



the globus alliance

www.globus.org



A Bright Future with OGSA Data Services

Malcolm Atkinson
Director

www.nesc.ac.uk

7th June 2004

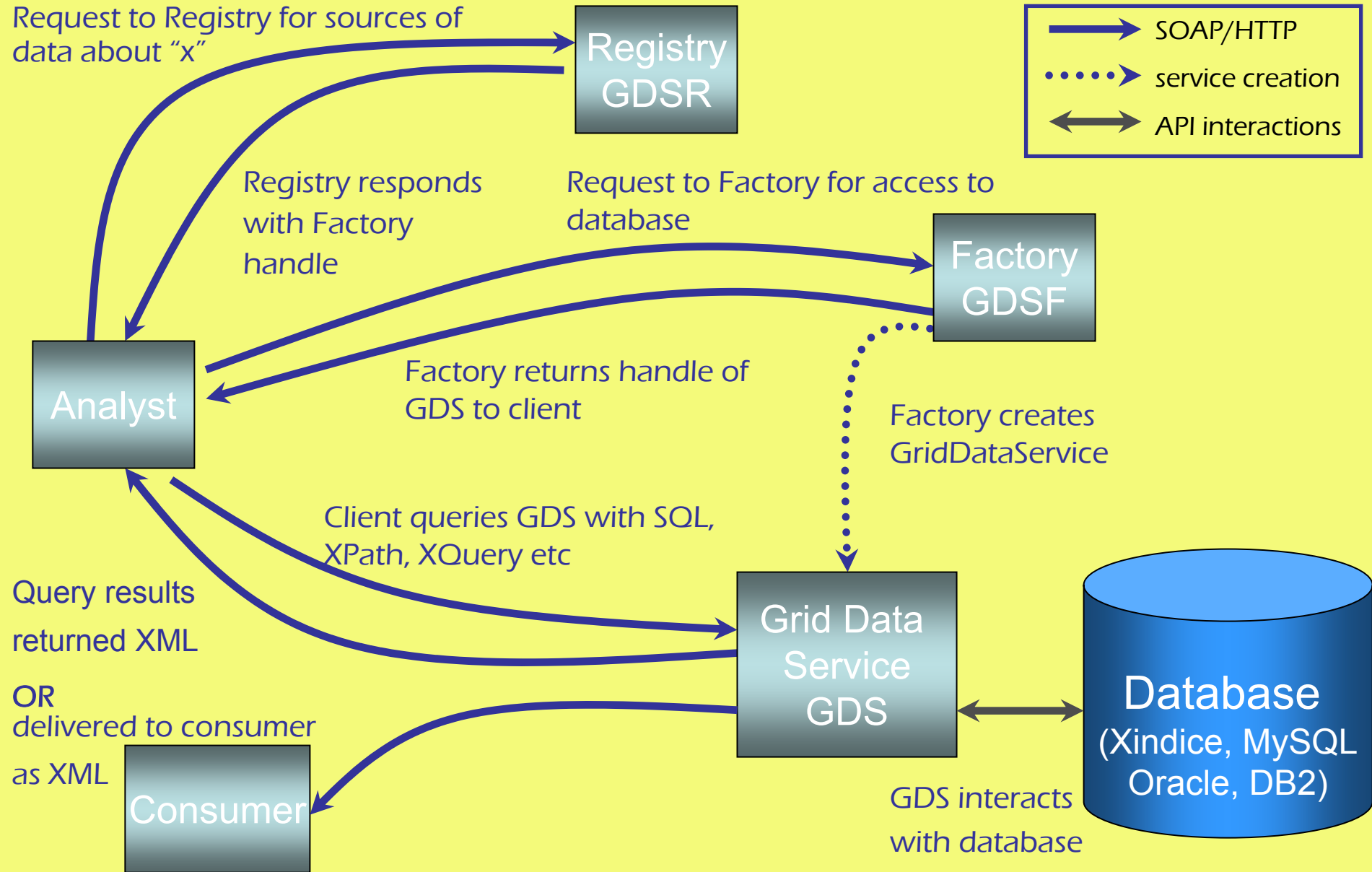


| epcc |



UNIVERSITY
of
GLASGOW

OGSA-DAI





Extensibility

- **Data resources**
 - Unbounded variety
- **Data access languages**
 - Established standards
 - ▶ With many variants
 - SQL, OQL, semi-structured query, domain languages
- **Investment in DBs, DBMSs, File Stores, Bulk stores, ...**
 - Not sensible to expect them to change to fit us
- **Data Access Models must be extensible**
 - Static extension used extensively by OGSA-DAI users

Should extensibility be supported by foundation interfaces?



Move Computation to Data

- **Code scale**
 - Depends on wet-ware
 - ▶ No noticeable rate of improvement
- **Data scale**
 - Grows Moore's Law or Moore's Law²
- **Analysis of data**
 - Extracts & derivatives used
 - ▶ Often smaller - more value for current investigation
- **Implies move code to data**
 - SQL, Xquery, Java code, ...
- **Extensibility mechanisms used by OGSA-DAlers**
- **Java mobility (e.g. DataCutter), database procedures, ...**

**Increasingly
necessary**

**Application
control or
higher-level
service
decisions**

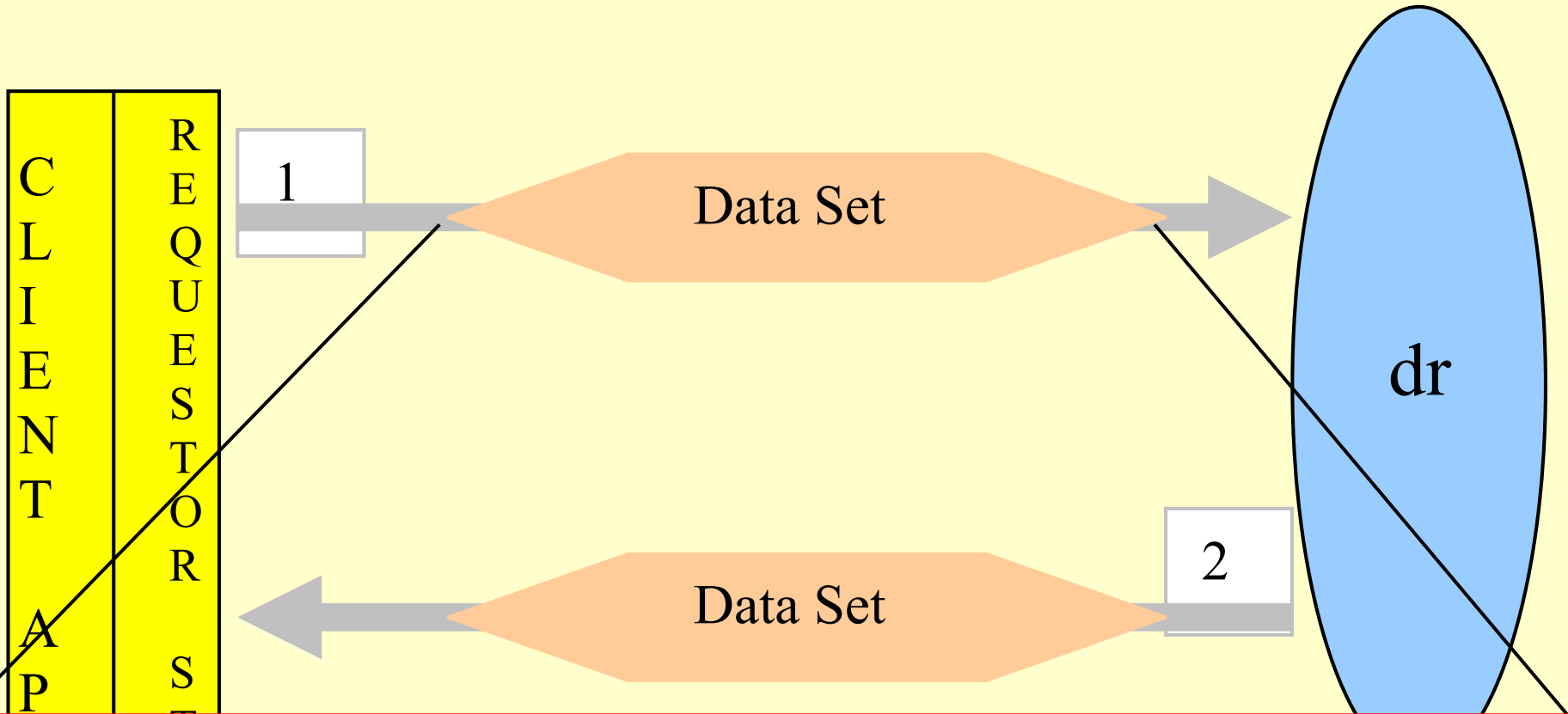


Integration is Everything

- No business or research team is satisfied with one data resource
- Domain-specialist driven
 - Dynamic specification of combination
 - Iterative processes - range of time scales
- Sources inevitably heterogeneous
 - Content, structure & policies time-varying
- Robust & stable steerable integration
 - Higher-level services over multiple resources
 - Fundamental requirements for (re)negotiation

Federation or Virtualisation preceding integration or kit of integration tools to be interwoven with an application?

Multiple tasks / request



Ident Type 7 Value	Ident Type 6 Value	Ident Type 5 Value	Ident Type 4 Value	Ident Type 3 Value	Ident Type 2 Value	Ident Type 1 Value	Ident Type 0 Value
--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------



Be Direct

- **Double Handling costs too much**
 - Memory cycles, bus capacity, cache coherency
- **Double Handling via discs pathological**
- **Data translation expensive**
 - Avoid or compose
- **Main memory is not big enough**
- **Couple generator & consumer directly**
 - Data pipe from RAM to RAM
 - Requires coupled computation execution

Breaks down boundaries and merges data, execution & transport requirements.

Demands smart workflow enactment service & foundation services