



# **CountryData Technologies for Data Exchange**

Introduction to SDMX Registry



# SDMX Registry (a.k.a. SDMX Registry/Repository)

- “...tasked with providing structure, organization, and maintenance and query interfaces for most of the SDMX components...”
- “...provides support for the submission and retrieval of structural metadata.”



# Registry Operations

- Receive submissions, store, disseminate structural metadata (DSD, codelists,...)
- Receive registrations of data and reference metadata sources, disseminate links to those sources
- Provide subscriptions, send users notifications of update to subscribed data / metadata



# Registry or Repository?

- SDMX Registry does not store any data. It can only store *registrations*, i.e. pointers to data located anywhere on the Web.
- SDMX Registry does, however, act as a repository for *structural metadata*.
- In other words, DSDs, Concepts, etc, uploaded to the SDMX Registry are actually stored there, unlike data.



# Interaction With the Registry

- All interactions can be done with SDMX messages submitted through a Web service
- Most implementations also offer a visual user interface in the form of a Web application



# Authentication / Authorization

- Supported, but SDMX leaves the mechanism up to the implementer



# Subscriptions and Notifications: How they work

- *Maintenance Agency* can upload structural metadata to the Registry.
- *Data Provider* can register SDMX data.
- *Data Consumer* can subscribe to various events, e.g.:
  - Structural Metadata update
  - Data registration
  - Metadata registration



# Subscriptions and Notifications: How they work

- When a *Data Provider* has updated their data, they send a message to inform the Registry that the data source has been updated.
- The registry then notifies subscribers through their chosen channel (either email or HTTP).






# Subscriptions and Notifications: How they work

- When *Maintenance Agency* updates SDMX artefacts, the registry will automatically notify all *Data Consumers* subscribed to this type of event.
- Notification messages carry the URL of data or metadata that has been updated. Based on this, recipients can act to update their own systems.



# Registry: Pull Model

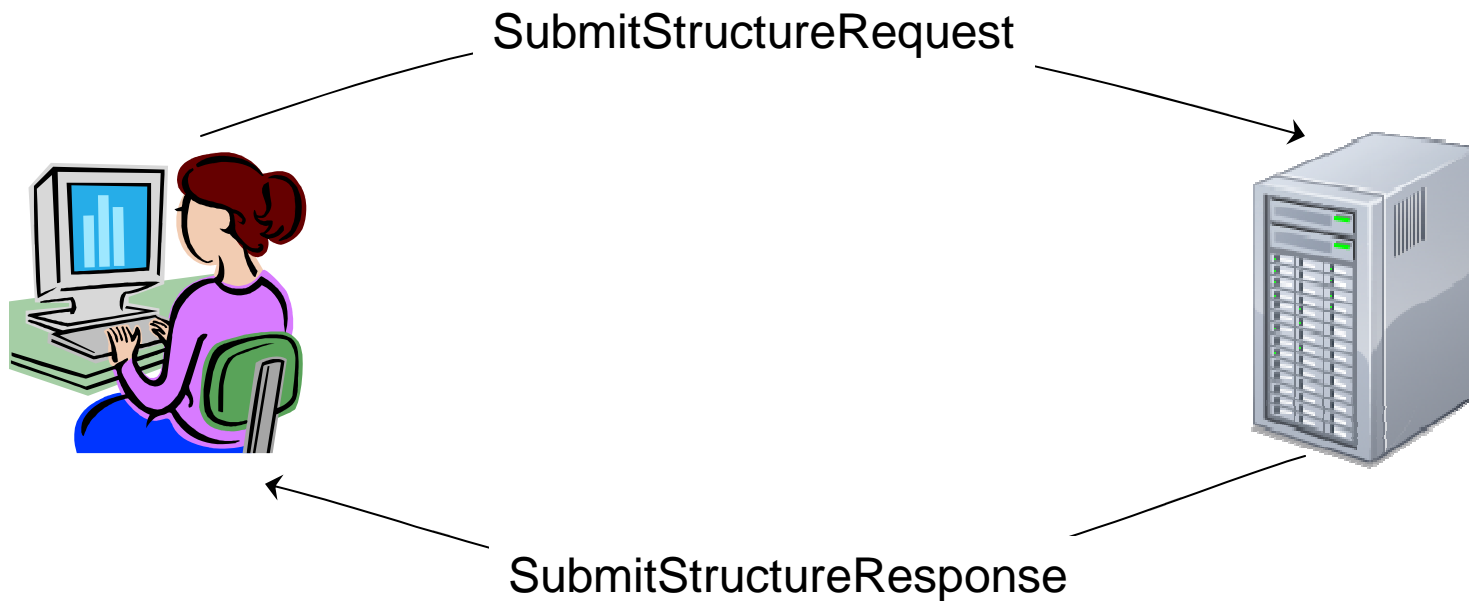
- Registry is an implementation of *pull model* for data exchange.
- Data provider never sends data to recipients. The recipient is free to consume the data or metadata as convenient.




# Registry Services: Structure Submission

- Supports submission of and querying for structural metadata
- Interfaces used:
  - SubmitStructureRequest
  - SubmitStructureResponse

# Registry interaction: Upload structural metadata

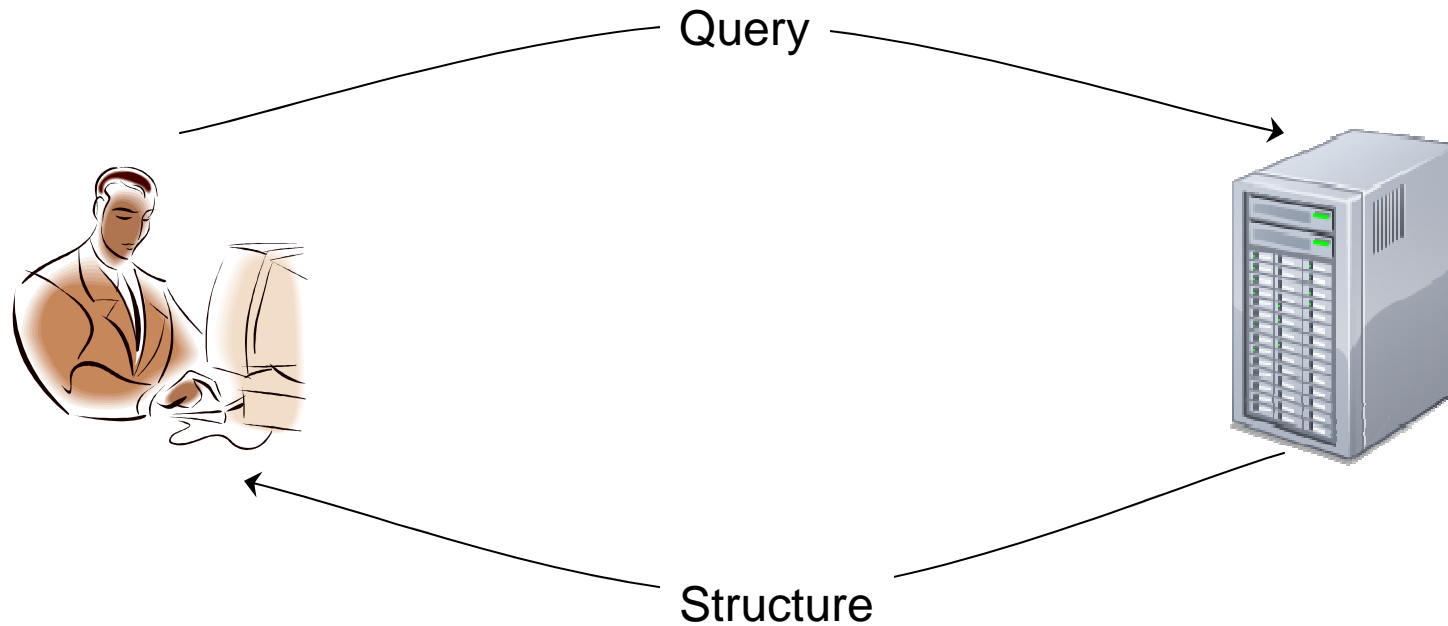




# Registry Services: Structure Query

- Supports a mechanism for querying for structural metadata.
- Submit SDMX-ML Query message, receive Structure message in response
  - Query message described here is for structural metadata only – not for data or reference metadata

# Registry interaction: Query for structural metadata

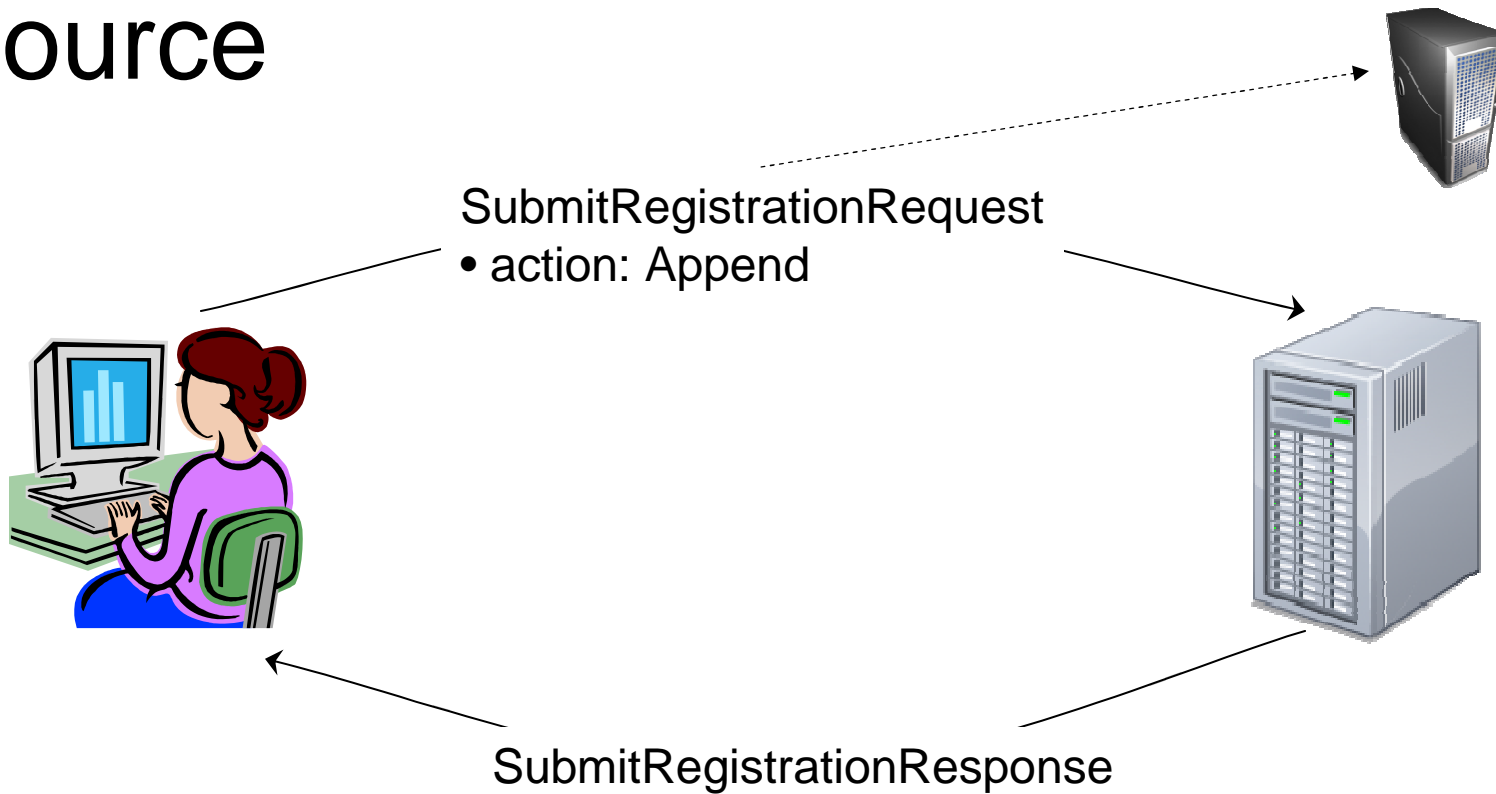





# Registry Services: Data and Reference Metadata Registration

- Supports registering and querying for data/metadata sources
- Interfaces used:
  - SubmitRegistrationRequest
  - SubmitRegistrationResponse

# Registry interaction: Register data source



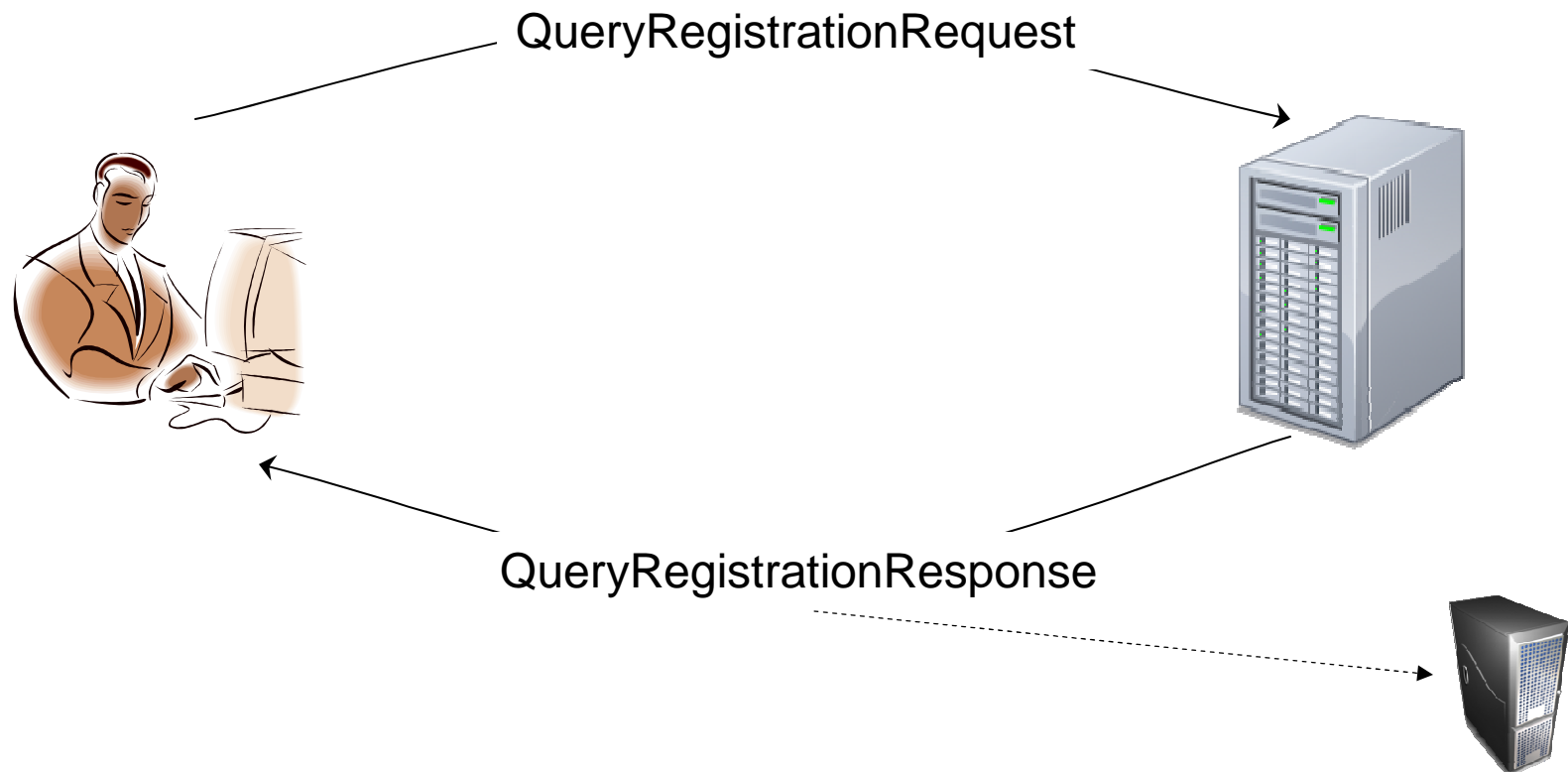




# Registry Services: Data and Reference Metadata Discovery

- Reports available data sources
- Interfaces used:
  - QueryRegistrationRequest
  - QueryRegistrationResponse

# Registry interaction: Discover data or reference metadata

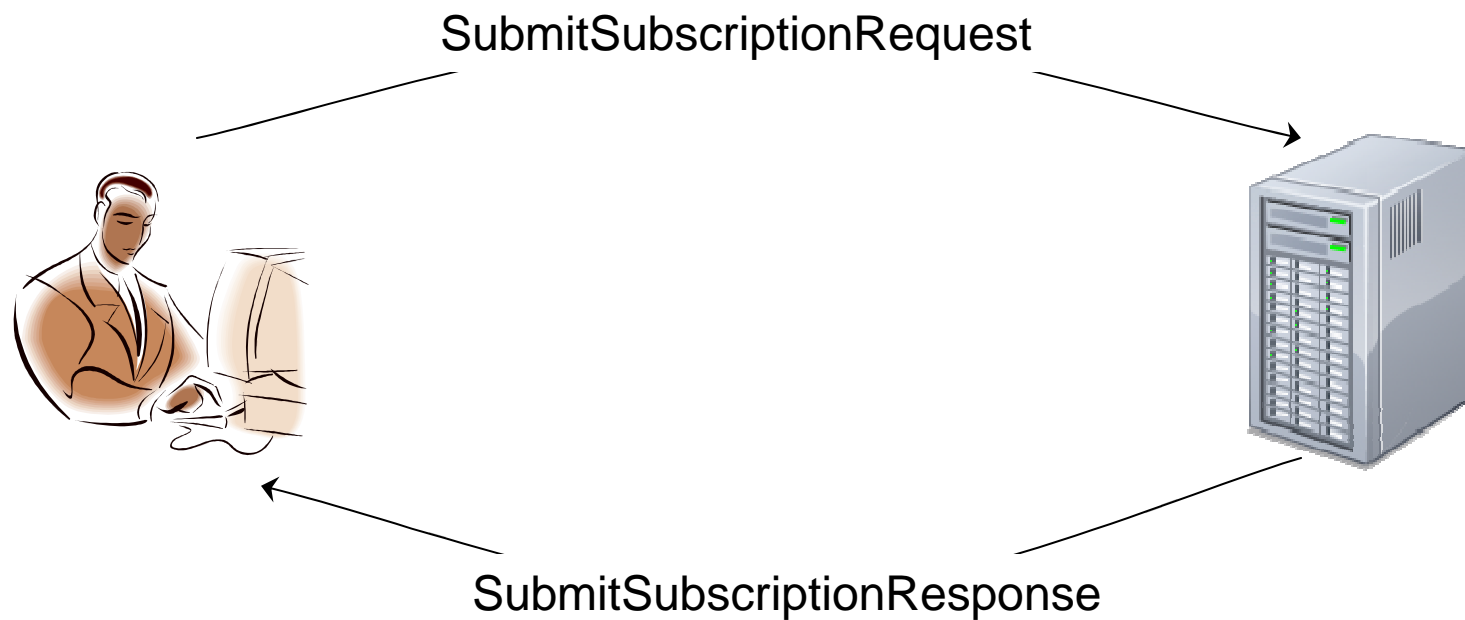




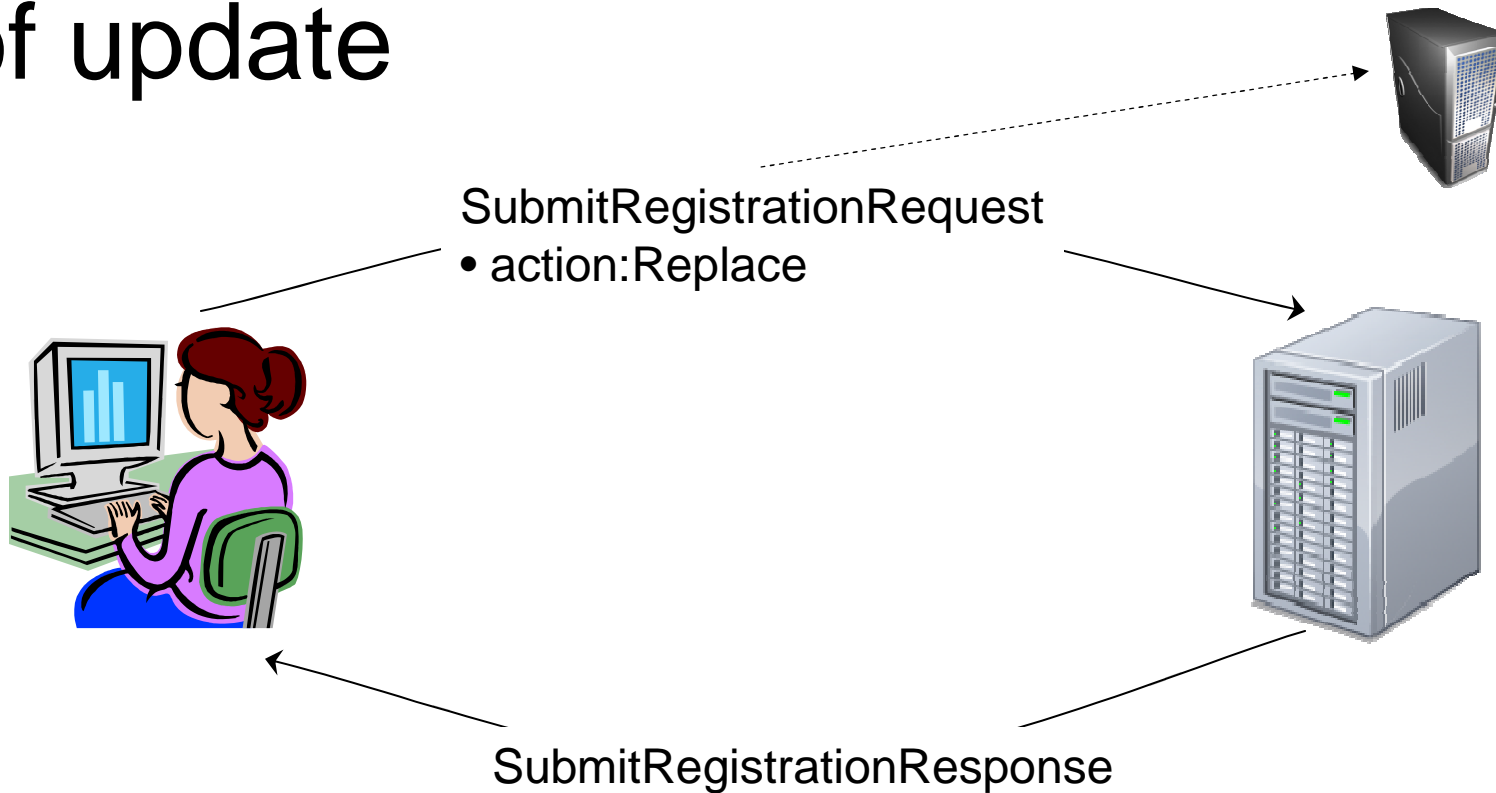
# Registry Services: Subscription and Notification

- Supports subscription to data, structural metadata, and reference metadata update events
- Interfaces used:
  - SubmitSubscriptionRequest
  - SubmitSubscriptionResponse
  - NotifyRegistryEvent

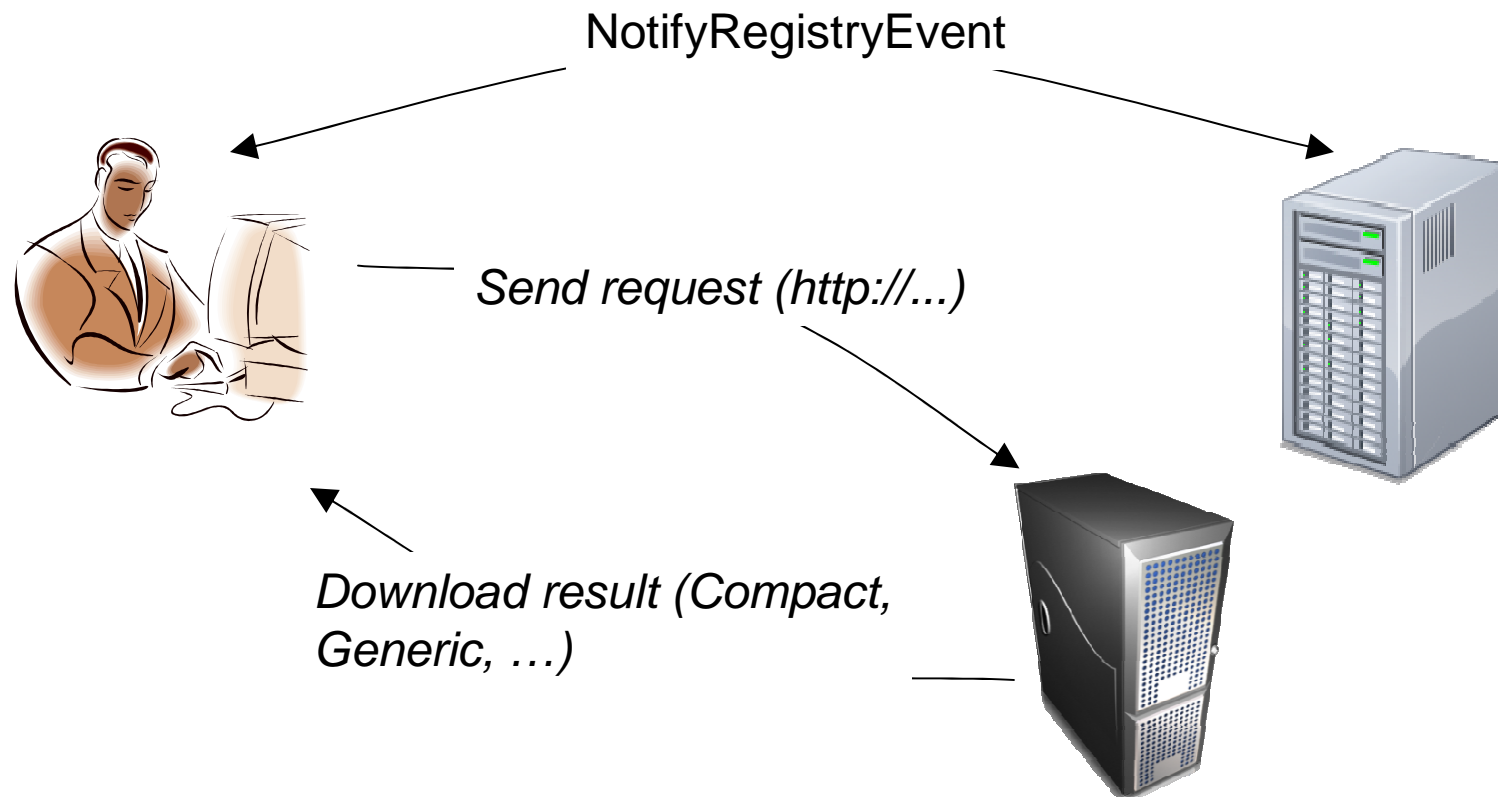
# Registry interaction: Subscribe to update events



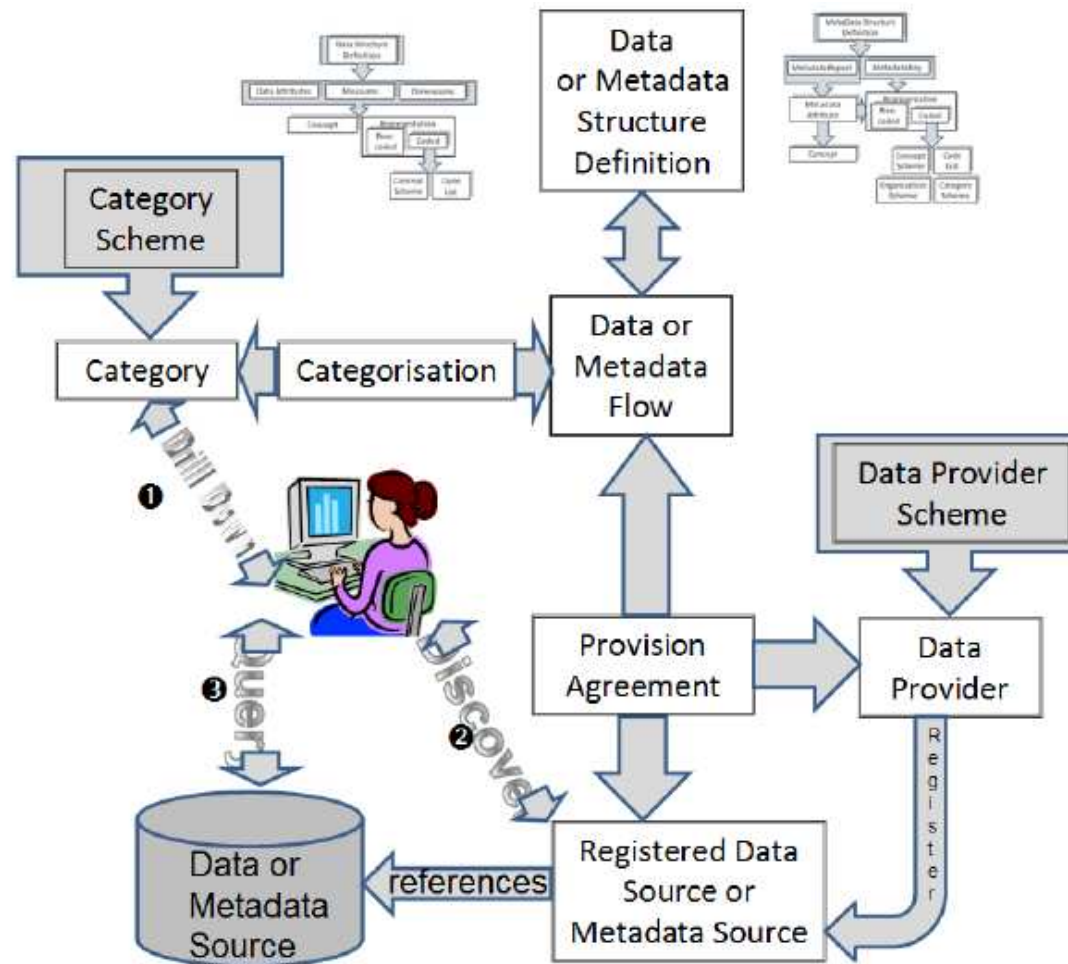
# Registry interaction: Inform registry of update



# Registry interaction: Receive notification, download data



# Data/Metadata Discovery and Query



Source: **SDMX Registry Specification: Logical Functionality and Logical Interfaces (SDMX 2.1)**



# Principal Artefacts

- Data Structure Definition / Metadata Structure Definition
  - Define the structure of data or metadata that can be registered at the Registry
- Data flow / Metadata flow
  - Defines a subset (view) of a DSD / MSD
  - Constraints can be used to restrict permitted data





# Principal Artefacts (2)

## ■ Category Scheme

- Used to group Categories, such as subject-matter domains, to which dataflows and other artefacts can be linked to enable browsing
- E.g. a Data Flow that groups health indicators can be linked to Category “Health”
- User can then discover and use Data Flows that are linked to Health



# Principal Artefacts (3)

- Data Provider Scheme

- Contains Data Providers, i.e. agencies that publish data

- Provision Agreement

- Links a provider and data/metadata flow:  
Provision Agreements specify which providers have agreed to publish which Data or Metadata flows.



# Principal Artefacts (4)

## ■ Constraint

- Specifies a complete or partial key (dimension values) which can be used to subset other artefacts such as DSD, Data Flow, Provision Agreement
- Can specify allowed combinations or disallowed combinations of dimension values