MRuby Backend for OpenLDAP

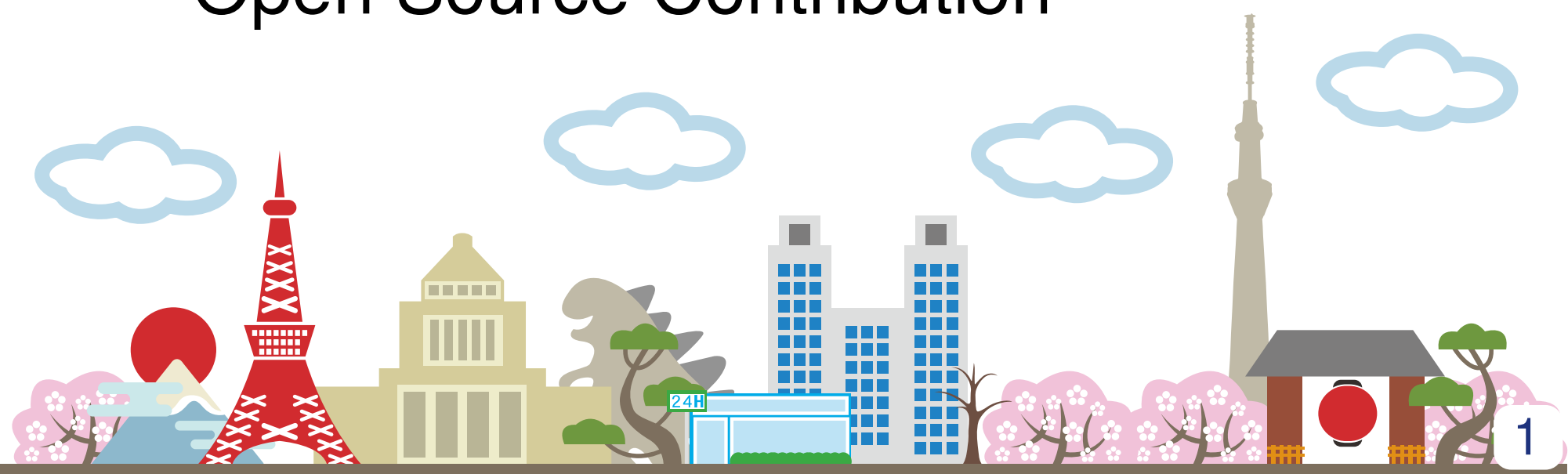
Open Source Solution Technology Corporation
HAMANO Tsukasa <hamano@osstech.co.jp>

LDAPCon 2017 Brussels October 2017

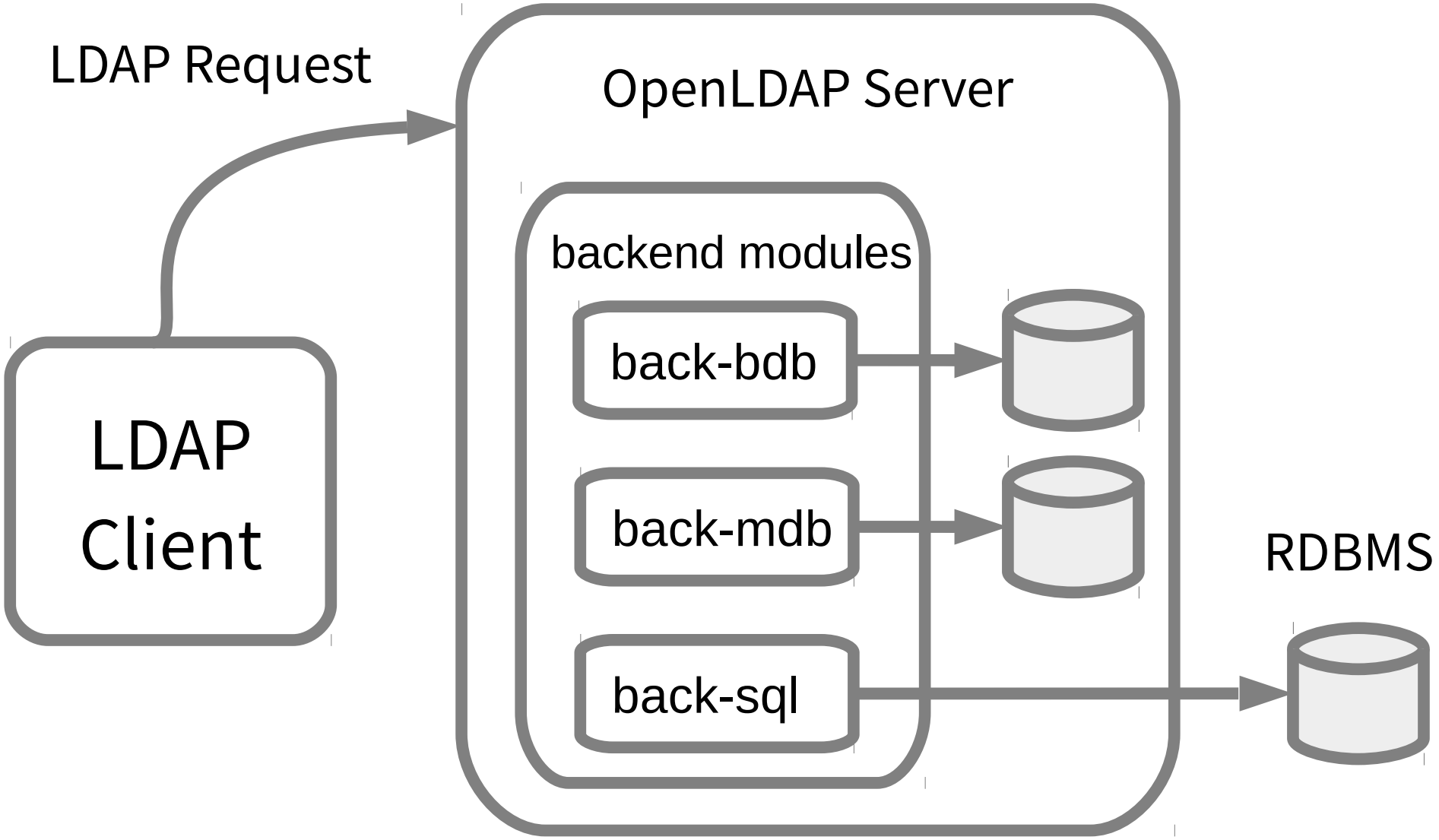
OSSTech

About OSSTech

- ID Management leading company in Japan.
- Storage Solution
- Open Source Contribution



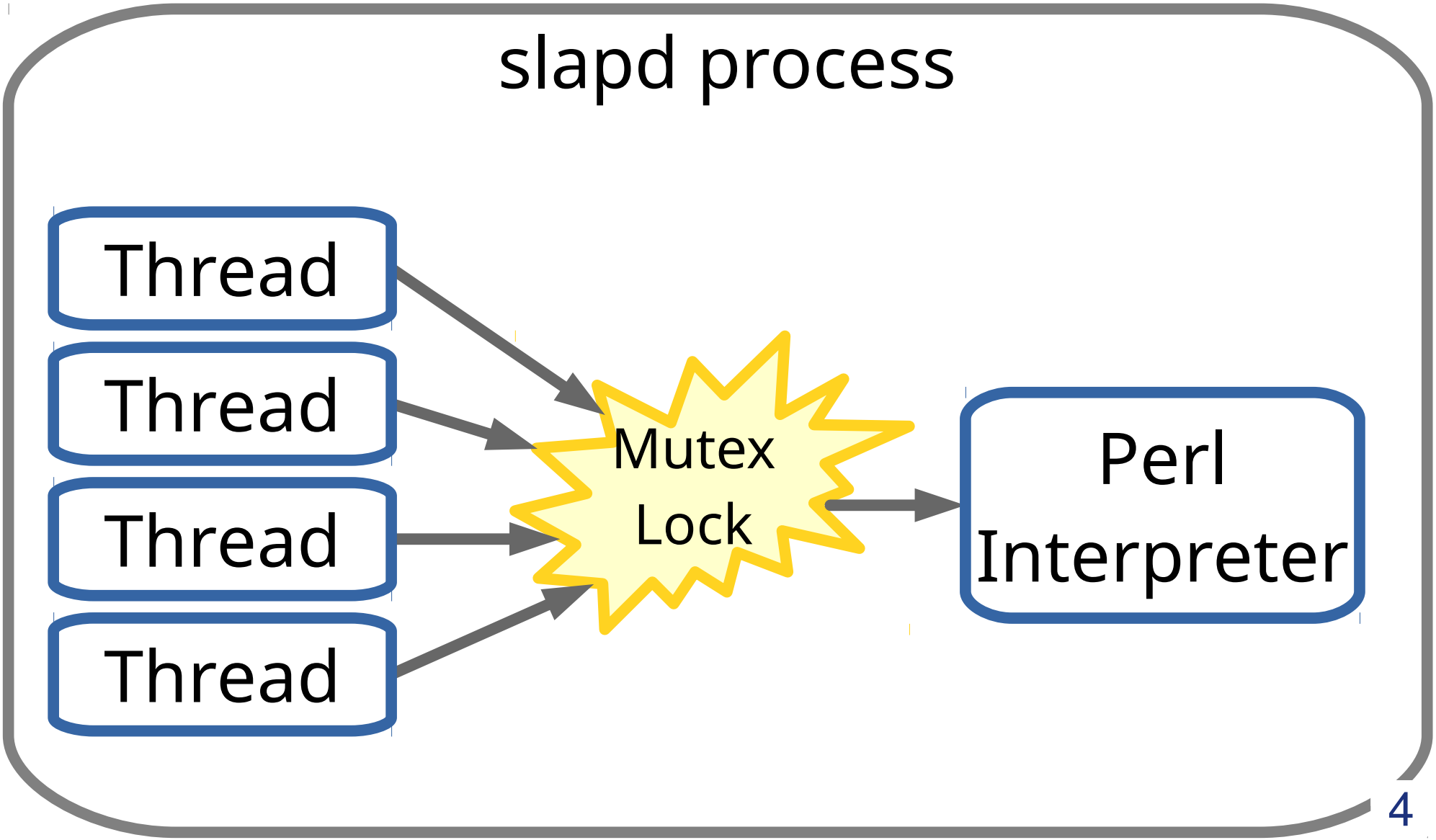
OpenLDAP backend



What's back-perl

- define LDAP handlers in Perl
- make non-LDAP datastore behave as LDAP server
- integrate with cloud services

Problem of back-perl



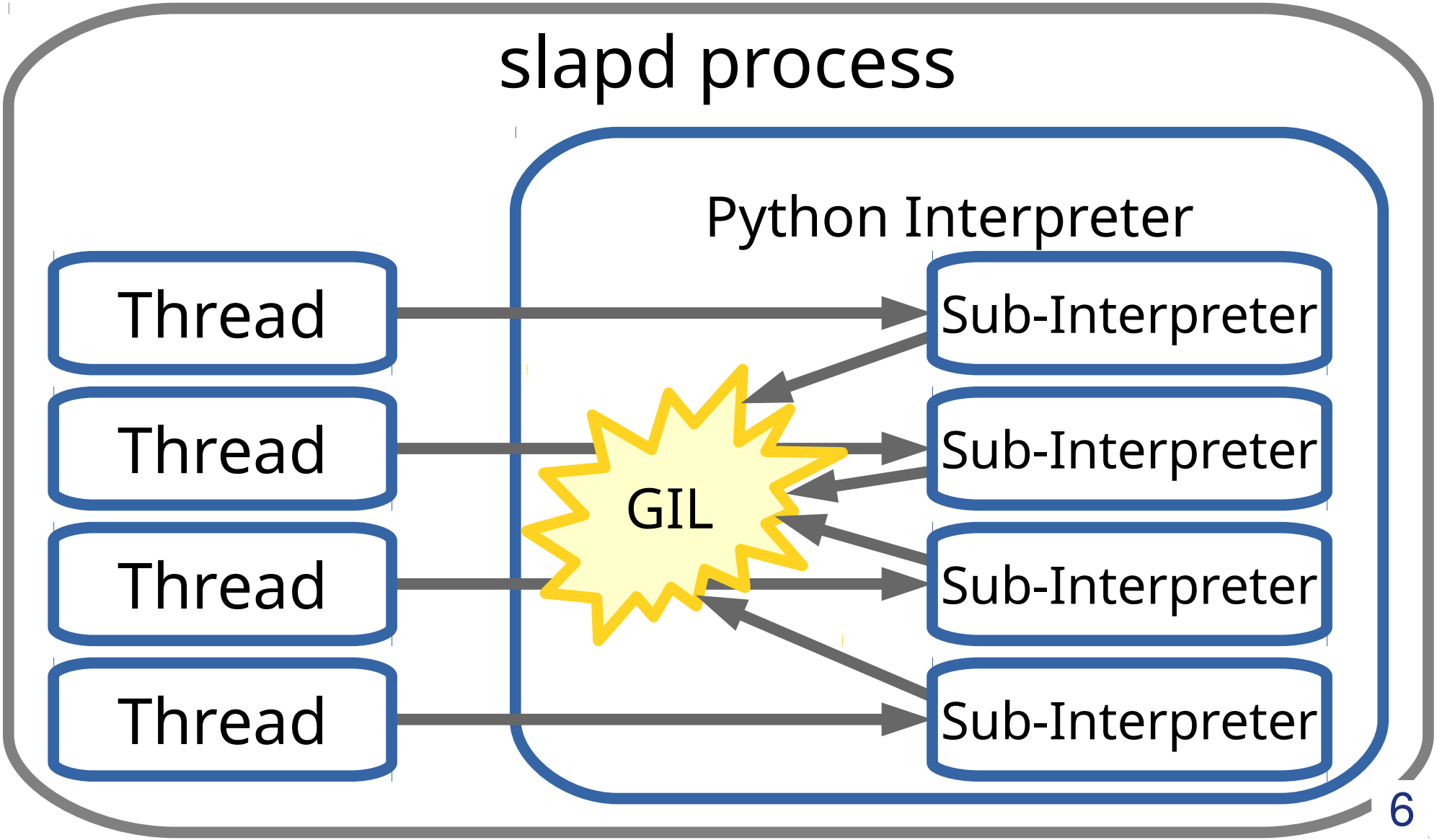
I ♥ Python

- Sub-interpreter support

Multiple python interpreters can be created in a process.

```
PyThreadState *Py_NewInterpreter()
```

Problem of back-python



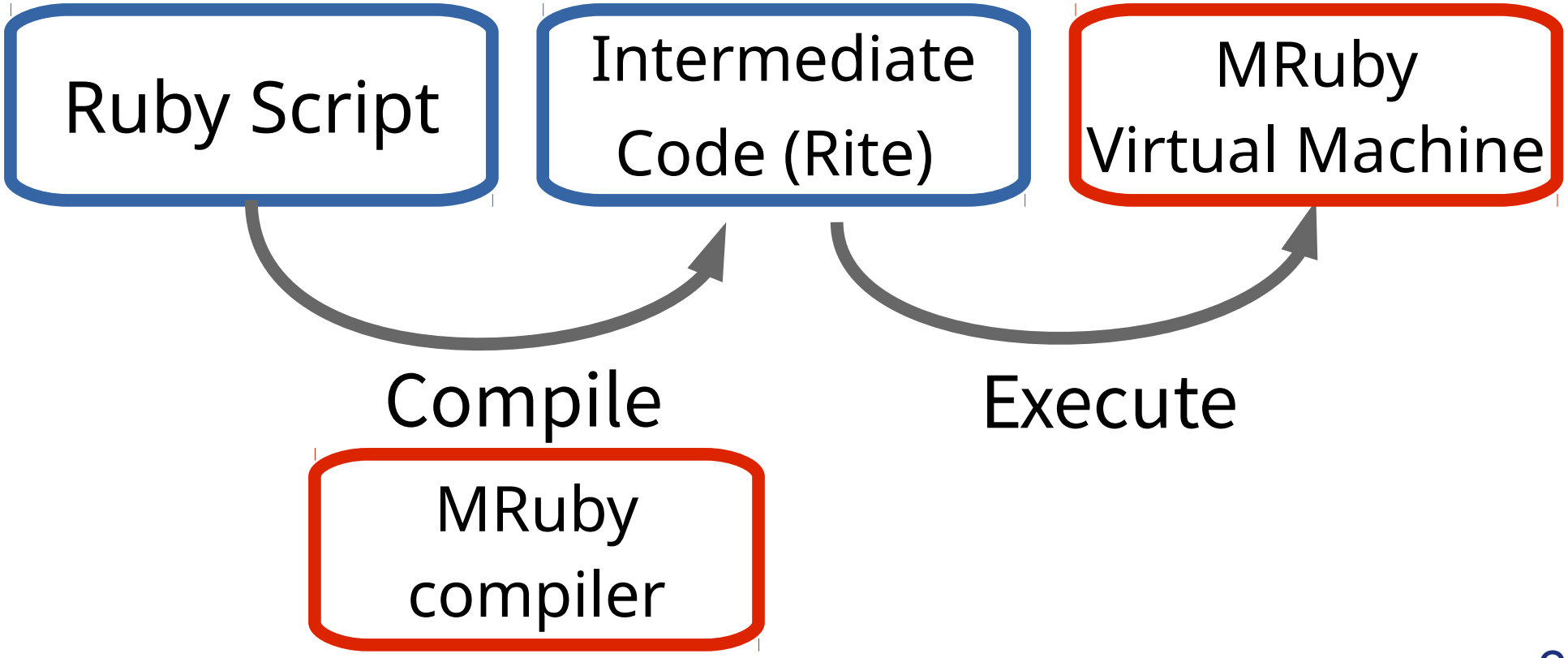
Ruby Implementations

- CRuby(MRI)
- JRuby
- IronRuby
- MRuby ⇐ NEW

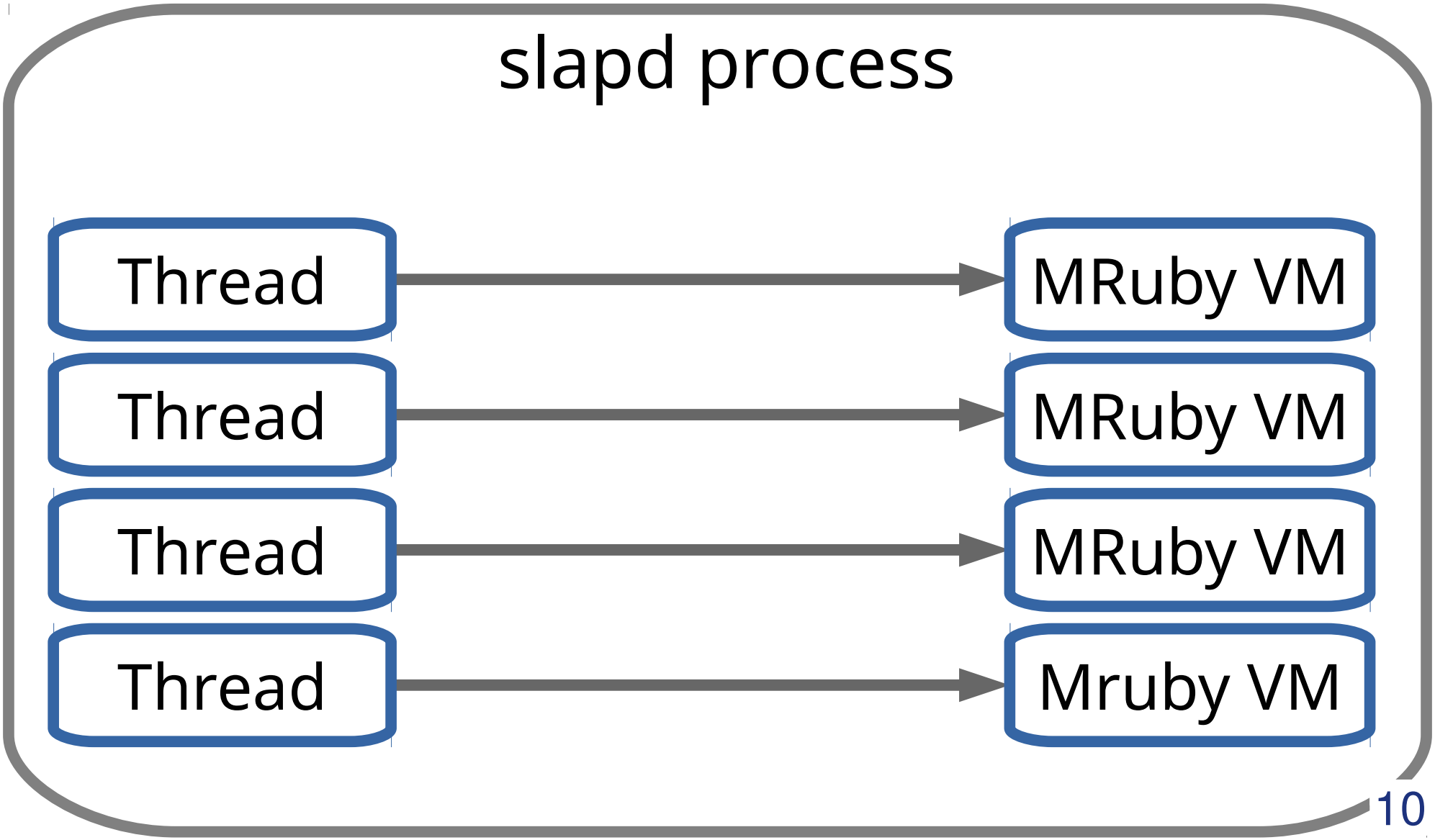
MRuby Features

- Small footprint
- Embed to application
- Soft realtime
- Portability

MRuby Virtual Machine



back-mruby


slapd process
Thread
Thread
Thread
Thread
MRuby VM
MRuby VM
MRuby VM
Mruby VM

Building

```
$ git clone -b mruby --depth=1 \
https://github.com/osstech-jp/openldap
$ cd openldap
$ autoconf
$ ./configure --enable-mruby
$ make
```

Requires: wget, bison, git, ruby

Configuration

In slapd.conf:

```
rubyfile example.rb
```

Tutorial: bind

```
def bind(op)
  if rand(2) == 0
    LDAP_SUCCESS
  else
    LDAP_INVALID_CREDENTIALS
  end
end
```

Tutorial: bind

```
def bind(op)
  users = {
    'uid=alice,dc=example,dc=com' =>
      'secret1',
    'uid=bob,dc=example,dc=com' =>
      'secret2',
  }
  ndn = op['ndn']
  if users[ndn] == op['req']['cred']
    LDAP_SUCCESS
  else
    LDAP_INVALID_CREDENTIALS
  end
end
```

Tutorial: search

```
def search(op)
  scope = op["req"]["scope"]
  if scope == LDAP_SCOPE_BASE
    search_base(op)
  elsif scope == LDAP_SCOPE_ONE
    search_one(op)
  else
    LDAP_NO_SUCH_OBJECT
  end
end
```

Tutorial: search

```
# Returns base entry.
def search_base(op)
  ldif = <<END_OF_LDIF
dn: dc=example,dc=com
objectClass: top
objectClass: dcObject
objectClass: organization
dc: example
o: example
END_OF_LDIF
  [ ldif ]
end
```

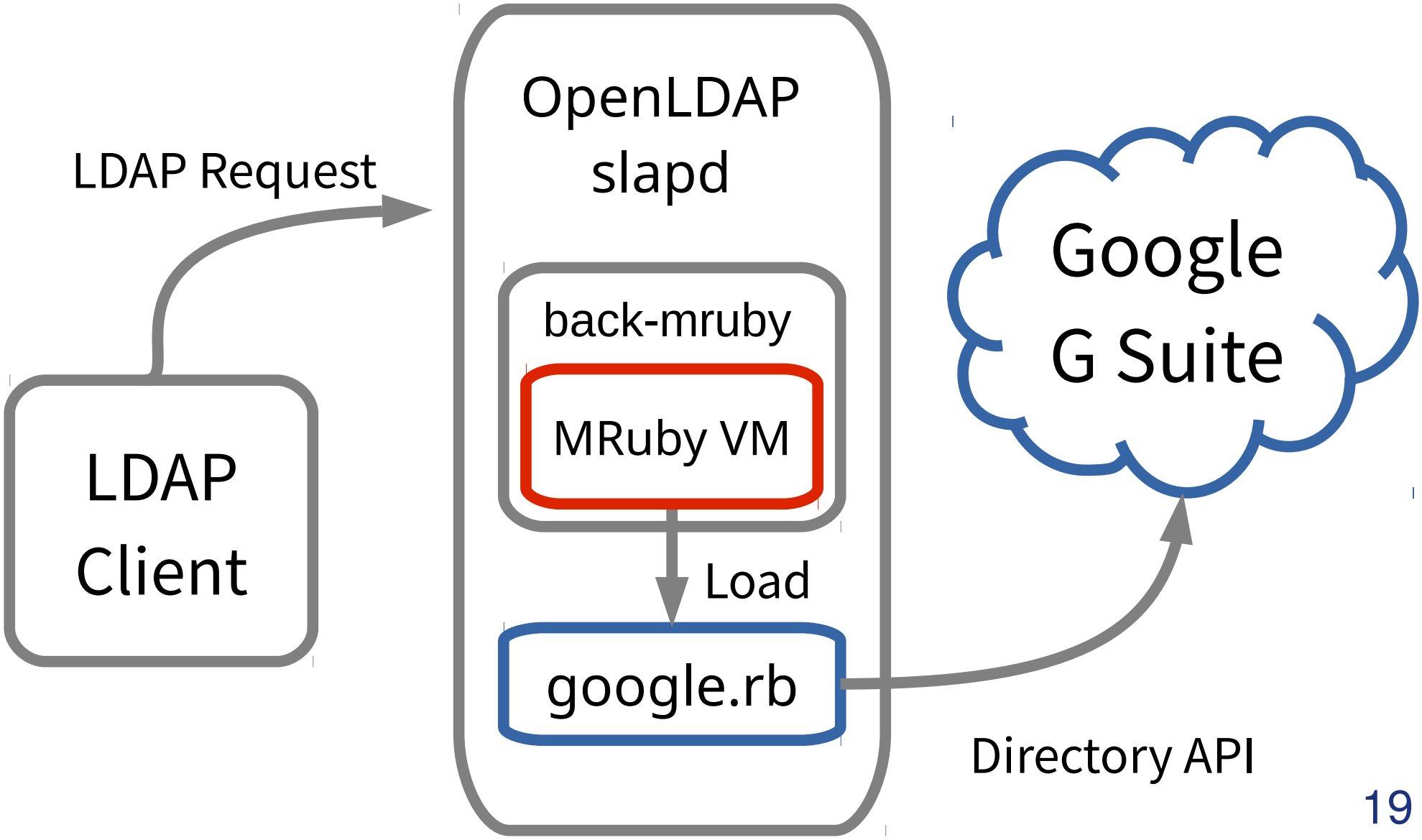
Tutorial: search

```
# Returns two user entries.
def search_one(op)
  entries = []
  ldif = <<END_OF_LDIF
dn: uid=%s,dc=example,dc=com
objectClass: top
objectClass: account
uid: %s
END_OF_LDIF
  entries << ldif % (["alice"] * 2)
  entries << ldif % (["bob"] * 2)
  entries
end
```

Tutorial: add

```
def add(op)
  p op
  # do the cooking
  LDAP_SUCCESS
end
```

Google G Suite Backend



Just Idea

- Amazon DynamoDB
- Google Cloud Storage
- NoSQL, RDBMS
- Salesforce

Future tasks

- Reusing VM
- Pagenation
- modify, modrdn

Other Works

- PBKDF2 module
- Datadog Agent
- WiredTiger Backend