

**FED4FIRE**

# **Fed4FIRE WP2: info for new testbeds**



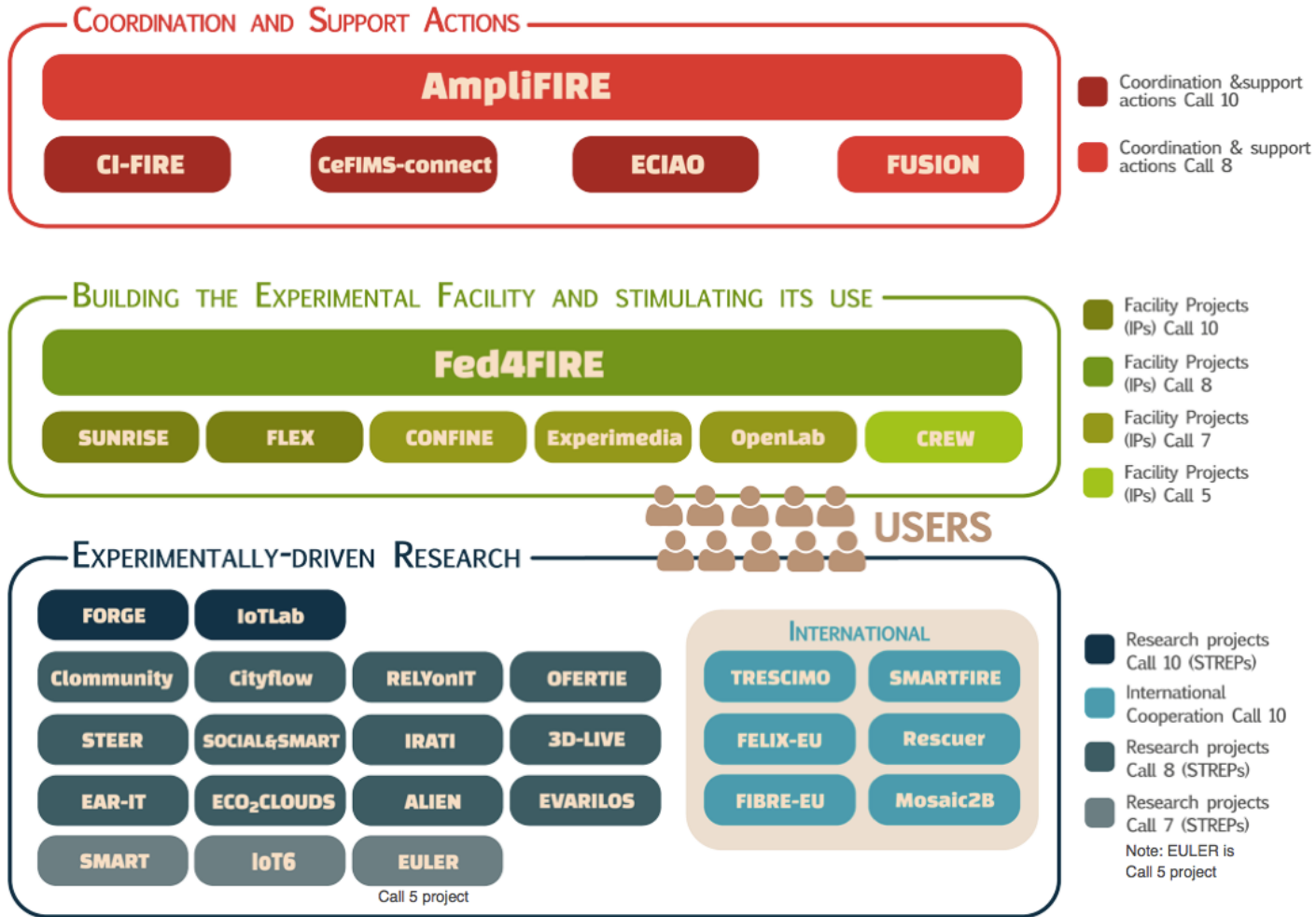
Brecht Vermeulen, Fed4FIRE WP2 lead  
December 17th, 2014

# Fed4FIRE – general info

- IP project coordinated by iMinds
- Total budget: 7.75 MEUR
- 10/2012 - 9/2016
- 28 partners



# Fed4FIRE's role in FIRE

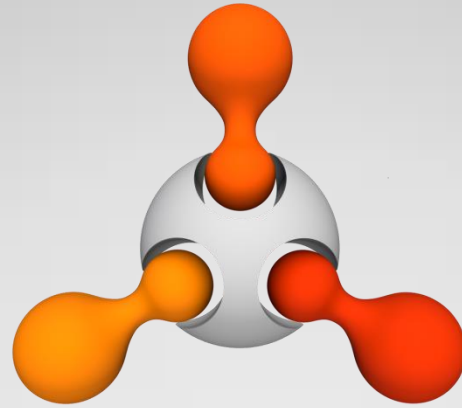


# Current testbeds



# Agenda

- Experiment workflow
  - Overview Fed4FIRE (<http://doc.fed4fire.eu>)
  - Technical workflow between components
  - Monitoring
  - Connectivity
    - Proxy
    - International federation and connectivity
  - Tools beyond resource provisioning
- Federation membership models
- Workflow adding testbed to the federation
  - Documentation and tutorials
  - jFed toolkit for testing and monitoring federation
  - How does the Aggregate Manager API look like
  - How to implement the Aggregate Manager

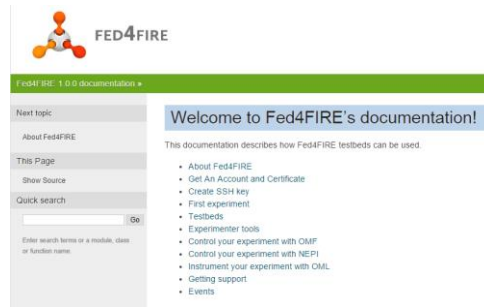


**FED4FIRE**

# Experiment workflow



# Experiment workflow

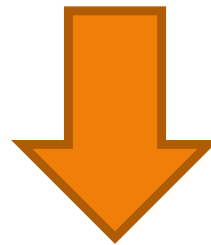


Create Account



Documentation  
<http://doc.fed4fire.eu>

Federation policy:  
experimenter can run  
tutorial experiments  
to learn testbeds



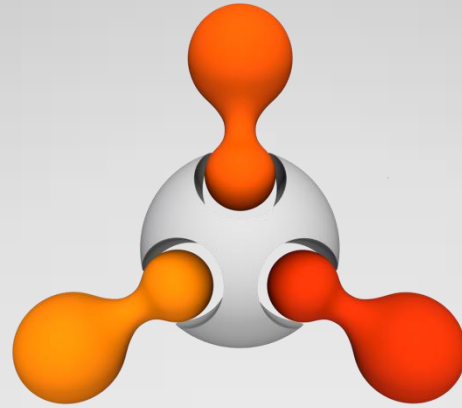
Do more experiments and tutorials:  
Provision resources, control resources  
(ask more quota to testbeds if needed as testbeds  
can have different policies)

# From account creation to first experiment (tutorial with client-server & emulated link)

The screenshot displays the jFed Experimenter Toolkit interface. The main window is titled "jFed Experimenter Toolkit" and features a toolbar with various controls. The toolbar is divided into sections: "Experiment" (Update Status, Renew, Terminate), "Advanced" (Reboot, Edit ssh-keys, Share), "Layout" (Auto Layout), and "Zoom" (Zoom In, Zoom Out, Reset Zoom). Below the toolbar, a network topology is shown on a green grid background. The topology consists of three nodes: "server", "link0", and "client", connected in a line. Below the topology, there are two additional nodes: "VM Bonfire" and "node3". The bottom section of the interface shows a "Progress" tab with a list of tasks, all of which are checked. The tasks include: "Initialize nodes at iMinds Virtual Wall 2", "Waiting for nodes from BonFire to become ready.", "Waiting for nodes from iMinds Virtual Wall 2 to become ready.", "Waiting for nodes from iMinds Virtual Wall 1 to become ready.", "Testing connectivity to nodes from BonFire.", "Testing connectivity to nodes from iMinds Virtual Wall 2.", and "Testing connectivity to nodes from iMinds Virtual Wall 1.". At the bottom left, there is a status bar indicating "This experiment run will expire in 1 hour, 41 minutes and 39 seconds." In the bottom right corner, a terminal window titled "n143-01b.wall1.ilabt.iminds.be - PuTTY" is open, showing the output of a ping command from the server to the client. The output shows 5 successful ping requests, each with a 200 ms response time. The terminal output is as follows:

```
bvermeul@server:~$ ping client
PING client-link0 (192.168.0.2) 56(84) bytes of data:
64 bytes from client-link0 (192.168.0.2): icmp_req=1 ttl=64 time=200 ms
64 bytes from client-link0 (192.168.0.2): icmp_req=2 ttl=64 time=200 ms
64 bytes from client-link0 (192.168.0.2): icmp_req=3 ttl=64 time=200 ms
64 bytes from client-link0 (192.168.0.2): icmp_req=4 ttl=64 time=200 ms
64 bytes from client-link0 (192.168.0.2): icmp_req=5 ttl=64 time=200 ms
^C
--- client-link0 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 200.259/200.271/200.276/0.400 ms
bvermeul@server:~$
```



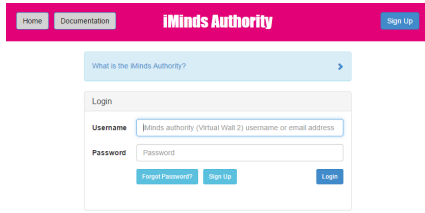


**FED4FIRE**

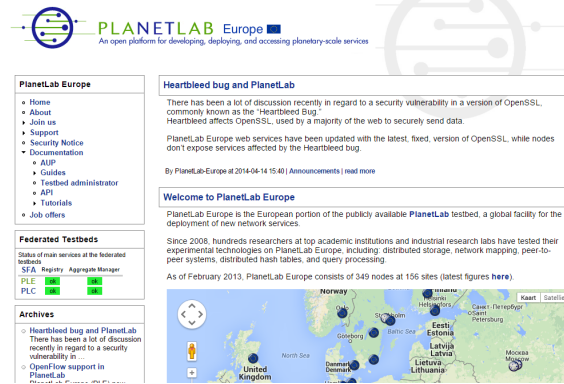
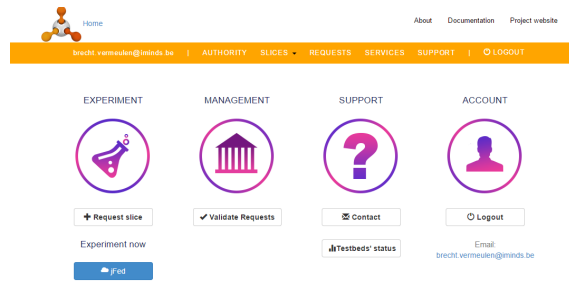
# Technical workflow between components



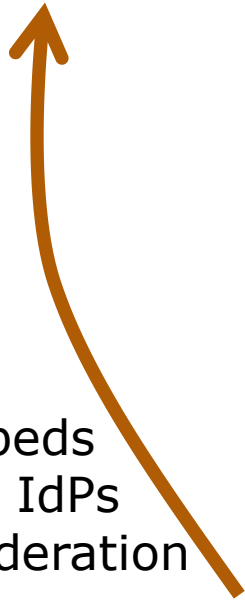
# Workflow



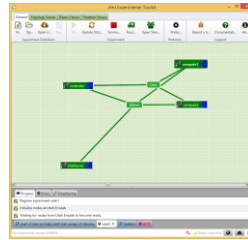
Powered by **emulab** Question or comment? Join the Help Forum Supported by iMinds and the EC © 2014 iMinds Lab



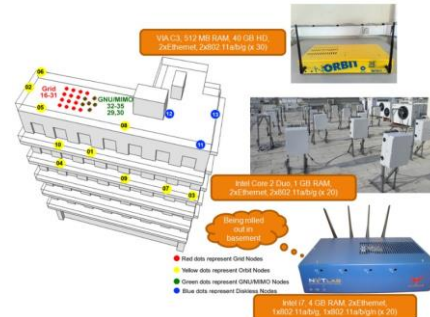
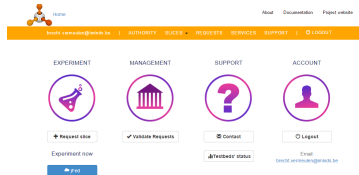
Signed X.509 certificate of **an** identity provider



Testbeds trust IdPs in federation



Use **a** tool



Use **one or more** testbeds

# Fed4FIRE Identity providers

Home Documentation **iMinds Authority** Sign Up

What is the iMinds Authority? >

Login

Username

Password

Forgot Password? Sign Up Login

Home brecht.vermeulen@iminds.be | AUTHORITY SLICES | REQUESTS SERVICES SUPPORT | LOGOUT

EXPERIMENT MANAGEMENT SUPPORT ACCOUNT

+ Request slice ✓ Validate Requests ✉ Contact ⌂ Logout

Experiment now jFed

Email: brecht.vermeulen@iminds.be

Powered by emulab Question or comment? Join the Help Forum Supported by iMinds and the EC © 2014 iMinds iLab.t

<https://authority.ilabt.iminds.be>

Portal: <https://portal.fed4fire.eu>



**PlanetLab Europe**

- Home
- About
- Join us
- Support
- Security Notice
- Documentation
  - AUP
  - Guides
  - Testbed administrator
  - API
  - Tutorials
- Job offers

**Federated Testbeds**

Status of main services at the federated testbeds

SFA	Registry	Aggregate Manager
PLE	OK	OK
PLC	OK	OK

**Archives**

- Heartbleed bug and PlanetLab
- OpenFlow support in PlanetLab

**Heartbleed bug and PlanetLab**

There has been a lot of discussion recently in regard to a security vulnerability in a version of OpenSSL, commonly known as the "Heartbleed Bug." Heartbleed affects OpenSSL, used by a majority of the web to securely send data.

PlanetLab Europe web services have been updated with the latest, fixed, version of OpenSSL, while nodes don't expose services affected by the Heartbleed bug.

By PlanetLab-Europe at 2014-04-14 15:40 | Announcements | read more

**Welcome to PlanetLab Europe**

PlanetLab Europe is the European portion of the publicly available PlanetLab testbed, a global facility for the deployment of new network services.

Since 2008, hundreds researchers at top academic institutions and industrial research labs have tested their experimental technologies on PlanetLab Europe, including, distributed storage, network mapping, peer-to-peer systems, distributed hash tables, and query processing.

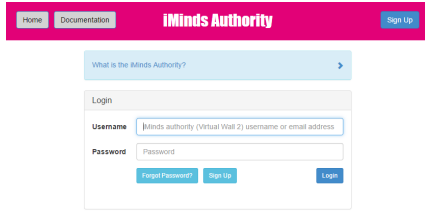
As of February 2013, PlanetLab Europe consists of 349 nodes at 156 sites (latest figures [here](#)).



Planetlab Europe <http://www.planet-lab.eu/>



# Workflow (protocol: XMLRPC over SSL)



Slice Authority API

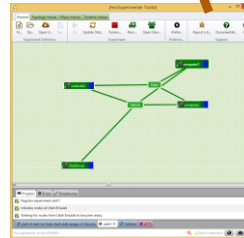
Member Authority API

3. Create slice/get credential (signed XML)

2. Get credential (signed XML)

Create Account and get certificate

1. Use signed certificate in tool

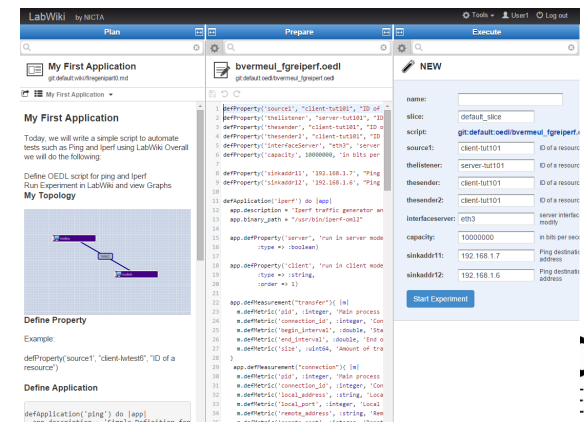


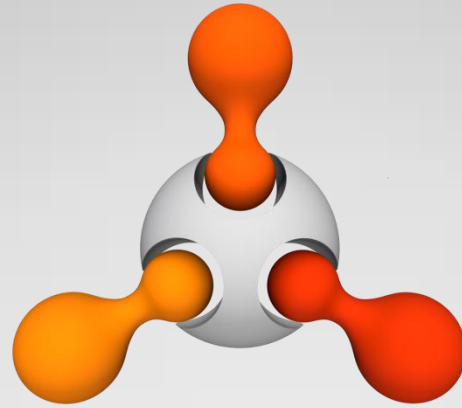
4. Provision resources

Aggregate Manager API



5. Control resources



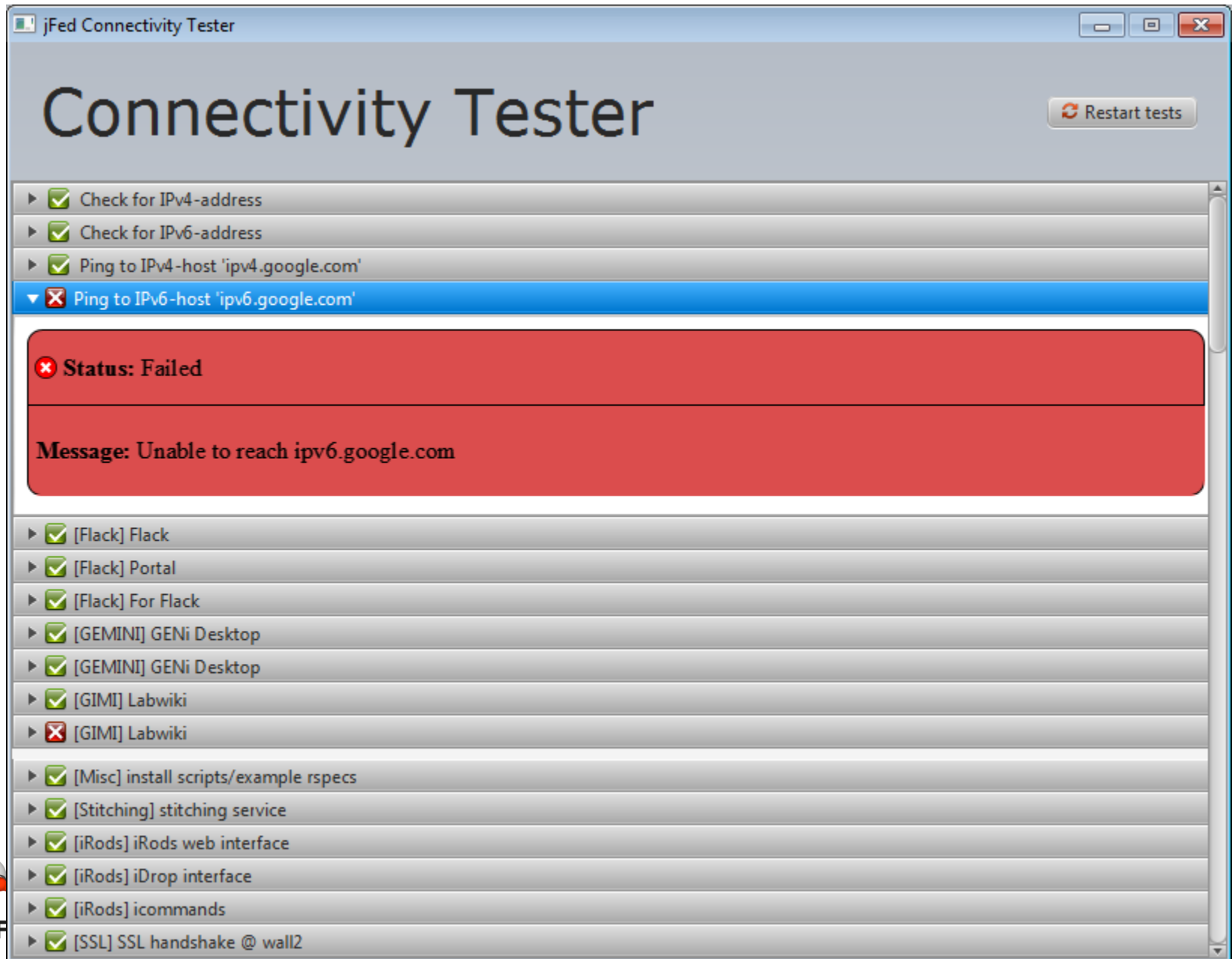


**FED4FIRE**

**Connectivity: proxy**



# Connectivity test (also in bug report)



The screenshot shows a window titled "jFed Connectivity Tester" with a "Restart tests" button in the top right. The main heading is "Connectivity Tester". Below this is a list of test items, each with a status icon (checkmark or X) and a right-pointing arrow:

- ▶  Check for IPv4-address
- ▶  Check for IPv6-address
- ▶  Ping to IPv4-host 'ipv4.google.com'
- ▶  Ping to IPv6-host 'ipv6.google.com'
- ▶  [Flack] Flack
- ▶  [Flack] Portal
- ▶  [Flack] For Flack
- ▶  [GEMINI] GENi Desktop
- ▶  [GEMINI] GENi Desktop
- ▶  [GIMI] Labwiki
- ▶  [GIMI] Labwiki
- ▶  [Misc] install scripts/example rspecs
- ▶  [Stitching] stitching service
- ▶  [iRods] iRods web interface
- ▶  [iRods] iDrop interface
- ▶  [iRods] icommands
- ▶  [SSL] SSL handshake @ wall2

The fourth item, "Ping to IPv6-host 'ipv6.google.com'", is highlighted in blue and shows a red X icon. Below it is a red box with the following text:

**✘ Status: Failed**

**Message: Unable to reach ipv6.google.com**

# TCP ports and firewalls

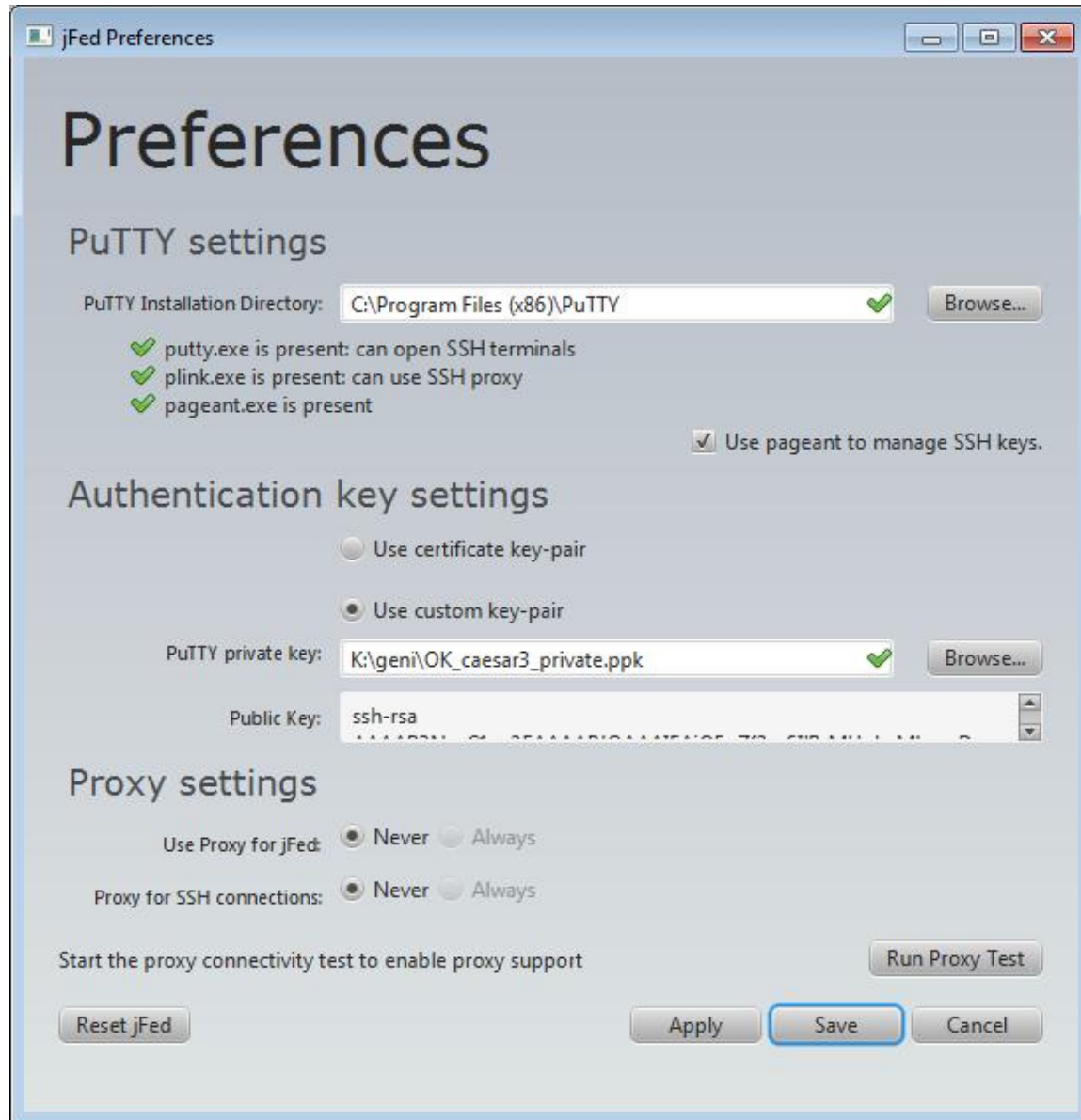
- 12369, 12346, 11443, 8081, ...
- IPv6 for node access

= problems

- First step: Detection (connectivity tester)
- Second step: work around -> SSH proxy

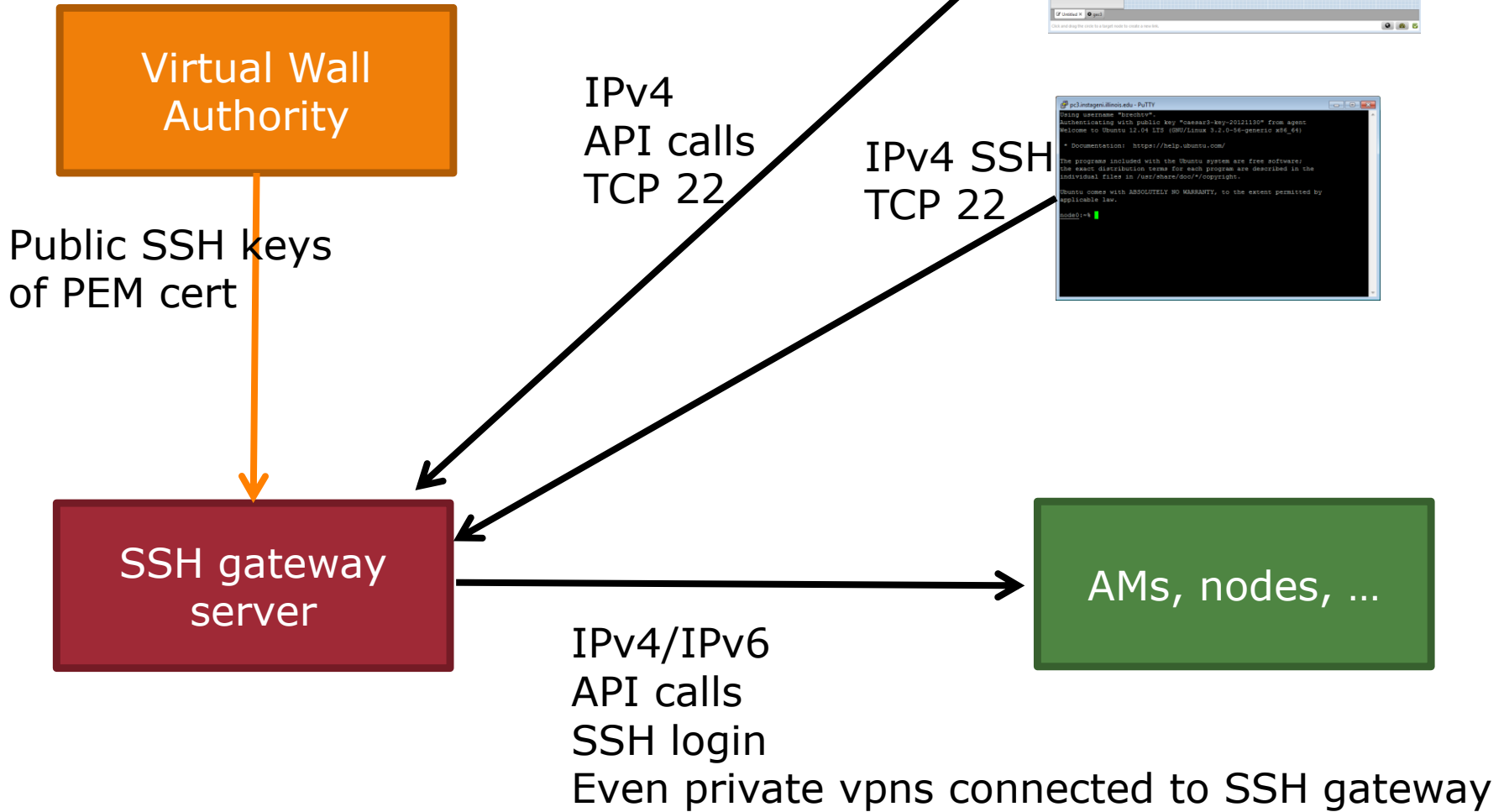
# SSH proxy (optional !)

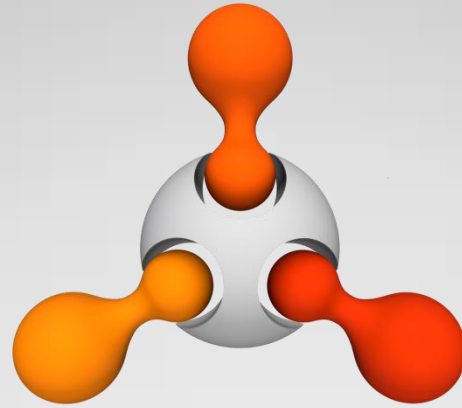
- For API calls
- For SSH login
- Automatic SSH agent for extra comfort





# SSH proxy: only TCP 22



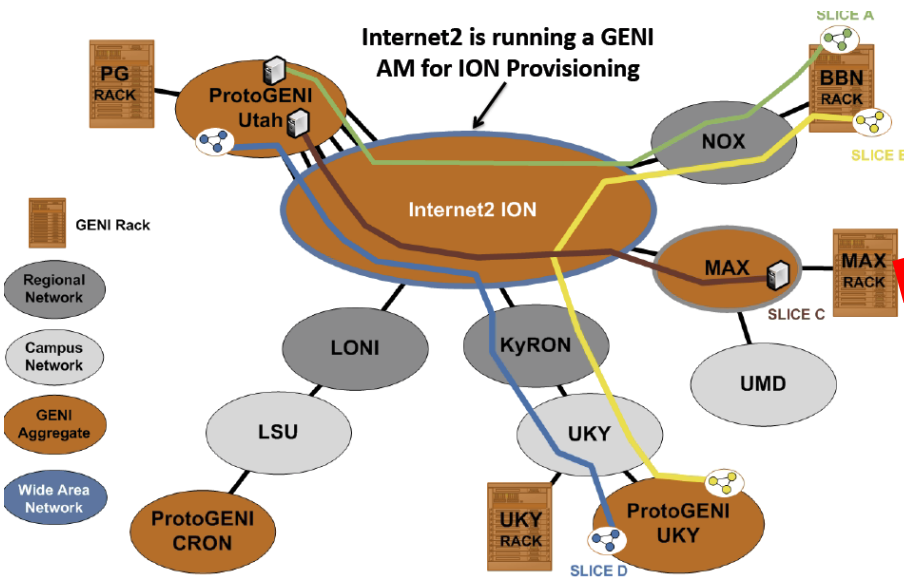


**FED4FIRE**

# International federation and connectivity



# Layer 2 connectivity = stitching VLANs



Meshed L2 connections possible

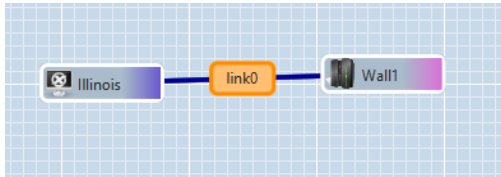
SDX = software defined exchange

eases connectivity (=exchange)

VLAN translation needed + SDN functionality

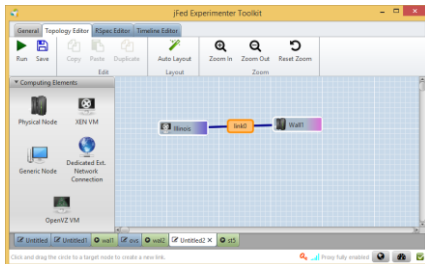
# Stitching workflow: iMinds to Illinois

1. Experimenter draws layer 2 link



Stitching  
Computation  
Service (SCS)

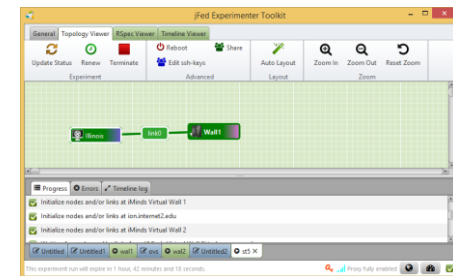
2. and starts provisioning



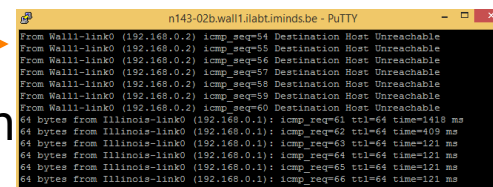
3. Tool contacts SCS to know the path and SCS sends back the path and workflow (e.g. some networks can do VLAN translation)



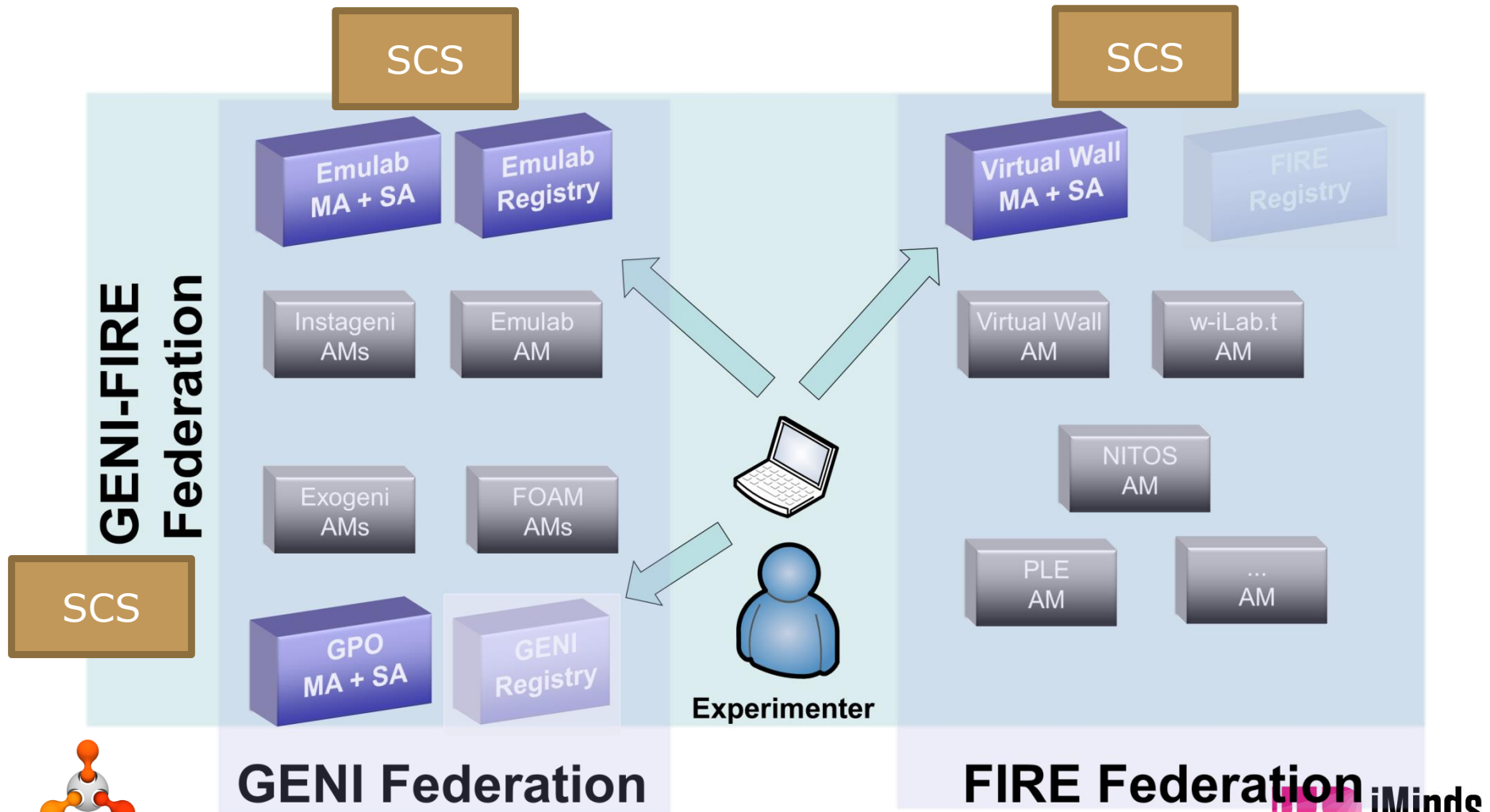
4. Tool provisions at all testbeds and networks through the aggregate manager API and negotiates the VLAN IDs

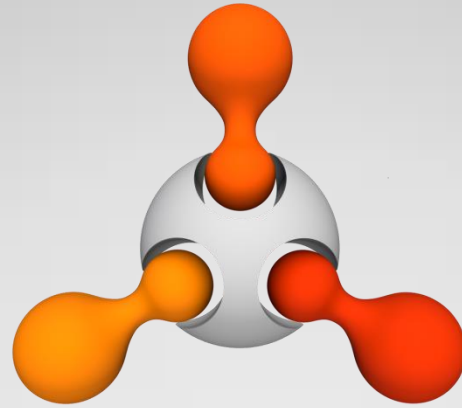


5. User logs in and can ping



# SCS per federation





**FED4FIRE**

**Tools beyond provisioning:  
experiment control**



# jFed: timebased experiment control

The screenshot displays the jFed Experimenter Toolkit interface. The window title is "jFed Experimenter Toolkit". The interface is divided into several sections:

- General Viewers:** Includes tabs for "General", "Topology Viewer", "RSpec Viewer", and "Timeline Viewer".
- Control Panel:** Contains buttons for "Update Status", "Renew", "Terminate", "Start", "Pause", "Stop", "Time", "Instant", "Save results", "Add Command", "Add barrier", "Zoom In", "Zoom Out", and "Reset Zoom".
- Timeline View:** Shows a timeline from 0:00 to 00:05:00. The left side shows a list of nodes: "server", "client", "Bonfire", and "node3". The right side shows a detailed view of the timeline with horizontal bars representing the execution of commands on different nodes. A red bar is visible on the "client" node around 00:03:00.
- Terminal:** Displays the output of a command. The command is: `>_ client@urn:publicid:IDN+wall1.ilabt.iminds.be+authority+cm: Command 2`. The output shows: 

```
09:32:27: Your command has started.
09:32:27: -----
09:32:27: Client connecting to server, TCP port 5001
09:32:27: TCP window size: 23.5 KByte (default)
09:32:27: -----
09:32:27: --
09:32:27: [ 3] local 192.168.0.2 port 48699 connected with 192.168.0.1 port 5001
09:32:37: [ID] Interval Transfer Bandwidth
09:32:37: [ 3] 0.0-10.0 sec 1.10 GBytes 944 Mbits/sec
09:32:37: Your command has finished.
```
- Taskbar:** Shows several open windows: "Untitled", "Untitled1", "wal1", "ovs", and "wal2".
- Status Bar:** At the bottom, it indicates "This experiment run will expire in 1 hour, 54 minutes and 14 seconds." and "Proxy fully enabled".

# Labwiki experiment control (OMF/OML)

LabWiki by NICTA

Tools User1 Log out

Plan Prepare Execute

My First Application  
git:default:wiki/firegenipart0.md

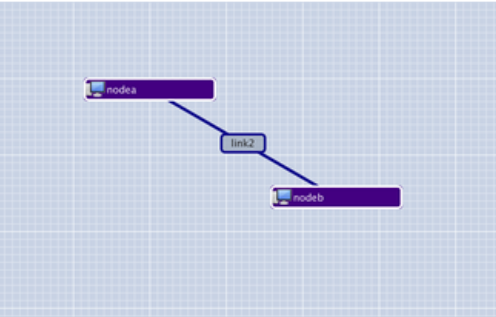
My First Application

### My First Application

Today, we will write a simple script to automate tests such as Ping and Iperf using LabWiki Overall we will do the following:

Define OEDL script for ping and Iperf  
Run Experiment in LabWiki and view Graphs

### My Topology



Define Property

Example:

```
defProperty('source1', "client-lwtest", "ID of a resource")
```

Define Application

```
defApplication('ping') do [app]
  app.description = 'Simple Definition for
```

bvermeul\_fgriperf.oedl  
git:default:oedl/bvermeul\_fgriperf.oedl

```
1 defProperty('source1', "client-tut101", "ID of
2 defProperty('thelister', "server-tut101", "ID
3 defProperty('thesender', "client-tut101", "ID o
4 defProperty('thesender2', "client-tut101", "ID
5 defProperty('interfaceServer', "eth3", 'server
6 defProperty('capacity', 10000000, 'in bits per
7
8 defProperty('sinkaddr11', '192.168.1.7', "Ping
9 defProperty('sinkaddr12', '192.168.1.6', "Ping
10
11 defApplication('iperf') do [app]
12   app.description = 'Iperf traffic generator an
13   app.binary_path = "/usr/bin/iperf-om12"
14
15   app.defProperty('server', 'run in server mode
16     :type => :boolean)
17
18   app.defProperty('client', 'run in client mode
19     :type => :string,
20     :order => 1)
21
22   app.defMeasurement("transfer"){ |m|
23     m.defMetric('pid', :integer, 'Main process
24     m.defMetric('connection_id', :integer, 'Con
25     m.defMetric('begin_interval', :double, 'Sta
26
27
28
29   app.defMeasurement("connection"){ |m|
30     m.defMetric('pid', :integer, 'Main process
31     m.defMetric('connection_id', :integer, 'Con
32     m.defMetric('local_address', :string, 'Loca
33     m.defMetric('local_port', :integer, 'Local
34     m.defMetric('remote_address', :string, 'Rem
35     m.defMetric('remote port', :integer, 'Remot
```

NEW

name:

slice:

script: [git:default:oedl/bvermeul\\_fgriperf.oedl](git:default:oedl/bvermeul_fgriperf.oedl)

source1:  ID of a resource

thelister:  ID of a resource

thesender:  ID of a resource

thesender2:  ID of a resource

interfaceserver:  server interface modify

capacity:  in bits per second

sinkaddr11:  Ping destination address

sinkaddr12:  Ping destination address

Start Experiment

<http://labwiki.test.atlantis.ugent.be:4000>

Minds  
VATE.CREATE

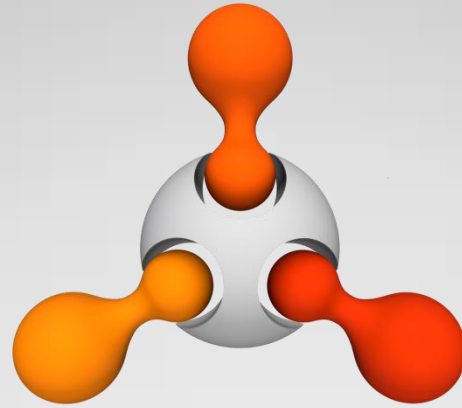


# NEPI experiment control

- <http://doc.fed4fire.eu/nepi.html>

```
n096-09b.wall2.ilabt.iminds.be - PuTTY
ec:~# pico -w ping.py
ec:~# python ping.py
2014-02-25 19:23:39,989 LinuxNode INFO   guid 1 - host n095-26.wall2.ilabt.iminds.be - Deploying node
2014-02-25 19:23:42,070 LinuxNode INFO   guid 1 - host n095-26.wall2.ilabt.iminds.be - Cleaning up processes
2014-02-25 19:23:42,085 LinuxNode INFO   guid 1 - host n095-26.wall2.ilabt.iminds.be - Cleaning up home
2014-02-25 19:23:42,991 LinuxApplication INFO  guid 2 - host n095-26.wall2.ilabt.iminds.be - Deploying command 'ping -c3 node2'
2014-02-25 19:23:43,007 LinuxApplication INFO  guid 2 - host n095-26.wall2.ilabt.iminds.be - Uploading command 'ping -c3 node2'
2014-02-25 19:23:44,246 LinuxApplication INFO  guid 2 - host n095-26.wall2.ilabt.iminds.be - Provisioning finished
2014-02-25 19:23:44,992 LinuxApplication INFO  guid 2 - host n095-26.wall2.ilabt.iminds.be - Starting command 'ping -c3 node2'
2014-02-25 19:23:47,152 LinuxApplication INFO  guid 2 - host n095-26.wall2.ilabt.iminds.be - Retrieving 'stdout' trace all
PING node2-link3 (10.10.1.1) 56(84) bytes of data:
64 bytes from node2-link3 (10.10.1.1): icmp_req=1 ttl=64 time=0.574 ms
64 bytes from node2-link3 (10.10.1.1): icmp_req=2 ttl=64 time=0.207 ms
64 bytes from node2-link3 (10.10.1.1): icmp_req=3 ttl=64 time=0.214 ms

--- node2-link3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.207/0.331/0.574/0.172 ms
2014-02-25 19:23:47,186 LinuxApplication INFO  guid 2 - host n095-26.wall2.ilabt.iminds.be - Releasing resource
ec:~#
```



**FED4FIRE**

# Federation membership

technical requirements



# Note

- This is about technical requirements
- There is also need for policy decisions (can a testbed join or not) – to be discussed in the sustainability task/federator (board) work
  - Although a testbed joining the federation is different from their users joining (their authority is not automatically allowed on existing testbeds, only vice versa: F4F experimenters can use the new testbed)

# What is a testbed (that can be federated)?

- Testbed = hardware + management software
- ‘Ssh/FRCP controlled resource’ testbeds
  - Ability to share resources between different users
    - Shared over time or in parallel (multiplexing, slicing)
    - Concept of credentials and dedicated access (e.g. ssh)
- ‘API only’ testbeds
  - A service with an API (proprietary or standard)
  - Concept of credentials
- \*\*\* better naming for these types needed, but the idea should be clear (infrastructure versus service is confusing)

# What types of federation

- Light federation
- Tight federation
- Associated testbeds

# Tight federation: min. requirements

- Support for AMv2 or AMv3 (or later versions)
  - Authentication, authorization: X.509 certificates, slice and user credentials, accepting root certificates of the main F4F authorities
  - Resource description and discovery: RSpec definition
  - Provisioning (instant): through the AM API
  - Control: through SSH with ssh public/private keys put in the API calls, FRCP control or openflow: point a controller for a switch
- Documentation (on a webpage maintained by the testbed)
  - Testbed description
  - RSpec description
  - URLs of the AM API
  - A basic experiment showing the testbed (and with a F4F tool), described as a tutorial
- Policies: everyone with a valid F4F certificate can execute the basic experiment without extra approval
- Facility monitoring
  - AM API tested from central location, if testbed has internal monitoring, send a summary through OML to the central OML server
- Connectivity: public IPv4 for AM, public IPv4 or IPv6 for ssh login (exceptions for VPN can be granted, but then the ssh gateway of the F4F federation will be a permanent client of the VPN)
- Testbed has to provide basic support on the testbed functionalities towards experimenters

# Tight federation: options

- Infrastructure monitoring
- Advanced reservation
- SLA
- Reputation
- Permanent storage
- Experiment control
  - FRCP enabled images
  - AMQP server
  - PDP
- Layer 2 connectivity between testbeds
  - VLAN stitching (federation runs stitching computation engine)
  - Tunnels (egre or gre option in RSpec link)

# Tight federation: what does the federation offer ?

- Testing tools for the AM API, test credentials, ...
- Nightly testing when federated
- Central monitor dashboard
- Min. 1 client tool having support for all federated infrastructure testbeds
- At least 1 authority to provide credentials
- Ssh gateway (to bridge e.g. to IPv6, VPNs, ...)
- Central documentation linking to all testbeds
- Central support (google group, NOC) for first help and single point of contact



# Light integration: min. requirement

- Support for Fed4FIRE credentials in client based SSL API
  - X.509 certificates, e.g. derived PKCS12 version which can be loaded in a webbrowser or other HTTPS tool
  - API is not the AM API
- Documentation (on a webpage maintained by the testbed)
  - Testbed description
  - Documentation on the specific API
  - URLs of the API
  - A basic experiment showing the testbed, in a tutorial format
- Policies: everyone with a valid F4F certificate can execute the basic experiment without extra approval
- Facility monitoring
  - API tested from central location, if testbed has internal monitoring, send a summary through OML to the central OML server
- Connectivity: public IPv4 for the API server

# Light federation: what does the federation offer ?

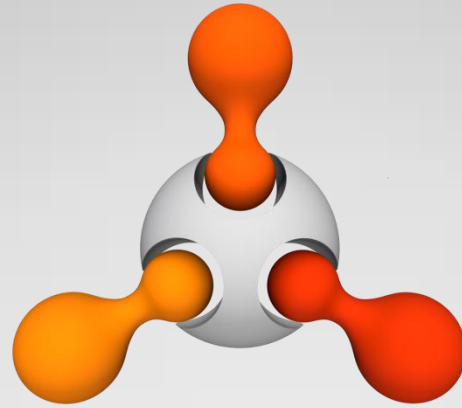
- Test credentials
- Information on enabling PKCS12 authentication
- Central monitor dashboard
- Min. 1 client tool exporting PKCS12 credentials from the X.509 certificate
- At least 1 authority to provide credentials
- Central documentation linking to all testbeds
- Central support (google group, NOC) for first help and single point of contact

# Associated testbeds

- No real federation (e.g. no credential exchange, no testing, ...)
- Only mentioning the testbed and linking to the testbed specific documentation
- Testbed has to organise its own support

# Matrix of possibilities

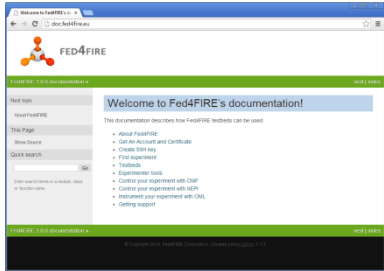
- ‘SSH/FRCP/openflow controllable testbeds’
  - Light federation (e.g. use Bonfire API with F4F credentials)
  - Tight federation (e.g. Bonfire with an AM, use F4F tool)
  - Associated tested
- ‘API only testbed’
  - Only Light federation possible (e.g. hadoop on demand service with F4F credentials)
  - Associated testbed
- Reason to make this clear: an ‘API only testbed’ can never do Tight federation, so it is not federated ‘less’, just at the moment, this is the maximum federation that is possible. (and that is demanded from experimenter view as far as we see)



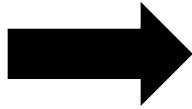
**FED4FIRE**

**Workflow adding a testbed  
to the federation**

# Adding a testbed to the federation



doc.fed4fire.eu



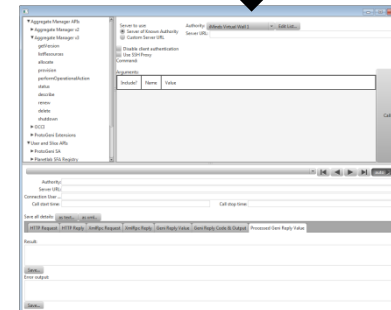
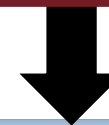
AM API doc



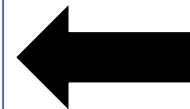
Design RSpecs



Implement AM API on top of testbed



Test with jFed probe



Testbed Name	Ping	latency (ms)	GetVersion	Status	Free Resources	Internal testbed monitoring status	Last check	Internal status
doc4fire	OK	100	OK	OK			2014-04-10 10:00:00	OK
CC-EP	OK	100	OK	OK			2014-04-10 10:00:00	OK
FUTURECO	OK	100	OK	OK			2014-04-10 10:00:00	OK
ITN	OK	100	OK	OK			2014-04-10 10:00:00	OK
NETWORKE	OK	100	OK	OK			2014-04-10 10:00:00	OK
NETOS-BRIDGE	OK	100	OK	OK			2014-04-10 10:00:00	OK
NETOS-SPANNING	OK	100	OK	OK			2014-04-10 10:00:00	OK
Openlab (Bristol openlab)	OK	100	OK	OK			2014-04-10 10:00:00	OK
Openlab (Bristol openlab)	OK	100	OK	OK			2014-04-10 10:00:00	OK
Openlab (CCAT openlab)	OK	100	OK	OK			2014-04-10 10:00:00	OK
Openlab (CCAT openlab)	OK	100	OK	OK			2014-04-10 10:00:00	OK
Planetsat Energy	OK	100	OK	OK			2014-04-10 10:00:00	OK
Planetsat Energy	OK	100	OK	OK			2014-04-10 10:00:00	OK
Virtual Host 1	OK	100	OK	OK			2014-04-10 10:00:00	OK
Virtual Host 2	OK	100	OK	OK			2014-04-10 10:00:00	OK
Virtual Host 2	OK	100	OK	OK			2014-04-10 10:00:00	OK
Virtual Host 2	OK	100	OK	OK			2014-04-10 10:00:00	OK

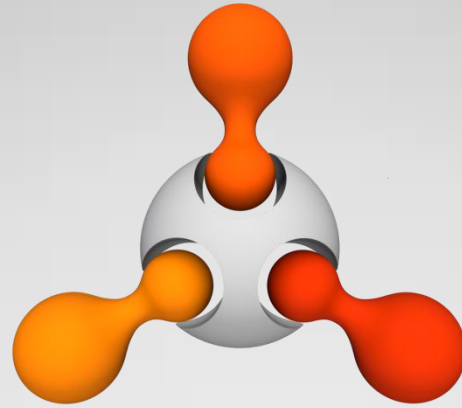
Dashboard and nightly testing (+internal testbed monitoring)



Document testbed



Add testbed in Experimenter tools



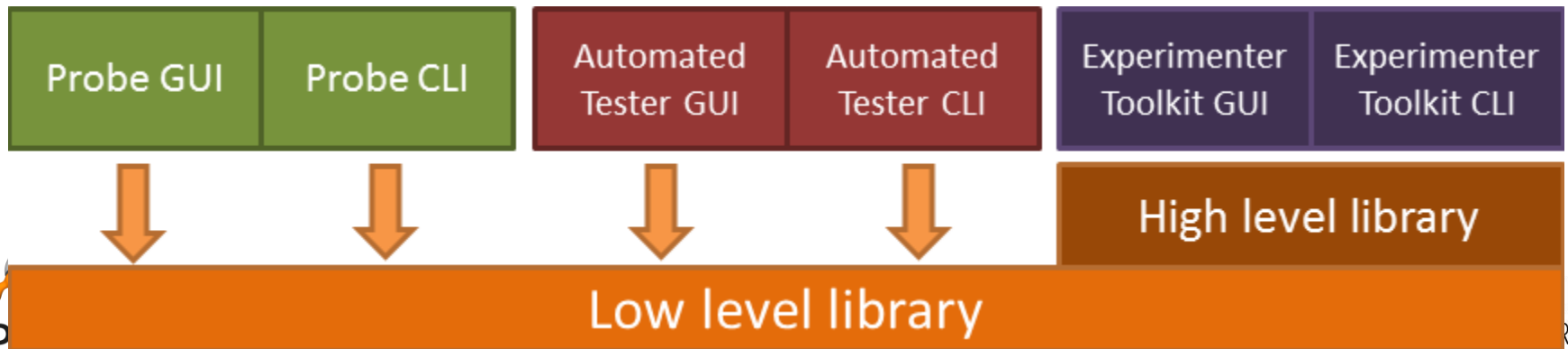
**FED4FIRE**

**jFed toolkit for federation  
testing and monitoring**



# jFed toolkit

- <http://jfed.iminds.be>, current release 5.4.0
- Speaks AM API, Federation (CH) APIs, SCS, ...
- Written in Java(FX)
- MIT license
- Experimenter tool, test and monitor federation





# Philosophy: jFed experimenter GUI

- Leverages APIs from jFed Probe testing
- Can be used by a new experimenter (abstract things !)
- Full power when needed
  - raw Rspec
  - API call insight
- Debug and support
  - Leverage API call analysis from jFed probe
  - For support: send all those calls to support !
- Can work around firewall port blocking stuff through SSH proxy
- Cross platform: Windows, OS X, Linux
- Saves and reads RSpecs

# Abstract resources

The screenshot displays the jFed Experimenter Toolkit interface. The main window is titled "jFed Experimenter Toolkit" and features a menu bar with "General", "Topology Editor", "RSpec Editor", and "Timeline Editor". Below the menu bar is a toolbar with icons for "Run", "Save", "Copy", "Paste", "Duplicate", "Auto Layout", "Zoom In", "Zoom Out", and "Reset Zoom".

The main workspace shows a network diagram on a blue grid background. The diagram consists of three nodes: "server", "link0", and "client", connected by a blue line. A "node3" node is highlighted with a green border. Other nodes visible in the workspace include "Virtual Machine", "Wireless Node", "Physical Node", "XEN VM", "Generic Node", "Dedicated Ext. Network Connection", and "OpenVZ VM".

A "Properties of node3" dialog box is open, showing the "Xen Options" tab. The dialog box contains the following information:

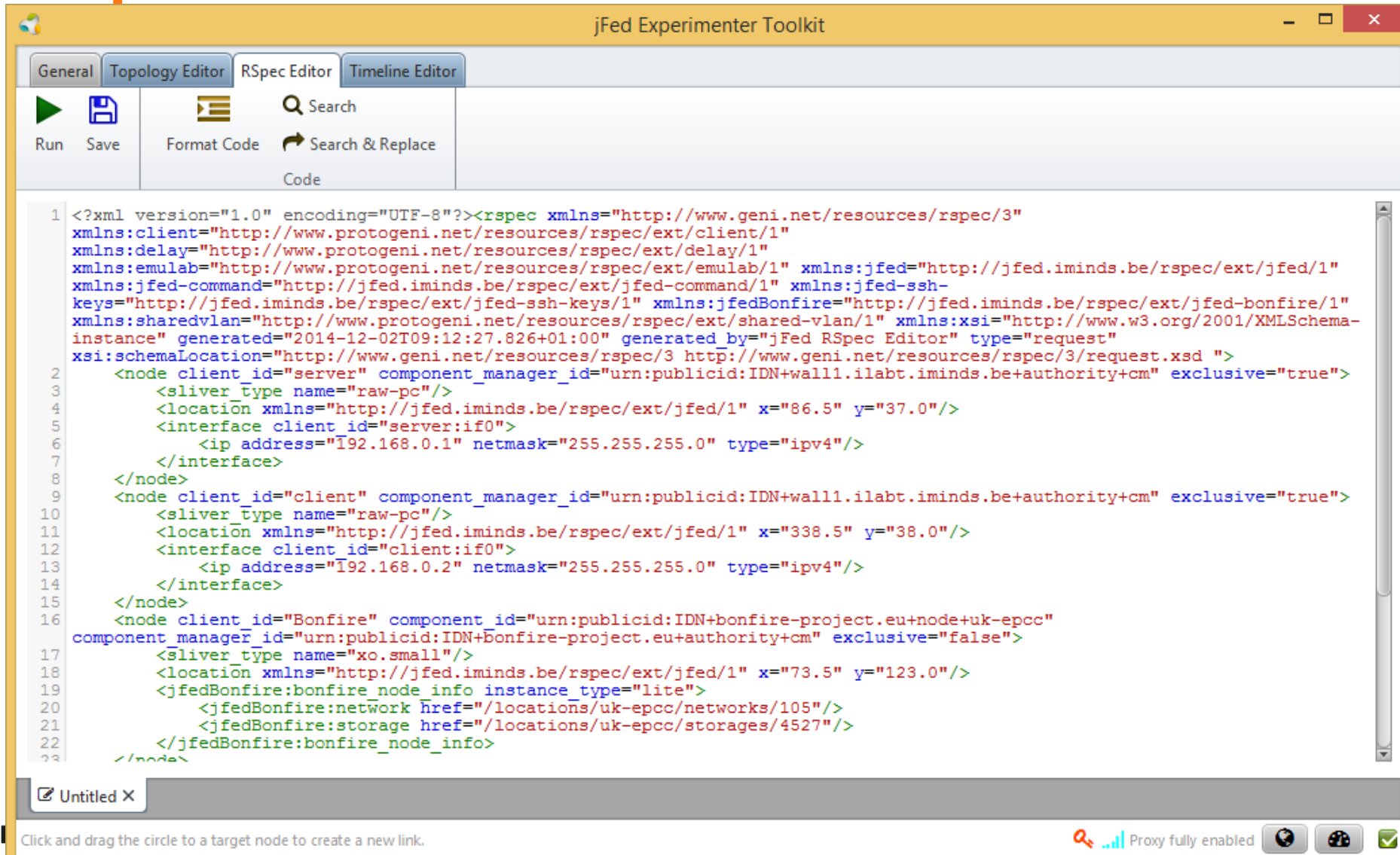
- Optional Xen VM requirements:
- Number of CPU Cores: 1
- Memory (MB): 512
- Disk Capacity (GB): 8

More information in the [ilab.t](#) documentation

The bottom of the interface shows a status bar with the text "Click and drag the circle to a target node to create a new link." and a "Proxy fully enabled" indicator.

# Raw RSpec editing: “support everything”

## RSpec = Resource Specification: describes experiment



The screenshot displays the jFed Experimenter Toolkit interface, specifically the RSpec Editor tab. The window title is "jFed Experimenter Toolkit". The interface includes a menu bar with "General", "Topology Editor", "RSpec Editor", and "Timeline Editor". Below the menu bar is a toolbar with icons for "Run", "Save", "Format Code", "Search", and "Search & Replace". The main area shows an XML document being edited, with line numbers 1 through 23 visible on the left. The XML code defines an RSpec with various namespaces and elements, including nodes for "server", "client", and "Bonfire".

```
1 <?xml version="1.0" encoding="UTF-8"?><rspec xmlns="http://www.geni.net/resources/rspec/3"
  xmlns:client="http://www.protogeni.net/resources/rspec/ext/client/1"
  xmlns:delay="http://www.protogeni.net/resources/rspec/ext/delay/1"
  xmlns:emulab="http://www.protogeni.net/resources/rspec/ext/emulab/1" xmlns:jfed="http://jfed.iminds.be/rspec/ext/jfed/1"
  xmlns:jfed-command="http://jfed.iminds.be/rspec/ext/jfed-command/1" xmlns:jfed-ssh-
  keys="http://jfed.iminds.be/rspec/ext/jfed-ssh-keys/1" xmlns:jfedBonfire="http://jfed.iminds.be/rspec/ext/jfed-bonfire/1"
  xmlns:sharedvlan="http://www.protogeni.net/resources/rspec/ext/shared-vlan/1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
  instance" generated="2014-12-02T09:12:27.826+01:00" generated_by="jFed RSpec Editor" type="request"
  xsi:schemaLocation="http://www.geni.net/resources/rspec/3 http://www.geni.net/resources/rspec/3/request.xsd ">
2   <node client_id="server" component_manager_id="urn:publicid:IDN+wall1.ilabt.iminds.be+authority+cm" exclusive="true">
3     <sliver_type name="raw-pc"/>
4     <location xmlns="http://jfed.iminds.be/rspec/ext/jfed/1" x="86.5" y="37.0"/>
5     <interface client_id="server:if0">
6       <ip address="192.168.0.1" netmask="255.255.255.0" type="ipv4"/>
7     </interface>
8   </node>
9   <node client_id="client" component_manager_id="urn:publicid:IDN+wall1.ilabt.iminds.be+authority+cm" exclusive="true">
10    <sliver_type name="raw-pc"/>
11    <location xmlns="http://jfed.iminds.be/rspec/ext/jfed/1" x="338.5" y="38.0"/>
12    <interface client_id="client:if0">
13      <ip address="192.168.0.2" netmask="255.255.255.0" type="ipv4"/>
14    </interface>
15  </node>
16  <node client_id="Bonfire" component_id="urn:publicid:IDN+bonfire-project.eu+node+uk-epcc"
  component_manager_id="urn:publicid:IDN+bonfire-project.eu+authority+cm" exclusive="false">
17    <sliver_type name="xo.small"/>
18    <location xmlns="http://jfed.iminds.be/rspec/ext/jfed/1" x="73.5" y="123.0"/>
19    <jfedBonfire:bonfire_node_info instance_type="lite">
20      <jfedBonfire:network href="/locations/uk-epcc/networks/105"/>
21      <jfedBonfire:storage href="/locations/uk-epcc/storages/4527"/>
22    </jfedBonfire:bonfire_node_info>
23  </node>
```

At the bottom of the window, there is a status bar with the text "Click and drag the circle to a target node to create a new link." and a "Proxy fully enabled" indicator.

# Debug by looking into API calls

The screenshot displays the 'jFed Calls Overview' application interface. On the left, a task list shows 16 tasks, with task 4, 'Fetch OCCl storages for location uk-epcc', highlighted in pink. The main area shows 'Task details' for the selected task, 'Create Sliver @ urn:publicid:IDN+wall1.ilabt.iminds.be+authority+cm', which is in a 'SUCCESS' state. It provides start and stop times and a duration of 28 seconds and 258 milliseconds. Below this, it lists tasks it depends on (tasks 8 and 9) and a box for tasks depending on it. The 'Task calls' section shows the API call 'Geni Aggregate Manager API v3 - Allocate' with various controls and a detailed XML response.

**Task details**

Name: Create Sliver @ urn:publicid:IDN+wall1.ilabt.iminds.be+authority+cm  
State: SUCCESS  
Start Time: Tue Dec 02 09:13:54 CET 2014  
Stop Time: Tue Dec 02 09:14:24 CET 2014  
Duration: 28 seconds and 258 milliseconds

This task depends on:

- 8 Get Slice Credential urn:publicid:IDN+wall2.ilabt.iminds.be:bvermeul+slice+wa
- 9 Get User SSH Keys

Tasks depending on this task:

...

**Task calls**

Geni Aggregate Manager API v3 - Allocate

Save all details: as text... as xml... Request size (byte): 9386 Reply size(byte): 3124

Connection HTTP Request HTTP Reply XmlRpc Request XmlRpc Reply Geni Reply Value Geni Reply Code & Output Processed Geni Reply Value ProtoGr

XmlRpc HashTable Received:

```
{
  "output": "",
  "code": {
    "protogeni_error_url": "https://www.wall1.ilabt.iminds.be/spewlogfile.php?logfile=16496d1f6e46b4d21f6f1fb3674236c0",
    "protogeni_error_log": "urn:publicid:IDN+wall1.ilabt.iminds.be+log+16496d1f6e46b4d21f6f1fb3674236c0",
    "am_type": "protogeni",
    "geni_code": 0,
    "am_code": 0
  },
  "value": {
    "geni_slivers": [
      {
        "geni_sliver_urn": "urn:publicid:IDN+wall1.ilabt.iminds.be+sliver+28295",
        "geni_allocation_status": "geni_allocated",
        "geni_expires": "2014-12-02T08:24:07Z"
      }
    ],
    "geni_rspec": "<rspec xmlns='http://www.geni.net/resources/rspec/3/' xmlns:emulab='http://www.protogeni.net/resources/rspec/ext/emulab/1' xmlns:client='http://www.geni.net/resources/rspec/ext/client/1'>
<node client_id='node0' exclusive='true' component_manager_id='urn:publicid:IDN+wall1.ilabt.iminds.be+authority+cm' component_id='urn:publicid:IDN+wall1.ilabt.iminds.be+sliver+28295'>
  <sliver_type name='raw-pc'/>
  <location xmlns='http://jfed.iminds.be/rspec/ext/jfed/1' x='104.0' y='105.0'/>
  <emulab:vnode name='n142-07a'/>
</node>
</rspec>"
  }
}
```

# Bug reports and support

The screenshot displays the jFed Experimenter Toolkit interface. The main window is titled "jFed Experimenter Toolkit" and features a menu bar with "General", "Topology Viewer", "RSpec Viewer", and "Timeline Viewer". Below the menu bar is a toolbar with icons for "New", "Open", "Open URL", "Save", "Run", "Update Status", "Terminate", "Recover", "Open Shared", "Preferences", "Report a bug", "Documentation", and "About". The main workspace shows a network diagram with nodes "server" and "client" connected by "link0", and other nodes like "Bonfire" and "node3". A progress log at the bottom left shows a list of tasks, all of which are completed with green checkmarks. A "jFed Bug Report" dialog box is open in the foreground, containing the following information:

- Bug description:** everything seems okay
- jFed version:** 5.3.1-SNAPSHOT (rev. #2179 at 2014-11-08 23:53:06) - build #5
- Environment:** Windows 8 6.2 x86 - Java 1.7.0\_71 (Oracle Corporation)
- Reporter credential:** urn:publicid:IDN+wall2.ilabt.iminds.be+user+bvermeul
- Reporter email address:** bvermeul@wall2.ilabt.iminds.be
- Included calls:** 55 calls

The "55 calls" text is circled in blue. The dialog box also includes "Submit" and "Cancel" buttons at the bottom right. At the bottom of the main window, a status bar indicates: "This experiment run will expire in 1 hour, 57 minutes and 31 seconds."

# RSpec and tutorial/classes world

The image shows the jFed Experimenter Toolkit interface. The window title is "jFed Experimenter Toolkit". The interface is divided into several sections:

- General** (selected): Contains icons for New, Open, Open URL, and Save.
- Topology Editor** (selected): Contains icons for Run, Update Status, Terminate, Recover, and Open Shared.
- RSpec Editor** (selected): Contains icons for Preferences and Report a bug.
- Timeline Editor** (selected): Contains icons for Documentation and About.

The main workspace displays a network diagram with four nodes (node0, node1, node2, node3) and three links (link4, link5, link6). Node0 is at the top, connected to node1 via link4, node2 via link5, and node3 via link6. Node1 is on the left, node2 is in the middle, and node3 is on the right.

A dialog box titled "Open Experiment Definition" is open in the foreground. It contains the text "Please enter the URL:" and a text input field with the URL `http://jfed.iminds.be/ovs.rspec`. The dialog has "OK" and "Cancel" buttons.

The bottom status bar shows the following tabs: "Untitled", "Untitled1", "wal1", and "ovs X". The system tray at the bottom right shows "Proxy fully enabled" and several system icons.

# jFed probe

The screenshot displays the jFed probe application interface. On the left, a tree view shows the hierarchy of Aggregate Manager APIs, with 'Aggregate Manager v3' expanded to show the 'getVersion' API selected. The main panel on the right is configured for the 'getVersion' command. It shows the 'Server to use' as 'Server of Known Authority', the 'Authority' as 'iMinds Virtual Wall 2', and the 'Server URL' as 'https://www.wall2.ilabt.iminds.be:12369/protogeni/xmlrpc/am/3.0'. The 'Command' is set to 'getVersion' and the 'Arguments' table is empty. Below the configuration, a 'Call' button is visible. The bottom section of the interface shows the results of the call, including the request size (253 bytes) and reply size (7473 bytes). The 'XMLRpc HashTable Received' section displays the following JSON response:

```
{
  "output": "",
  "geni_api": 3,
  "code": {
    "protogeni_error_url": "https://www.wall2.ilabt.iminds.be/spewlogfile.php3?logfile=b30808cb31a3443b7d2cf53ce7eed2a6",
    "protogeni_error_log": "urn:publicid:IDN+wall2.ilabt.iminds.be+log+b30808cb31a3443b7d2cf53ce7eed2a6",
    "am_type": "protogeni",
    "geni_code": 0,
    "am_code": 0
  },
  "value": {
    "url": "https://www.wall2.ilabt.iminds.be:12369/protogeni/xmlrpc/am",
    "hmn": "iminds-wall2.cm",
    "geni_credential_types": [

```

# jFed testing and monitoring

<https://flsmonitor.fed4fire.eu>

<http://monitor.ilabt.iminds.be>

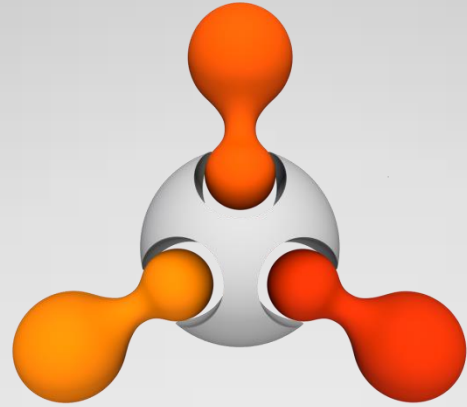
## API testing

Testbed Name	Ping latency (ms)	GetVersion Status	Free Resources	Internal testbed monitoring status	Last chec
BonFIRE	31.17	N/A	N/A	ok	2014-12-0
C-Lab	52.15	ok	113	ok	2014-12-0
FUSECO	15.77	ok	19	ok	2014-12-0
Koren	280.88	ok	3	N/A	N/A
NETMODE	61.02	ok	20	ok	2014-12-0
NITOS Broker	68.11	ok	38	ok	2014-12-0
NITOS SFAWrap	30.34	ok	111	ok	2014-12-0
Norbit	N/A	N/A	N/A	ok	2014-12-0
Ofelia (Bristol openflow)	16.94	ok	48	ok	2014-12-0
Ofelia (Bristol vtam)	16.92	ok	2	ok	2014-12-0
Ofelia (i2CAT openflow)	16.92	ok	5	ok	2014-12-0
Ofelia (i2CAT vtam)	16.98	ok	6	ok	2014-12-0
Planetlab Europe	30.36	ok	273	ok	2014-12-0
SmartSantander	58.9	ok	0	ok	2014-12-0
Virtual Wall 1	0.1	ok	70	N/A	N/A
Virtual Wall 2	0.12	ok	64	ok	2014-12-0
Virtual Wall 2 (openflow)	0.55	ok	2	ok	2014-12-0
w-iLab.t 2	4.71	ok	61	ok	2014-12-0

- ✓ setUp
- ✓ getVersion
- ✓ getTestUserCredential
- ✓ getTestUserInfo
- ✓ retrieveCredentialSomehow
- ✗ createProject
- ✓ createSlice
- ✗ lookupProjectsByUrnNoFilter
- ✗ lookupProjectsByNameNoFilter
- ✓ getSliceCredentials
- ✗ updateProject
- ✓ lookupSlicesNoFilter
- ✗ lookupProjectsNoFilterAfterUpdate
- ✓ updateSlice
- ✗ lookupProjectMembers
- ✗ lookupProjectsForMember
- ✓ lookupSlicesNoFilterAfterUpdate
- ✓ lookupSliceMembers

Test Name	Last Test Start Time (CET)	Last Test Duration	Last Partial Success	Last Full Success	Time since last Failure	Last Log	History
Confine	2014-12-01 21:05:03	10 minutes and 38 seconds	SUCCESS	SUCCESS	4 days and 11 hours	log	history
Fuseco	2014-12-01 21:27:02	27 seconds	FAILURE	FAILURE		log	history
NETMODE	2014-12-01 22:36:37	1 minute and 40 seconds	SUCCESS	FAILURE		log	history
Nitos Broker	2014-12-01 22:38:18	1 minute and 18 seconds	SUCCESS	FAILURE		log	history
Nitos SFAWrap	2014-12-01 22:39:37	14 seconds	FAILURE	FAILURE		log	history
Planetlab Europe	2014-12-02 03:43:26	10 minutes and 49 seconds	SUCCESS	SUCCESS	6 days and 21 hours	log	history
Virtual Wall	2014-12-02 03:39:06	2 minutes and 51 seconds	SUCCESS	SUCCESS	3 days and 22 hours	log	history
Virtual Wall	2014-12-02 03:36:25	2 minutes and 40 seconds	SUCCESS	SUCCESS	3 days and 22 hours	log	history
Virtual Wall 1	2014-12-02 03:32:11	4 minutes and 12 seconds	SUCCESS	SUCCESS	3 days and 22 hours	log	history
Virtual Wall 1	2014-12-02 03:29:14	2 minutes and 56 seconds	SUCCESS	SUCCESS	3 days and 22 hours	log	history
Wilab.t	2014-12-02 03:42:42	44 seconds	WARN	WARN		log	history
Wilab.t	2014-12-02 03:41:58	44 seconds	WARN	WARN		log	history





**FED4FIRE**

**How does the AM work**

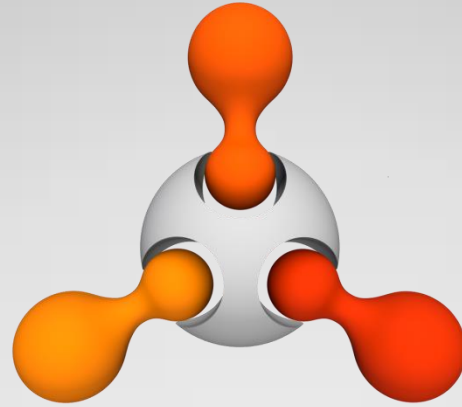


# AM

- AM v2
  - [http://groups.geni.net/geni/wiki/GAPI\\_AM\\_API\\_V2](http://groups.geni.net/geni/wiki/GAPI_AM_API_V2)
- AM v3
  - [http://groups.geni.net/geni/wiki/GAPI\\_AM\\_API\\_V3](http://groups.geni.net/geni/wiki/GAPI_AM_API_V3)
- Upcoming AM, but can help in understanding it better (does not differ much from AM v3):
  - <https://fed4fire-testbeds.ilabt.iminds.be/asciidoc/federation-am-api.html>
  - <https://github.com/open-multinet/federation-am-api>
  - (on github you can request for clarifications, report problems on the standard API description)

# Workflow

- 3 types of Rspecs: advertisement, request, manifest
- Getversion: informative
- Listresources: advertisement RSpec
- Createsliver (v2) vs allocate/provision/performoperationalaction (v3): send request, receive manifest
- SliverStatus (v2) vs Status (v3): check
- Listresources (v2) vs Describe (v3): get overview
- Renew: to extend duration
- DeleteSliver (v2) vs Delete (v3)



**FED4FIRE**

# How to implement the AM



# Possibilities for AM implementation

- If you have only hardware, no mgmt software for your testbed, pick testbed software which has an AM implementation and which is closest to your HW
  - Emulab (contact iMinds for more information)
  - Nitos Broker with OMF (contact University of Thessaly for more information)
  - Foam (openflow + flowvisor) (contact iMinds for more information)
  - GRAM (works with Openstack) (contact Inria Grid 5000 if you want more information)
- If you have software for managing your testbed, you can wrap it with the AM API:
  - SFAwrap (python) – contact UPMC/Inria Sophia Antipolis
  - Fiteagle (java) – contact TU Berlin
  - Geni Control Framework (GCF) – GENI BBN - <http://trac.gpolab.bbn.com/gcf>
- Implement yourself the AM API on top of an existing testbed
- Choice depends on what you have and what you want