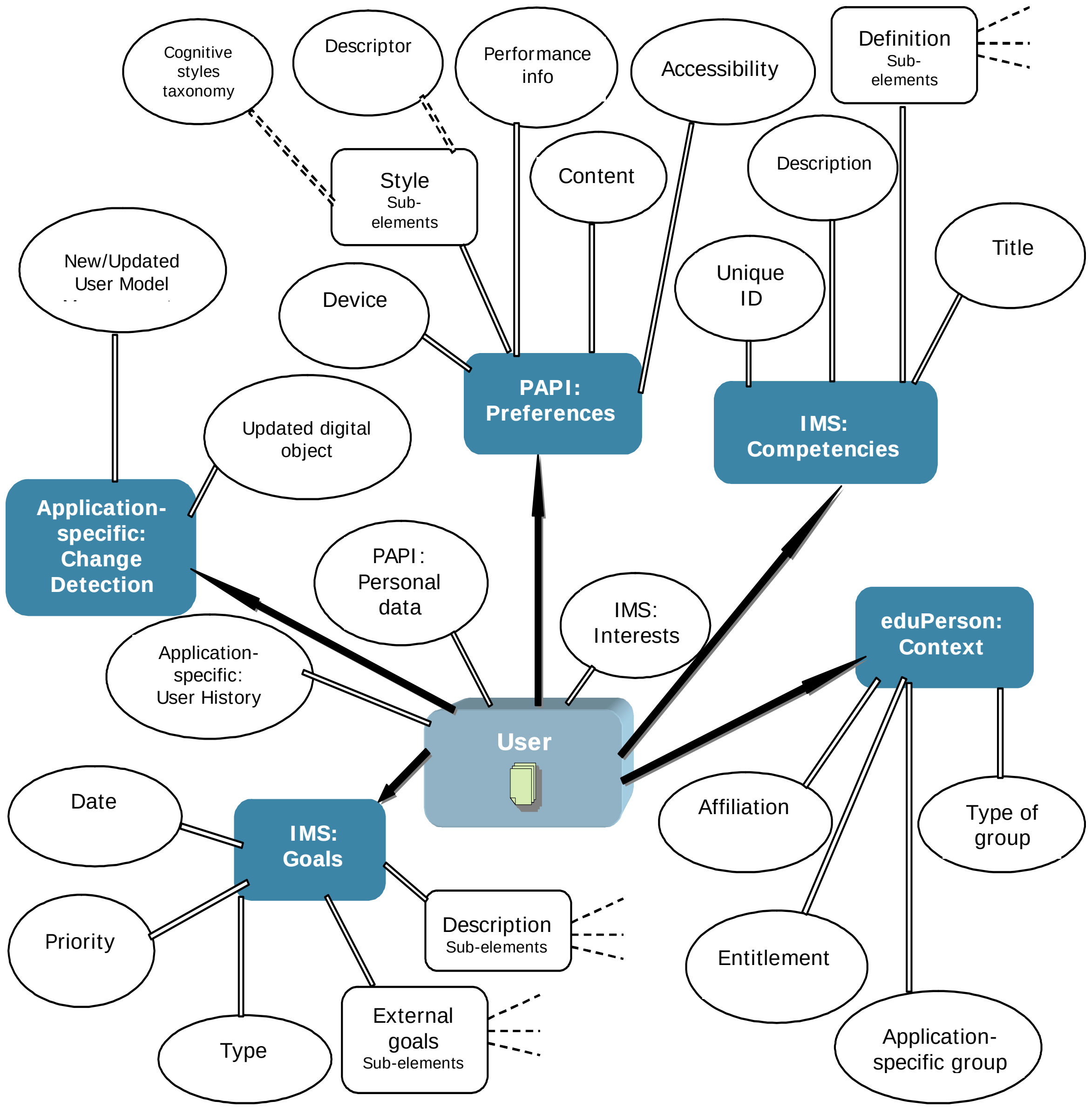
From the functional definition and scope of a specific service an abstract model of behaviour and data can be developed, which describe the expected behaviour of a realisation of this service and the data model (e.g. using XML) it deals with or exchanges.

A service can be realised in a number of ways, such as a Web service (e.g. using WSDL) and Application Programming Interfaces for particular programming languages. The various services interact to provide the complete functionality

User data models

A variety of data models are available for describing user aspects, such as OUNL-EML, PALO, PAPI, **IMS-LIP**, **ARIADNE**.

Encoding user profiles in RDF provides flexibility to include elements from multiple schemata, to enrich them with additional elements, when necessary, and to maintain interoperability with other systems



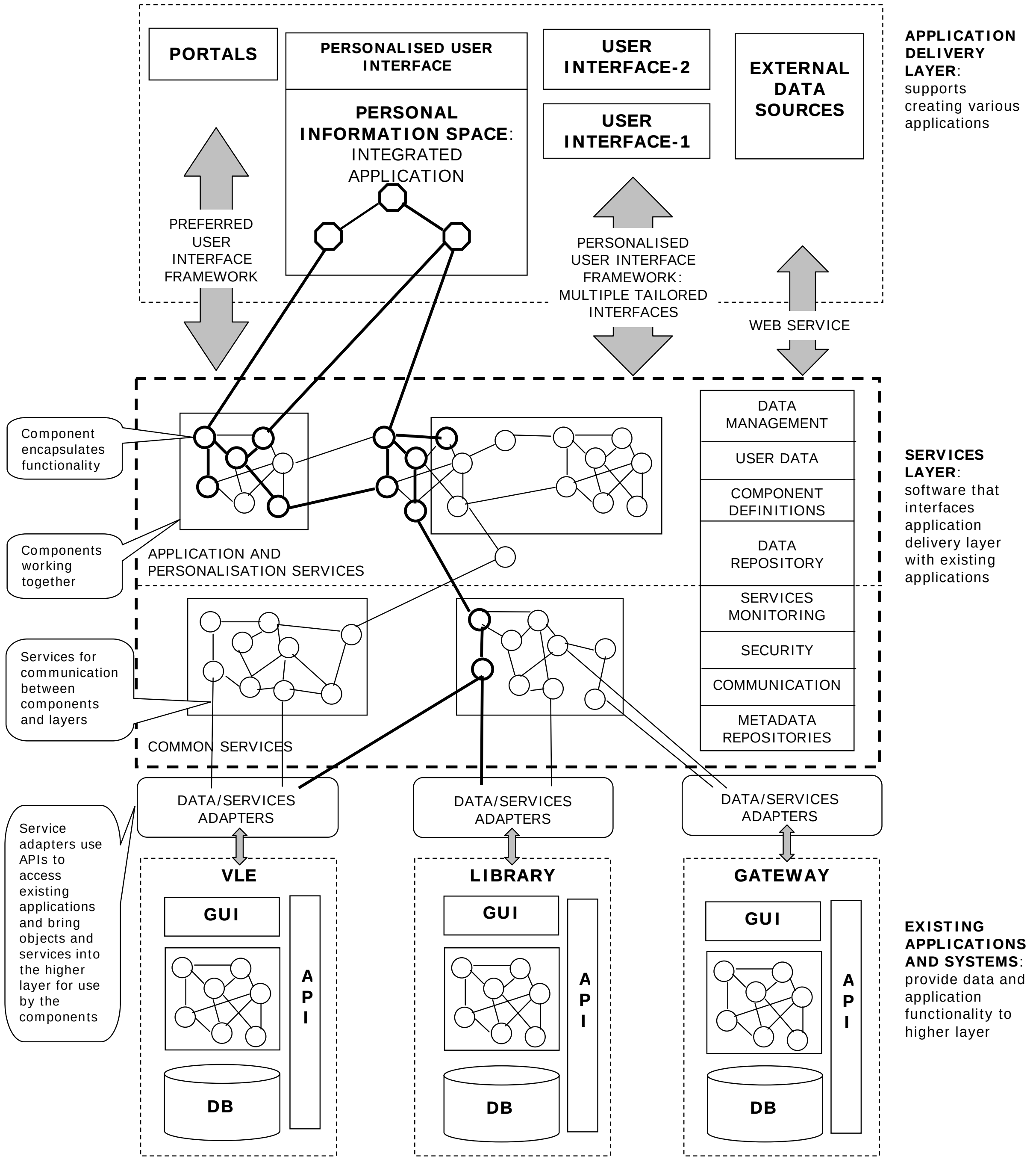
Conclusions and future work-1

- The service-oriented approach models an information space on the basis of **services**, which work on data structures/objects, and **processes** that describe sequences of steps and the services and data involved in each step, in order to tailor the information and the interface to the needs of the individual.
- Personalisation in this context emerges through the **aggregation of services** that implement a personalised function. It can also be materialised by **creating, managing and storing "personal views" or relationships** between information from a diverse set of existing applications

Conclusions and future work-2

Low level services, such as an authentication service, can be considered as general purpose; they do not rely on other services, and are standardised across all applications.

high-level services include services, processes and objects/data structures that are shared across applications, aggregate low level services functionality, manage user/application data, define processes, control objects/services


PORTALS
PERSONALISED USER INTERFACE
PERSONAL INFORMATION SPACE: INTEGRATED APPLICATION
USER INTERFACE-2
USER INTERFACE-1
EXTERNAL DATA SOURCES
APPLICATION DELIVERY LAYER: supports creating various applications
PREFERRED USER INTERFACE FRAMEWORK
PERSONALISED USER INTERFACE FRAMEWORK: MULTIPLE TAILORED INTERFACES
WEB SERVICE
Component encapsulates functionality
Components working together
APPLICATION AND PERSONALISATION SERVICES
DATA MANAGEMENT
USER DATA
COMPONENT DEFINITIONS
DATA REPOSITORY
SERVICES MONITORING
SECURITY
COMMUNICATION
METADATA REPOSITORIES
SERVICES LAYER: software that interfaces application delivery layer with existing applications
Services for communication between components and layers
COMMON SERVICES
DATA/SERVICES ADAPTERS
DATA/SERVICES ADAPTERS
DATA/SERVICES ADAPTERS
Service adapters use APIs to access existing applications and bring objects and services into the higher layer for use by the components
VLE
LIBRARY
GATEWAY
GUI
API
DB
EXISTING APPLICATIONS AND SYSTEMS: provide data and application functionality to higher layer

londonknowledgelab