IVOA interop - 18-23 May 2014 - Madrid, Spain

TAP services integration at IA2 data center

Pietro Apollo
INAF - Astronomical Observatory of Trieste

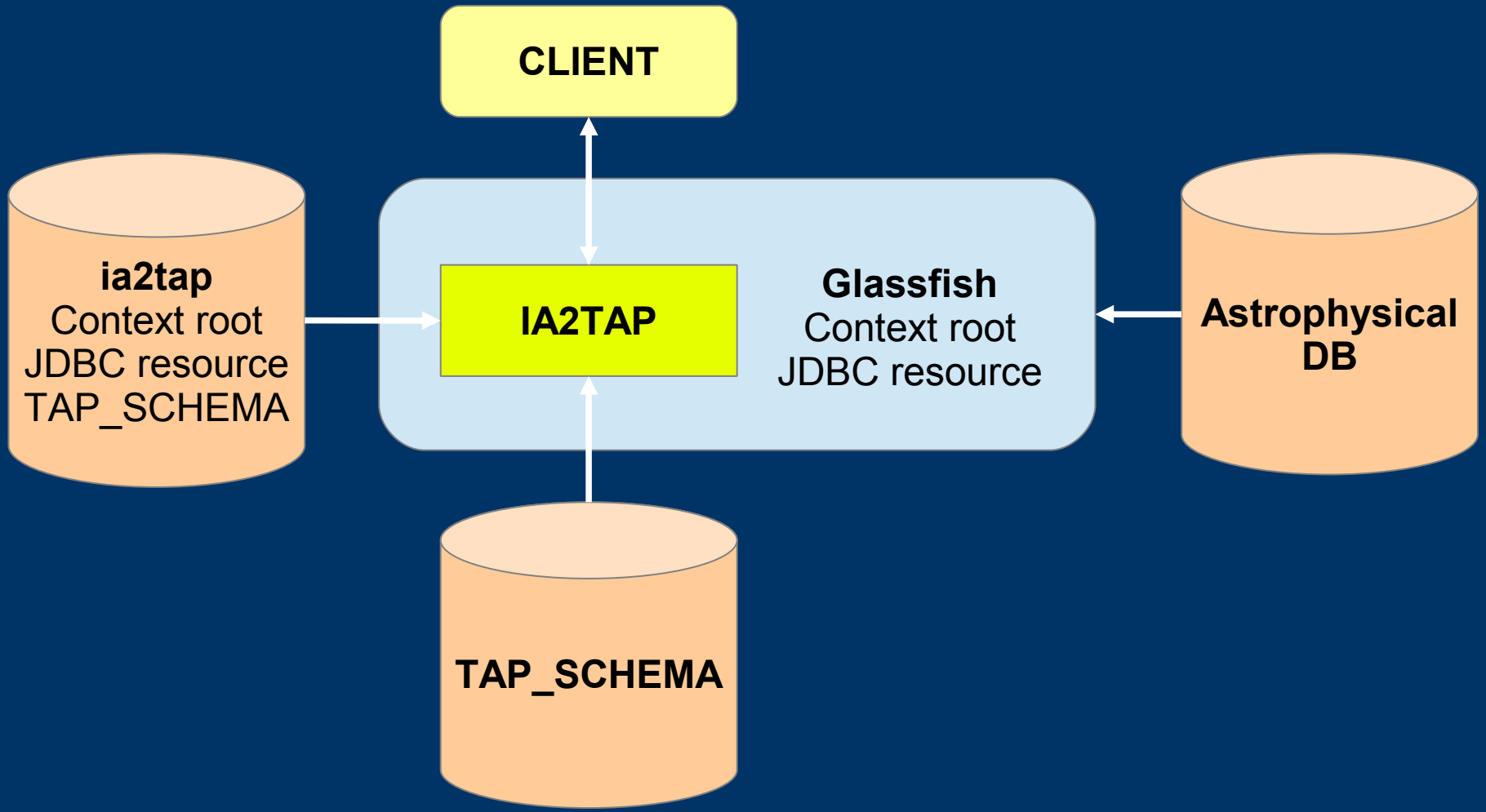
Outline

- IA2TAP:
  - TAP implementation
  - 2 + 1 + 1 services
- TapSchemaManager:
  - a supporting web application for managing TAP_SCHEMA schemata
- Summary

IA2TAP: general overview

- IA2TAP is a web application developed at the INAF – OATs IA2:
  - Coded in Java (JDK 1.6)
  - Runs on Glassfish web container (tested on version 3.1)
  - Uses a MySQL DBMS (tested on version 5.5):
    - to store its internal configuration such as service descriptions, job details, logging
    - to connect to the astrophysical archives
  - It has been developed on top of the openCADC libraries, to fulfill the main TAP requisites

IA2TAP block diagram



Dynamic endpoints

- IA2TAP implementation builds the endpoints (capabilities, availability) on the base of the configuration, instead of using XML configuration files
- The tables endpoint is dynamically built on the base of the metadata exposed by TAP_SCHEMA

IA2TAP: multiple TAP services

- The openCADC libraries have been customized to allow multiple TAP services running side by side on the same web server
- Multiple deployments of a unique WAR package are used to expose on the Glassfish web container different TAP services

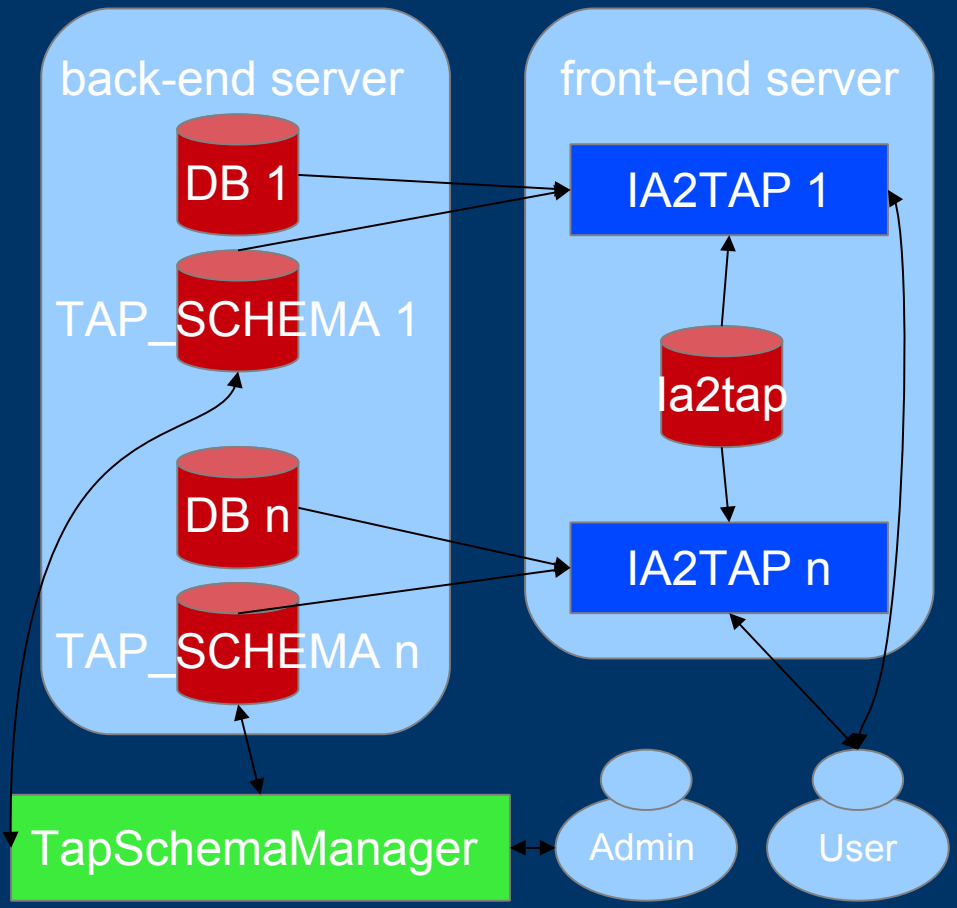
IA2TAP: multiple JDBC resources

- Customised implementation of openCADC libraries:
  - every TAP service:
    - extracts its own astrophysical data from the schemata available through a dedicated JDBC connection (provided by Glassfish)
    - loads the dedicated JDBC resource (connection pool) from the Glassfish web container as specified by *ia2tap* internal configuration

IA2TAP: multiple TAP_SCHEMA schemata

- Flexible TAP_SCHEMA naming clashes with the TAP specification:
  - each service uses a different TAP_SCHEMA schema:
    - i.e. a schema not named TAP_SCHEMA that acts like the TAP_SCHEMA
    - the schema name is stored in the internal configuration
- This IA2TAP feature joined with the recommended fixed schema name forces an ADQL modification at query level to interrogate the proper custom_TAP_SCHEMA
  - letting TAP_SCHEMA be an optional element of the TAP specification or allowing custom naming can solve this problem

TAP services architecture at IA2: diagram



- IA2TAP implementation allows deployment of tablesets served by MySQL servers running on servers, different from the one hosting the web application
- IA2 TAP services architecture separates back-end and front-end on different servers

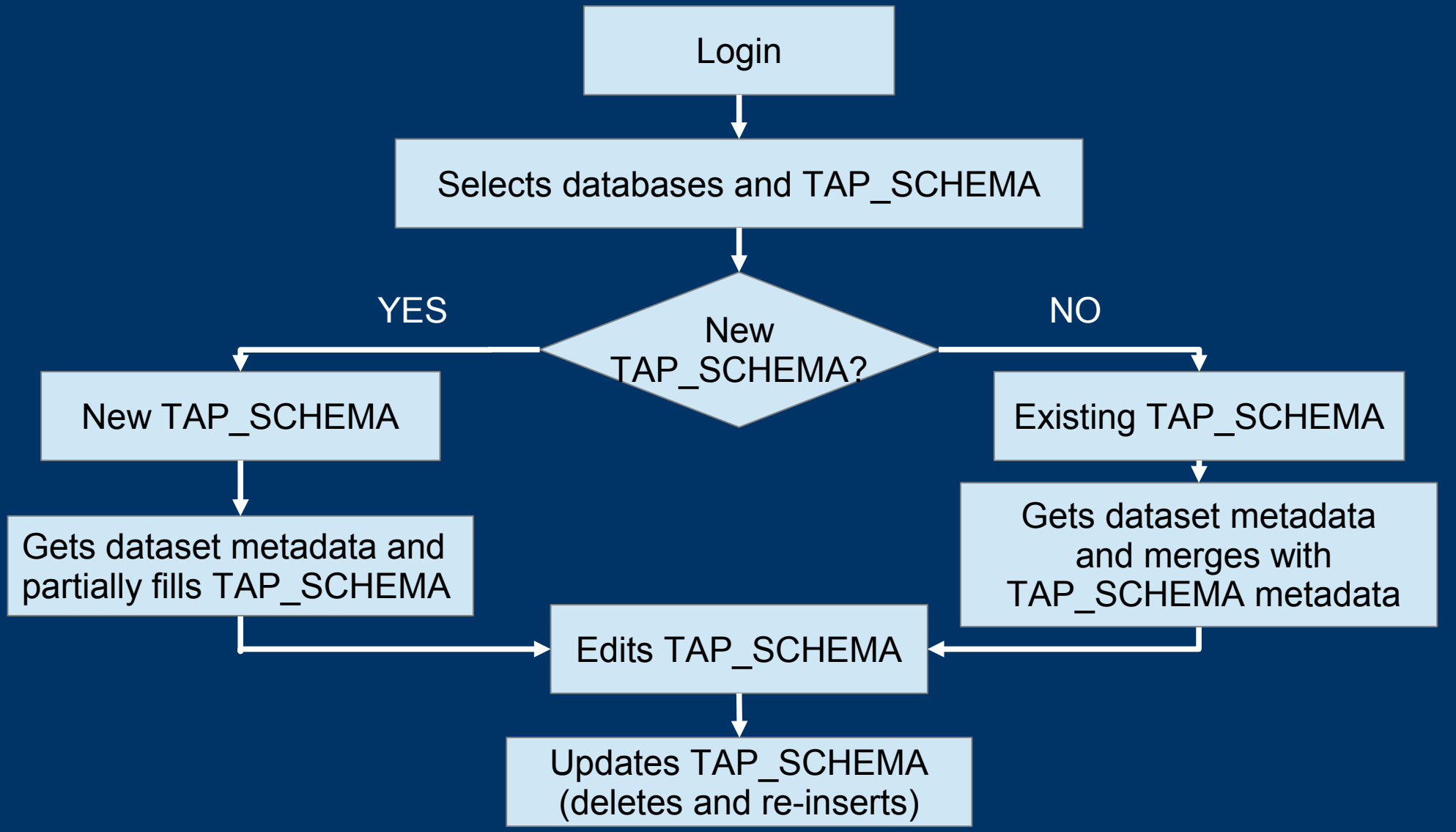
Deployed services

- IA2 registered TAP services:
  - Catalogue of WGE photometric redshifts for SDSS candidate quasars and galaxies
  - NASA dust catalogue (EPN-TAP);
- Custom internal TAP service for Asiago data
- Relational Registry Schema TAP interface:
  - Under development, not fully compliant

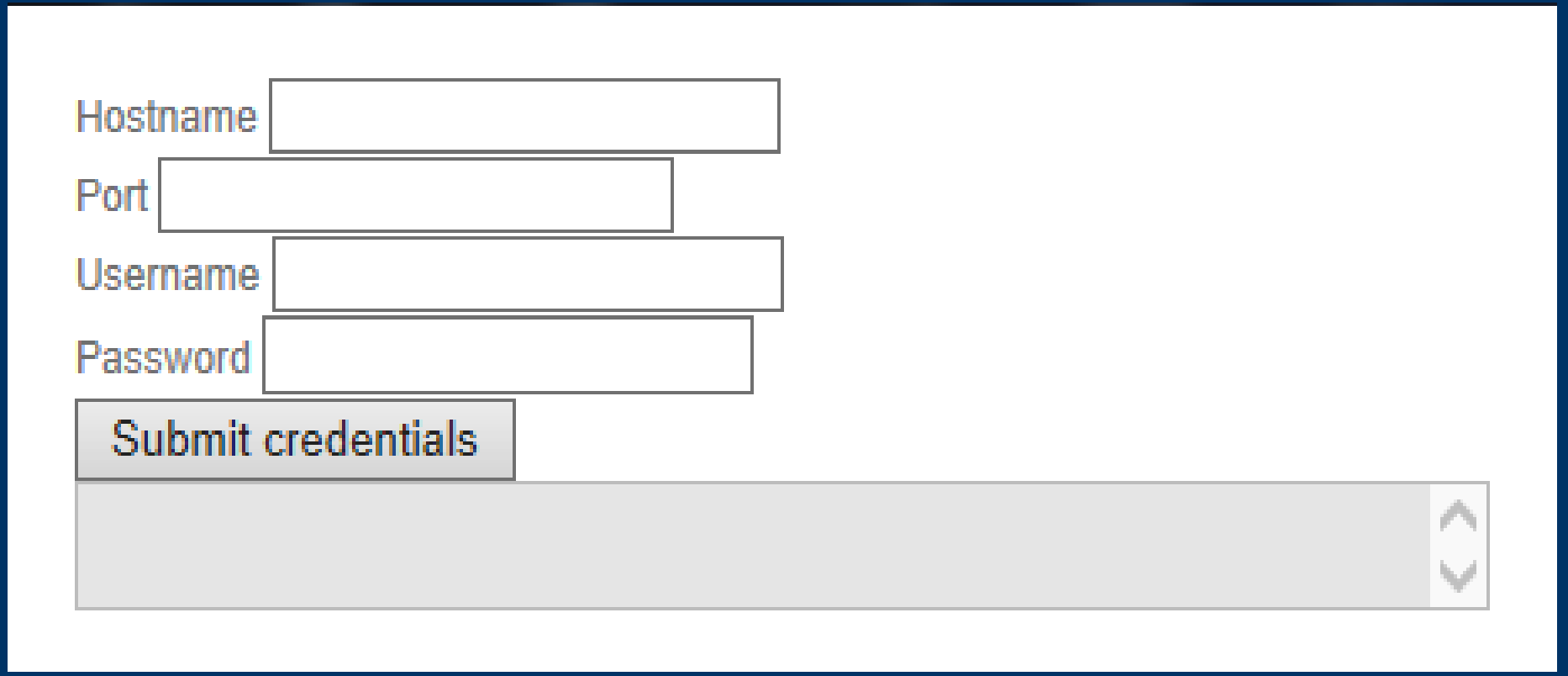
TapSchemaManager overview

- TapSchemaManager is a supporting web application
  - coded in Java
  - using the Wicket framework
- TapSchemaManager has a graphical interface so that a user can manage the TAP_SCHEMA schemata in a visual and flexible way

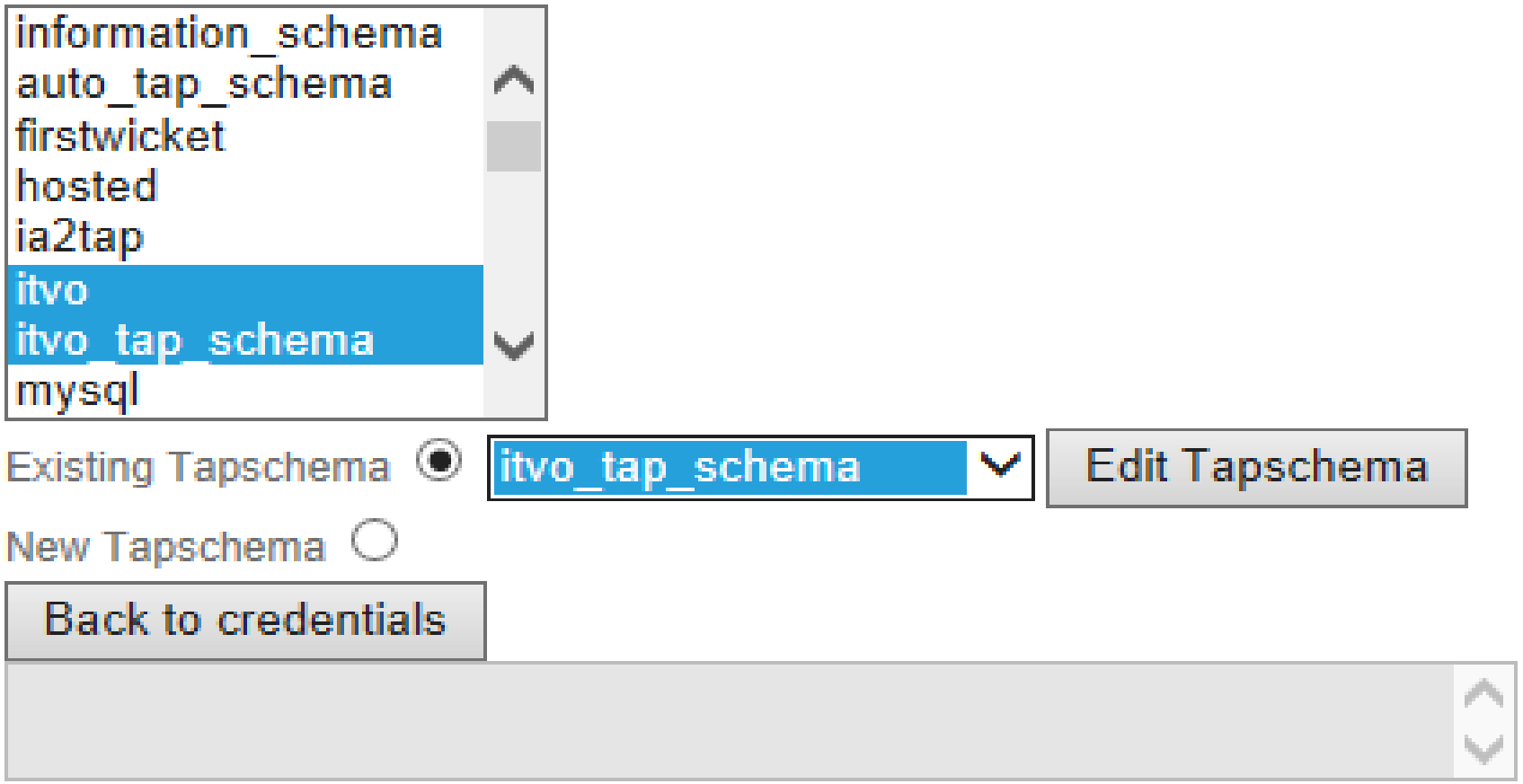
TapSchemaManager usage workflow



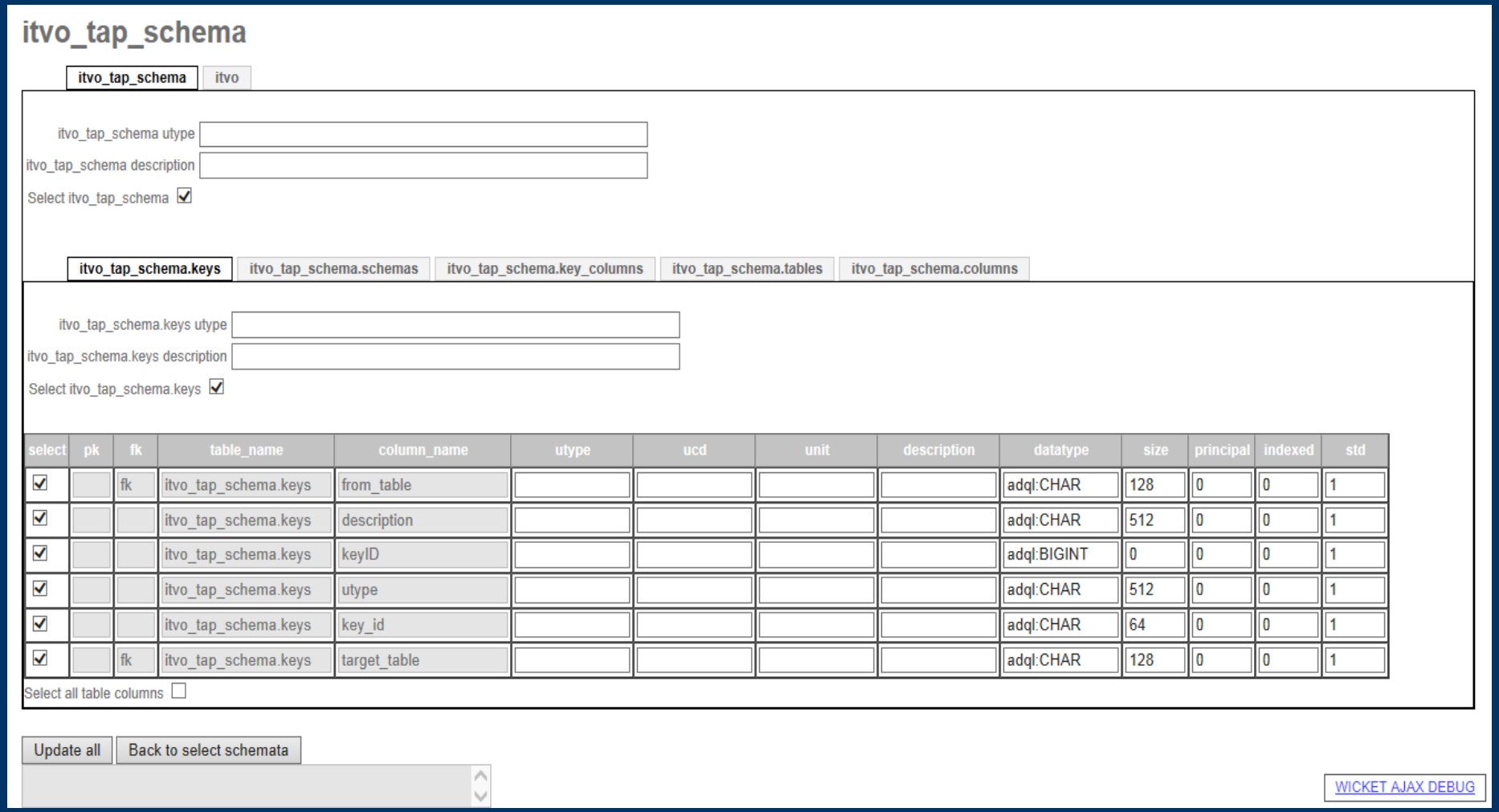
Login screenshot



Resource selection for TAP_SCHEMA generation screenshot



Metadata management page screenshot



itvo_tap_schema

itvo_tap_schema | itvo

itvo_tap_schema utype

itvo_tap_schema description

Select itvo_tap_schema

itvo_tap_schema.keys | itvo_tap_schema.schemas | itvo_tap_schema.key_columns | itvo_tap_schema.tables | itvo_tap_schema.columns

itvo_tap_schema.keys utype

itvo_tap_schema.keys description

Select itvo_tap_schema.keys

select	pk	fk	table_name	column_name	utype	ucd	unit	description	datatype	size	principal	indexed	std
☑		fk	itvo_tap_schema.keys	from_table					adql:CHAR	128	0	0	1
☑			itvo_tap_schema.keys	description					adql:CHAR	512	0	0	1
☑			itvo_tap_schema.keys	keyID					adql:BIGINT	0	0	0	1
☑			itvo_tap_schema.keys	utype					adql:CHAR	512	0	0	1
☑			itvo_tap_schema.keys	key_id					adql:CHAR	64	0	0	1
☑		fk	itvo_tap_schema.keys	target_table					adql:CHAR	128	0	0	1

Select all table columns

Update all | Back to select schemata

WICKET AJAX DEBUG

IA2TAP and TapSchemaManager future developments

- A better dynamic generation of the results of the TAP
- A more exhaustive logging system (e.g. statistics and more information about the execution of jobs)
- Implementation of the ADQL geometrical functions using mysql_sphere library
- Integration of IA2TAP internal configuration with the VO-Dance one
- Integration of TapSchemaManager into an administration suite for both the VO-Dance and IA2TAP service publishers
- Guided identification of UCDs, UTYPEs and UNITs to fill the created TAP_SCHEMA

Thank you for your attention
Pietro Apollo (and the IA2 team)