

# MySlice overview

Jordan Augé, Loïc Baron (UPMC)

OpenLab plugfest – January 23-25, 2013 – Paris, France



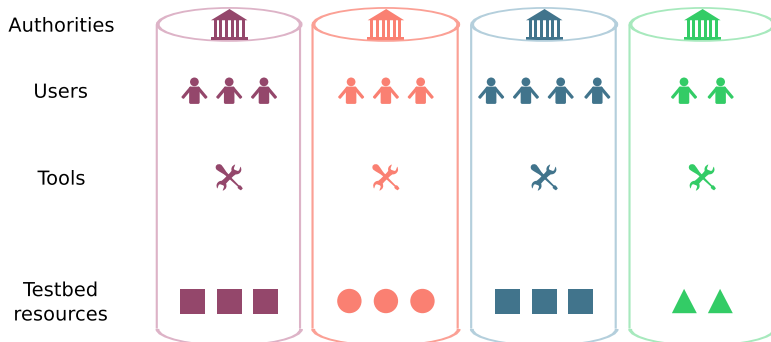
# Outline

- 1 Overview of MySlice
- 2 Extending MySlice with Gateways
- 3 Extending MySlice with plugins

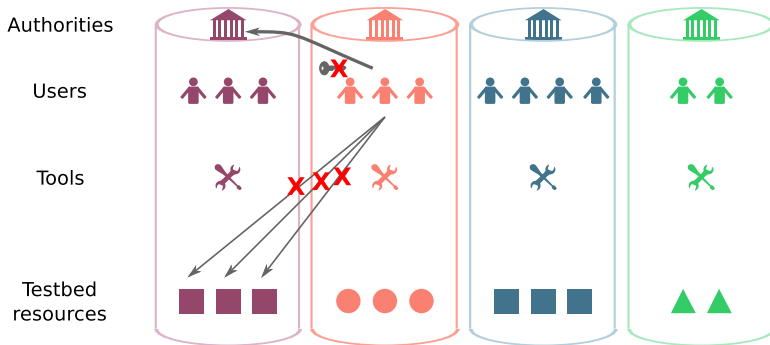
# Overview of MySlice

- A **user-centric** tool to support users' interaction with the federation of testbeds
- tailored to support the full **experiment lifecycle**
- based on an **open and extensible** framework

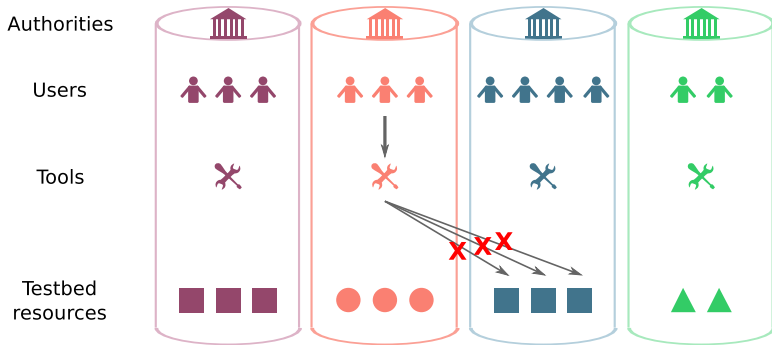
# Where we were recently



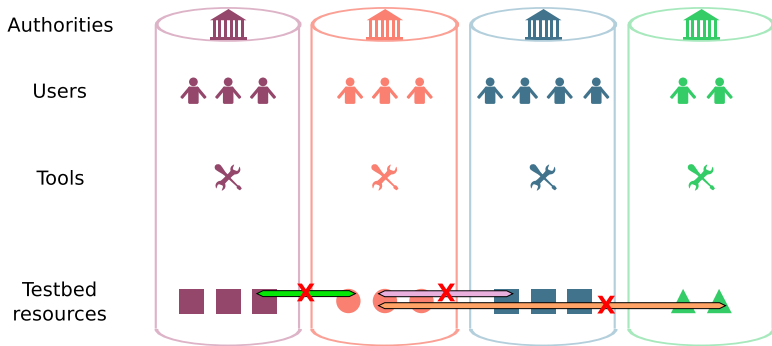
# Where we were recently



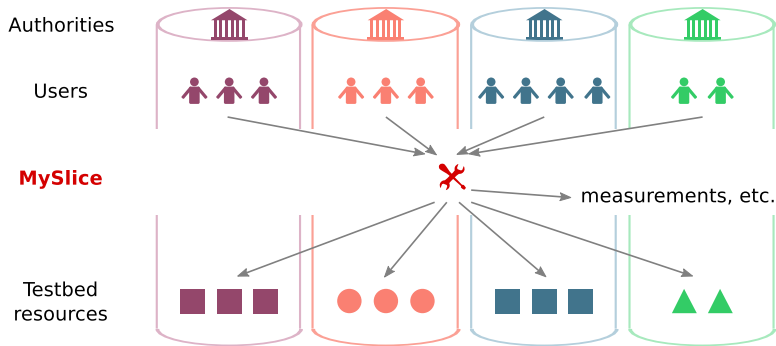
# Where we were recently



# Where we were recently



# Where we were recently

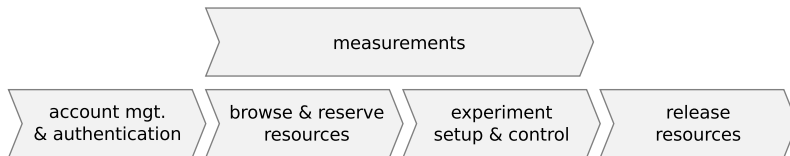




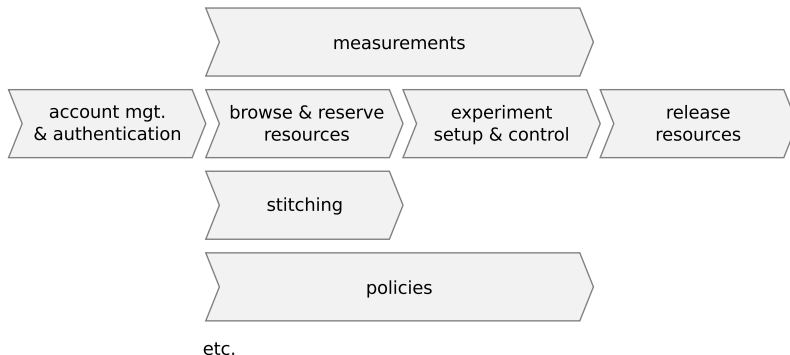
# Hiding the complexity of the experimental lifecycle



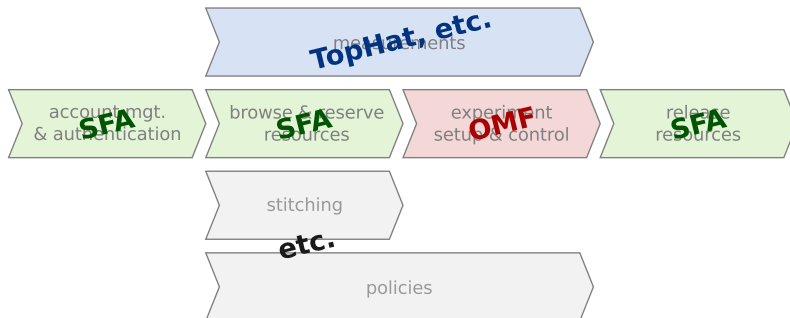
# Hiding the complexity of the experimental lifecycle



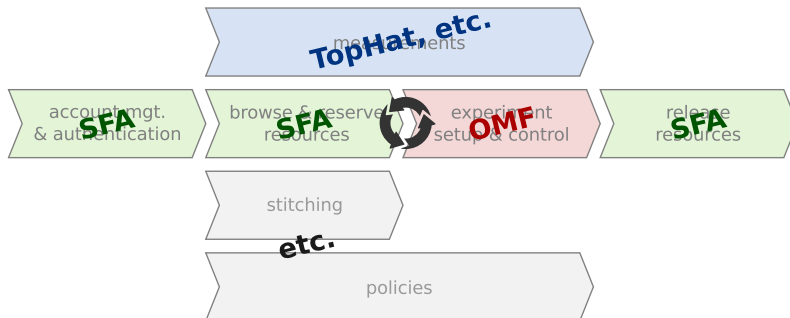
# Hiding the complexity of the experimental lifecycle



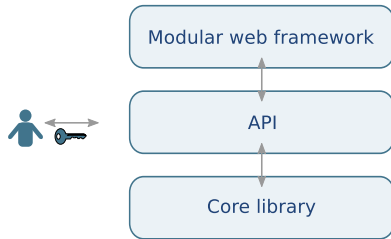
# Hiding the complexity of the experimental lifecycle



# Hiding the complexity of the experimental lifecycle

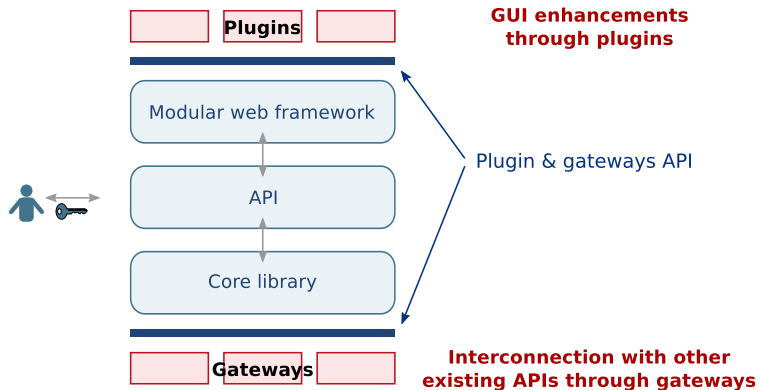


# MySlice architecture

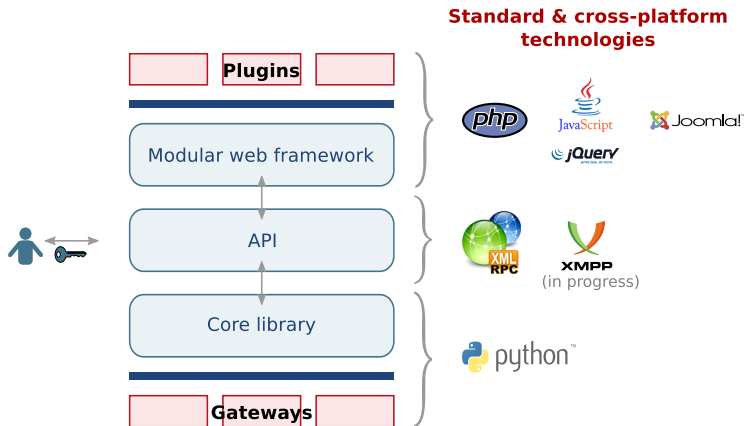


**A wide-range of user access interfaces to accommodate the diversity of users' needs**

# MySlice architecture

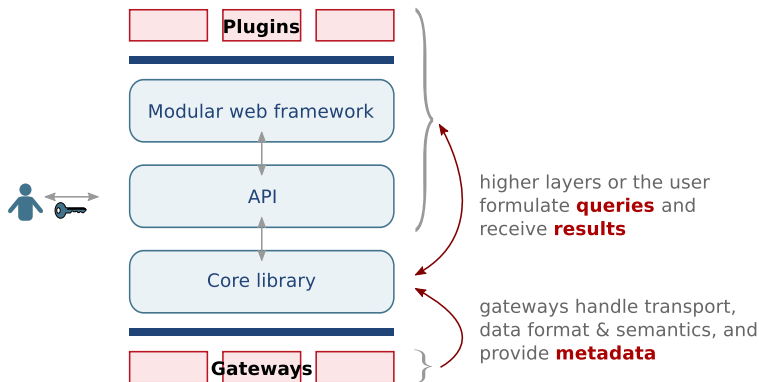


# MySlice architecture

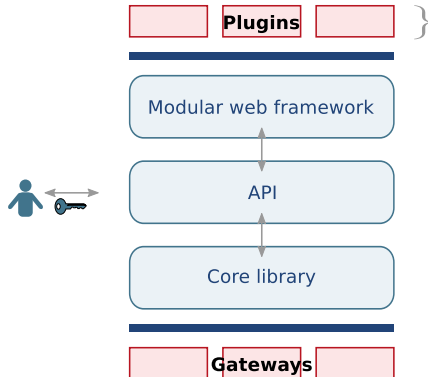




# MySlice architecture



# MySlice architecture



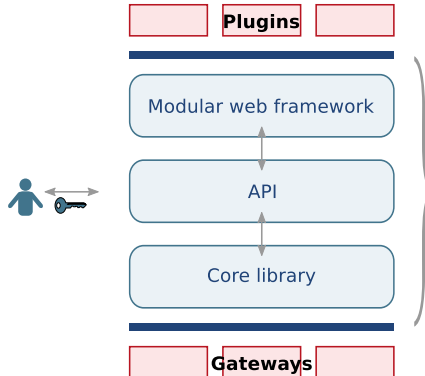
plugins are isolated from the gateways diversity thanks to the **query** abstraction

plugins remain **independent** one from each other thanks to a publish/subscribe communication framework.

*They can for example*

- *publish queries*
- *subscribe to results*

# MySlice architecture

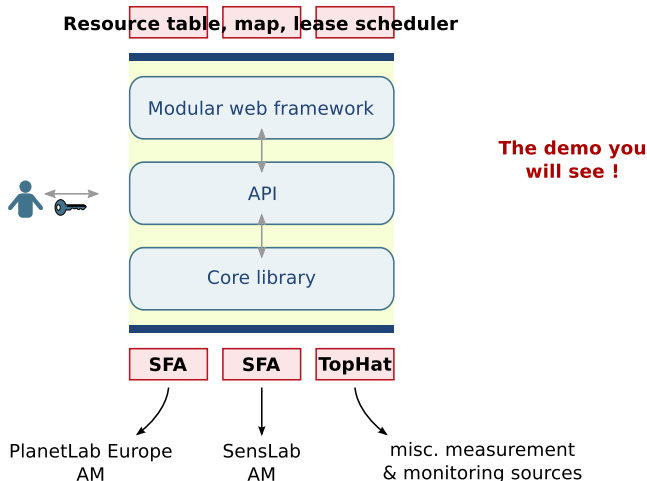


MySlice architecture provides a convenient **aggregation** and **interoperability** layer between the various services and the UI.

It provides plugins with:

- an async. query mechanism
- transparent access to all data and functions
- authentication information
- caching and query optimization (work in progress)

# MySlice architecture



# Pointers

## For users

- Project website: `http://www.myslice.info`
- Demo website: `https://demo.myslice.info`
  - documentation and tutorials

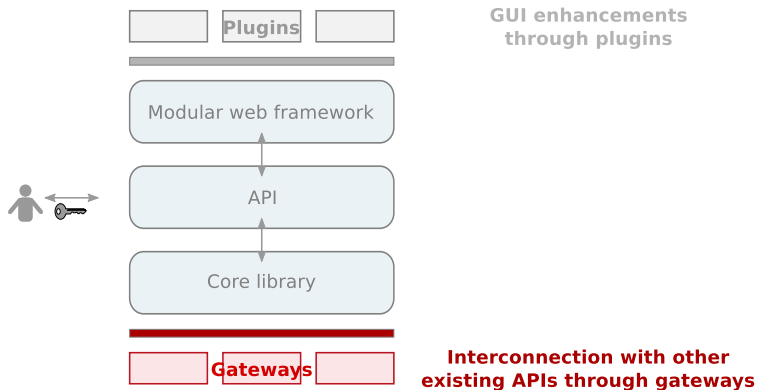
## For testbed owners and developers

- Debian packages
- GIT repository: `http://git.myslice.info`
- TRAC: `https://trac.myslice.info` (new)
- mailing lists, IRC channel, etc.

# Outline

- 1 Overview of MySlice
- 2 Extending MySlice with Gateways**
- 3 Extending MySlice with plugins

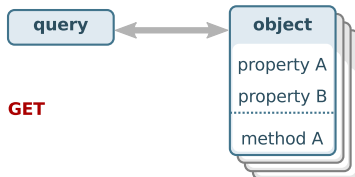
# MySlice architecture



# MySlice objects

identified by a **UUID**:

- to get results
- to edit
- etc.

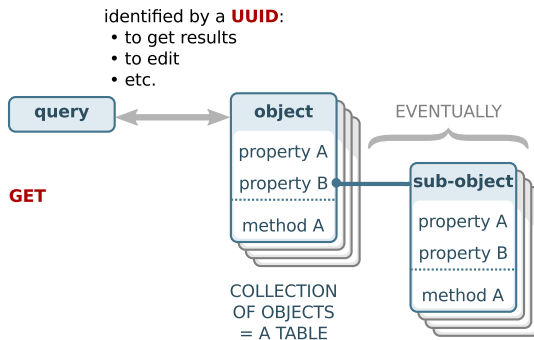


**GET**

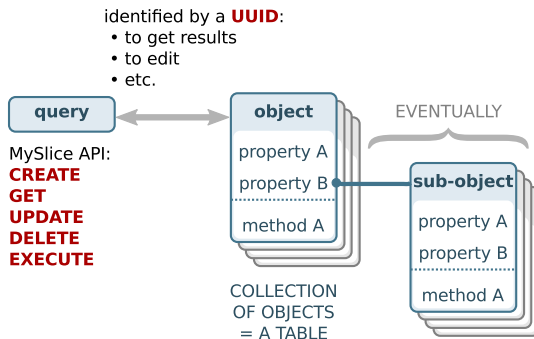
COLLECTION  
OF OBJECTS  
= A TABLE



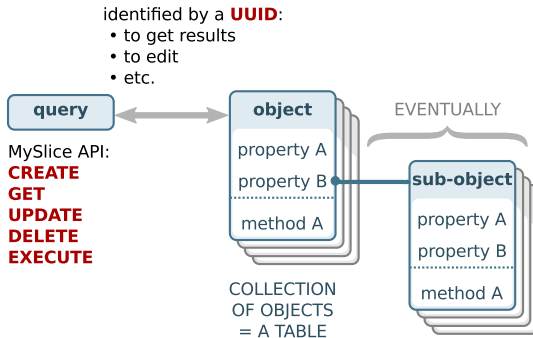
# MySlice objects



# MySlice objects



# MySlice objects



- Extending MySlice = adding new objects, or extending existing ones
- Like simplified distributed, streaming, object oriented database
- MySlice core makes the integration transparent

# Querying the objects with MySlice API

Action(auth, method, filters, params, fields, ts, callback)

Action	method	filters	params	fields	ts	callback
CREATE	✓			✓		!
GET	✓	✓		✓	✓	!
UPDATE	✓	✓	✓	✓		!
DELETE	✓	✓				!
EXECUTE	✓	✓	✓	✓		!

# MySlice/TopHat gateways

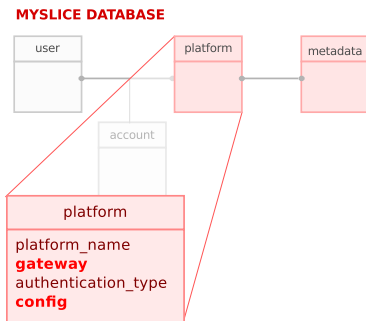
## Existing gateways

SFA, MySlice/TopHat, MaxMind, Team Cumry, SONoMA<sup>†</sup>, ETOMIC<sup>†</sup>

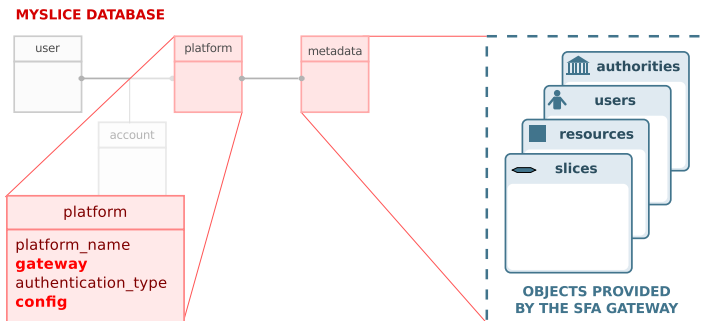
## Developing a new gateway

- ❶ write a gateway module (Python)
  - translate MySlice query into platform query
  - translate back platforms results into MySlice table format
  - handles transport, data formats, semantics
- ❷ write metadata (own format: .h file)
  - ~ enhanced C++ .h file / database schema
  - describes object and its properties / methods
  - and platform capabilities: filtering, column selection, sorting, joining, etc.

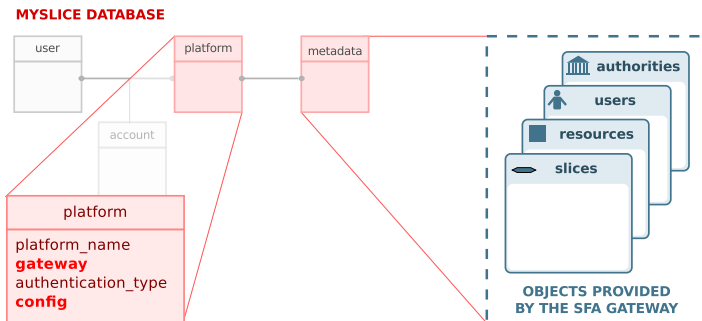
# The case of SFA : metadata



# The case of SFA : metadata



# The case of SFA : metadata

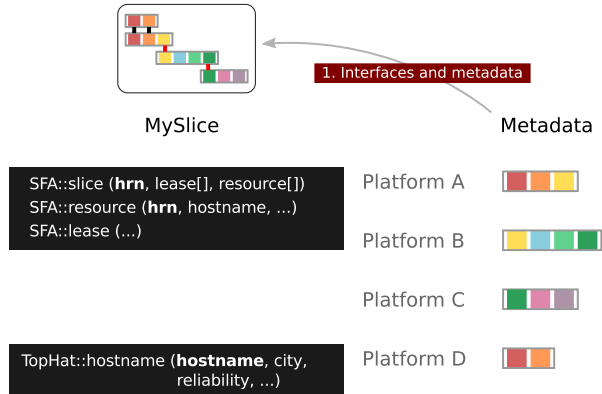


## EXAMPLE OF METADATA FILE

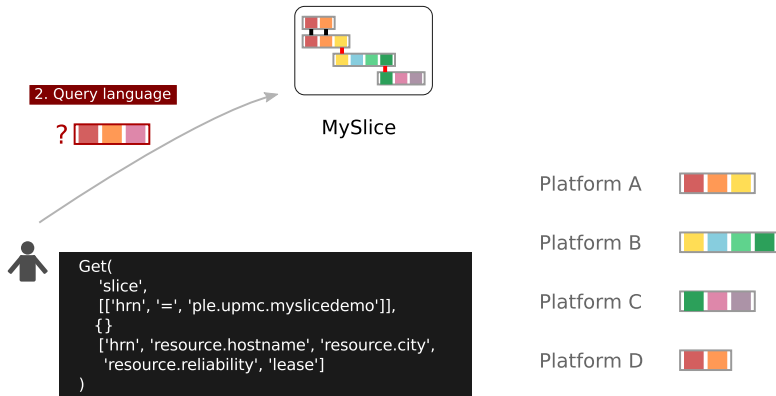
```
class slice {
  const text slice_hrn; /**< Slice Human Readable name */
  resource resource; /**< List of resources associated to the slice */
  lease lease; /**< List of leases associated to the slice */
  user user; /**< List of users associated to the slice */
  KEY(slice_hrn);
};
...
```



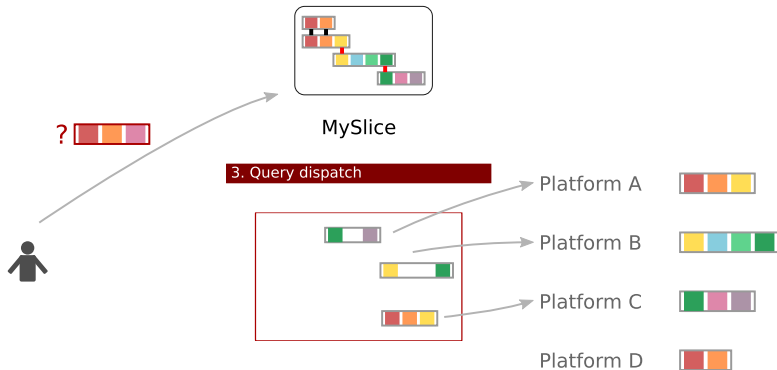
# Interconnection framework



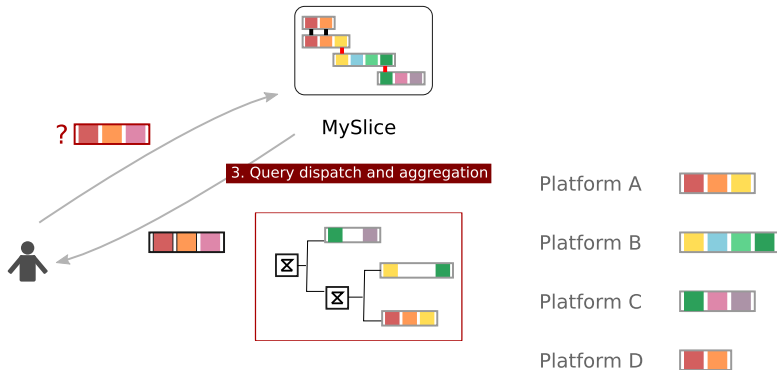
# Interconnection framework



# Interconnection framework



# Interconnection framework



# Aggregating SFA and measurements

## SAMPLE MYSLICE QUERY

```
srv.Get(  
  auth,  
  "slice",  
  [{"slice_hrn", "=", "ple.upmc.myslicedemo"}],  
  {},  
  ["slice_hrn",  
   "resource.network", "resource.type", "resource.hrn",  
   "resource.hostname", "resource.asn", "resource.country"])
```

# Aggregating SFA and measurements

## SAMPLE MYSLICE QUERY

```
srv.Get(  
  auth,  
  "slice",  
  [{"slice_hrn", "=", "ple.upmc.myslicedemo"}],  
  {},  
  [{"slice_hrn",  
    "resource.network", "resource.type", "resource.hrn",  
    "resource.hostname", "resource.asn", "resource.country"}])
```



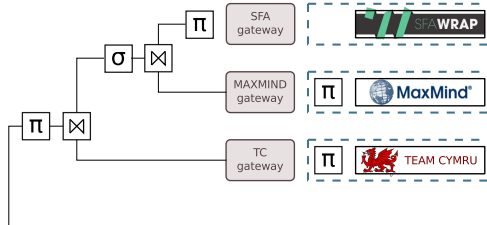
# Aggregating SFA and measurements

## SAMPLE MYSLICE QUERY

```
srv.Get(
  auth,
  "slice",
  [{"slice_hrn", "=", "ple.upmc.myslicedemo"}],
  {},
  [{"slice_hrn",
    "resource.network", "resource.type", "resource.hrn",
    "resource.hostname", "resource.asn", "resource.country"}])
```

### MYSLICE CORE

### GATEWAYS



- $\bowtie$  JOINTURE = JOIN  
join tables
- $\sigma$  SELECTION = WHERE  
filters : filter result objects
- $\pi$  JOINTURE = JOIN  
fields : field selection

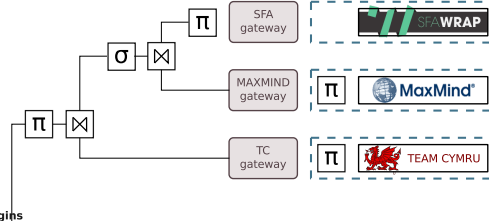
# Aggregating SFA and measurements

## SAMPLE MYSLICE QUERY

```
srv.Get(
  auth,
  "slice",
  [{"slice_hrn", "=", "ple.upmc.myslicedemo"}],
  {},
  [{"slice_hrn",
    "resource.network", "resource.type", "resource.hrn",
    "resource.hostname", "resource.asn", "resource.country"}])
```

## MYSLICE CORE

## GATEWAYS



## MYSLICE WEB GUI + plugins

hrn	hostname	asn	country	+/-
ple.ueople.ueopl1	ueopl1.essex.ac.uk	786	United Kingdom	<input checked="" type="checkbox"/>
ple.upmc.ple2	ple2.ipv6.ilp6.fr	1307	France	<input checked="" type="checkbox"/>
ple.ibbtpl.planck249ple\test\iminds\be	planck249ple.test.iminds.be	2611	Belgium	<input type="checkbox"/>
ple.paderbornple.plane-lab-pb2\uni-paderborn\de	plane-lab-pb2.uni-paderborn.de	680	Germany	<input type="checkbox"/>

$\bowtie$  JOINTURE = JOIN  
join tables

$\sigma$  SELECTION = WHERE  
filters : filter result objects

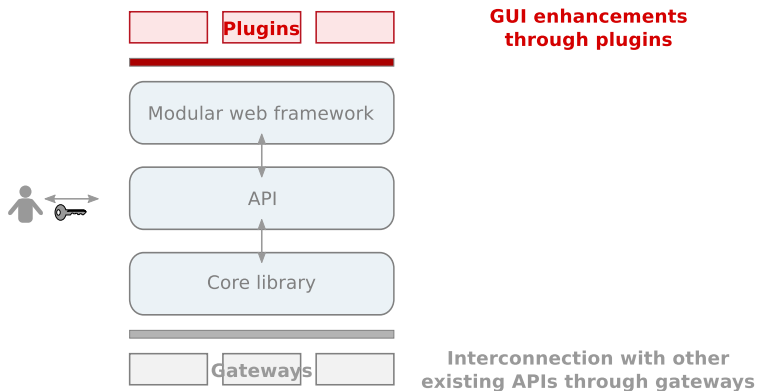
$\pi$  JOINTURE = JOIN  
fields : field selection



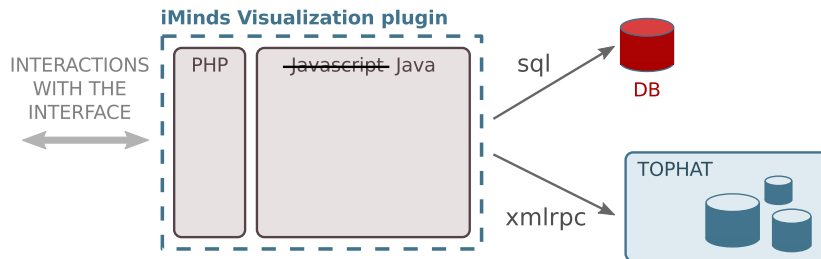
# Outline

- 1 Overview of MySlice
- 2 Extending MySlice with Gateways
- 3 Extending MySlice with plugins**

# MySlice architecture



## Example: iMinds visualization plugin



# Conclusion

- An open solution for users to access the global federation of testbeds
- Support for the complete experimental lifecycle
- Available for download, deployment in progress

## References

- J. Augé, T. Parmentelat, N. Turro, T. Friedman – Tools to foster a global federation of testbeds – Computer Networks – Special issue on Future internet testbeds (in submission)
- L. Baron, J. Augé, T. Friedman, S. Fdida – Towards an integrated portal for networking testbed federation: an open platform approach – FIRE Engineering workshop, Nov 6-7, 2012, Ghent, Belgium
- Jordan Augé, Loïc Barton, Timur Friedman, Serge Fdida – Supporting the experiment lifecycle with MySlice – Invited talk @ GENI Engineering Conference, GEC15 – Oct. 23-25, 2012 – Houston, TX