

# AutoGOLE MEICAN Pilot

## AutoGOLE MEICAN Pilot

### Pilot Proposal

Run a pilot of the RNP's Cipó Service front-end interface – the MEICAN– being used by participant research and education networks (RENs) and exchange point (IXP) operators to configure multidomain point to point circuits. The participants will evaluate the MEICAN as the main interface for AutoGOLE GLIF Project.

### Participants

Research and Education Networks and Exchange Points	Contacts
1. <a href="#">Cenic / Pacific Wave</a>	John Hess
2. <a href="#">NetherLight</a>	Hans Trompert
3. <a href="#">RNP (Cipó)</a>	Alex Moura Marcos Schwarz
4. <a href="#">SouthernLight (SOL)</a>	Alex Moura Marcos Schwarz
5. <a href="#">StarLight (Northwestern University)</a>	Jim Chen
6. <a href="#">SURFnet</a>	Gerben van Malenstein

### Suggestions

As a guideline for experimenting MEICAN we propose that each participant evaluates the list of features below:

- Circuit Management
- Topology Visualization
- Bug reports
- Suggestions

Other interesting features

- Approval Workflow
- Automated Tests
- Monitoring (Beta)

## Documentation

- [MEICAN Quick Start Guide \(PDF\)](#)
- [AutoGOLE Service URLs](#)

## ABOUT AUTOMATED GOLE PROJECT

### The GLIF “Automated GOLE Pilot” Project

The Global Lambda Integrated Facility (GLIF) is a cooperation between NRENs and research institutes with a twofold goal: to create a forum for engineers and researchers to exchange experiences, and to cooperate to make lambdas available for researchers and projects involved in data-intensive research.

In order to make global optical networking possible, GLIF exchange points have been created where long connections can be linked together to create multi-domain lightpaths. These exchange points typically connect users on layers below L2. In order to clarify the policy applied at exchange points, the concept of a GLIF Open Lightpath Exchange (GOLE) has been defined. This mandates that GOLEs treat everyone the same, and be open about any policy that may be applied [[GOLE](#)].

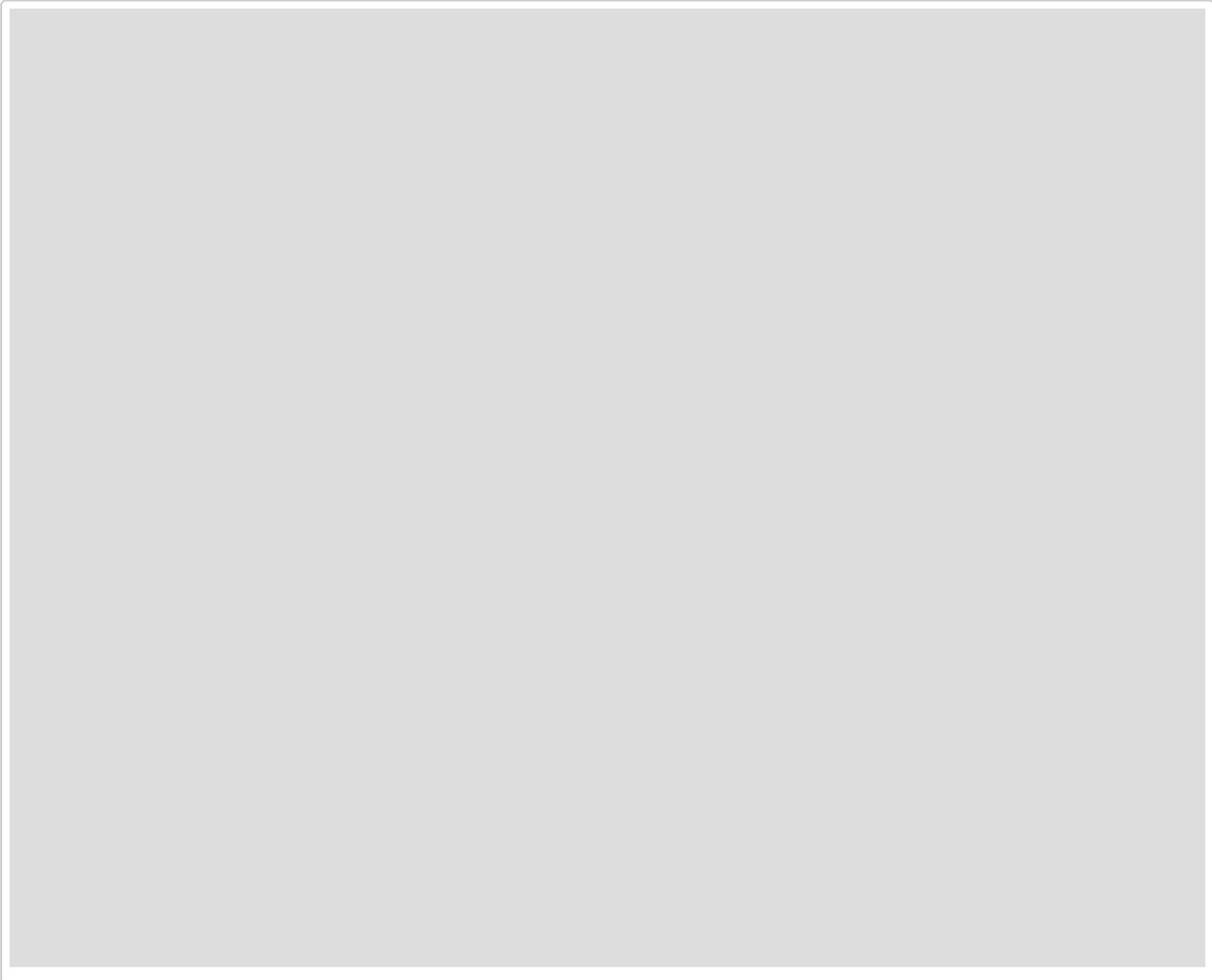
Since the start of GLIF, inter-domain lightpath provisioning has involved much manual processing and actions, since each actor has to play an active role in the process of creating the lightpath, ranging from approving the request to changing the configuration of the network to accommodate the request. Often these are distributed over several time zones, which means that a lightpath request can take about two weeks to be implemented. Based on the Network Services Interface [[NSI](#)], the Automated GOLE testbed created by thirteen GOLEs with different implementations of the NSI Connection Service has made it possible for lightpaths to be created and destroyed within seconds. The early experiments with the Automated GOLE testbed have also helped to improve the NSI Connection Service version 2, which is now submitted as a draft standard.

Ref.: [https://geant3plus.archive.geant.net/Resources/White\\_Papers/Documents/MS101\\_MJ1-2-1\\_Network-Architectures-for-Cloud-Services.pdf](https://geant3plus.archive.geant.net/Resources/White_Papers/Documents/MS101_MJ1-2-1_Network-Architectures-for-Cloud-Services.pdf), Page 17

## Documentation and Contacts

## Pilot Results

- **Pilot Report**
- **WRNP'17 SURFnet Talk about the MEICAN Pilot**



## Documentation

- **[MEICAN User Guide](#)**
- **[Testing Procedure](#)**

## Links

- MEICAN: <http://meican-test.cipo.rnp.br/meican> (in production)  
(ask for user credentials by email to [gre@rnp.br](mailto:gre@rnp.br))
- MEICANPlayground: <http://meican-test.cipo.rnp.br/playground/>  
(User:autogole / Password:autogole)

## AutoGOLE Contacts

- [AutoGOLE mail list](#)

## Basic initial tests

Try to create a circuit reservation and test communication between endpoints using hosts

1. Create a basic circuit reservation
2. Try to create circuit between hosts (from the list of available hosts)
3. Try communication between hosts (ping etc)

## Development Efforts

1) Parser update (priority: **low**) (complexity: **low**)

Review of the URN parsing for NMWG format to consider the URN unified parser and support ESnet intradomain topology format.

Support for extracting geographic coordinates from topology file as example from ESnet.

Support for extracting bandwidth information from topology file as example from ESnet.

2) Reservation (priority: **low**) (complexity: **low**)

Support reservation requests for Zero Bandwidth (low)

Support reservation with Undefined Start time (medium)

Support reservation with Undefined End time (medium)

Support reservation with the Protection tag (RNP will ask ESnet to implement the Protection tag support on the OSCARS NSI Bridge (medium)

3) STP representation review (priority: **medium**) (complexity: **high**)

MEICAN inherited the STP interpretation from OSCARS where the URN of an STP encodes the domain/network, device and port. This logic is not applicable to the NSI URN representation, which is only officially composed of two informations only: network-id and local-id. Consider the definition at [4].

It is necessary a new STP representation internally at MEICAN that doesn't break the NSI specification and is compatible with other domains not based on OSCARS.

An initial proposal is to merge the current representation for device and port to a single entity called local-id.

The representation of this new entity on the map visualization can be done initially as follows:

1. All STPs from a given network that have the same coordinate should be plotted on a single point in the map. And should be labeled with the longest unique prefix match.
2. All STPs from a given network that don't have coordinates should be plotted as a single point in the map. Using the Network coordinate and label.
3. In the reservation form merge the device and port fields into a single local-id or STP field. Defining a more intuitive name is recommended.
4. An additional possibility for this proposal is to provide a custom parser configuration for the domain, to allow.

3.b) STP representation temporary fix (priority: **high**) (complexity: **medium**)

Since 3) is considered too complex, a temporary fix will be to develop a feature to disable the device parsing on a domain (as in OpenNSA domains) and use the local-id part of the URN as port and use the network for device.

4) User management (priority: **medium**) (complexity: **medium/high**)

Better user management scalability, through the ability of a domain administrator to add new users to that given domain, or to add/remove permission to its domain to existing users. Suggestions?

5) NSI Authentication and Authorization compliance (priority: **high**) (complexity: **medium**)

SURFnet implements authentication using OAuth2, but meican currently doesn't provide a way to input the authentication token. The Authentication and Authorization in NSI protocol is documented in the following file: [draft-gwidi-trompert-nsi\\_aa-public-comment-v7.pdf](#)