BIND 10 Tidbits

Jelte Jansen

Internet Systems Consortium (ISC)

jelte@isc.org

Sponsors


cira
.ca


jPRS
JAPAN REGISTRY SERVICES


CNNIC
中国互联网络信息中心
China Internet Network Information Center


Afilias


.fr
AFNIC


.de
DENIC


RIPE
NCC


nominet


SIDN

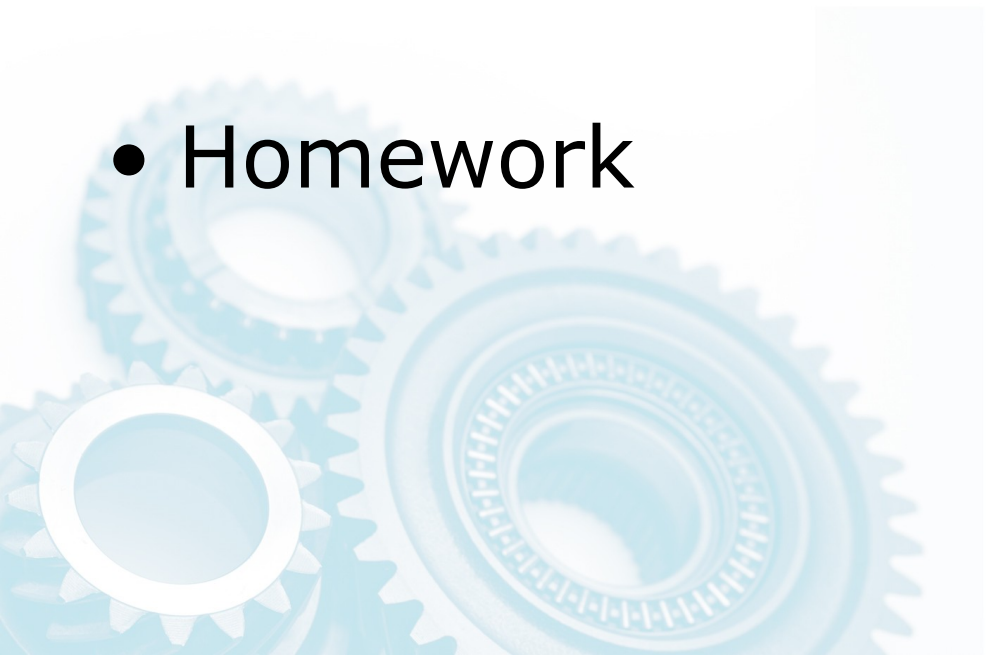

.zadna


registro.br


ISC

Contents

- Architecture
  (from space)
- DNS Library
  (C++/Python)
- Homework



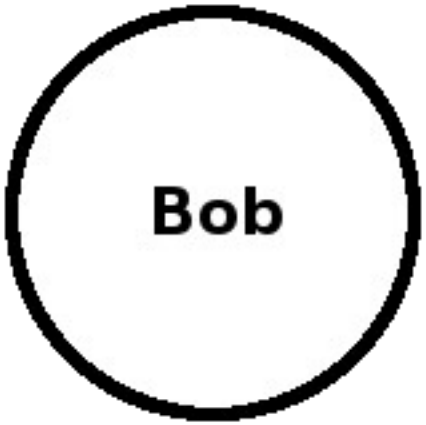
Architecture

- Cooperating processes
  (vs monolithic system)

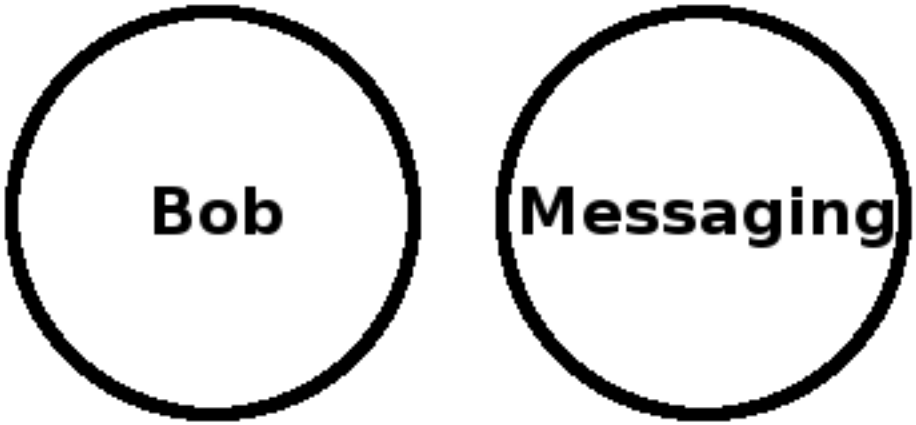
BIND 10 Core

BIND 10 Core

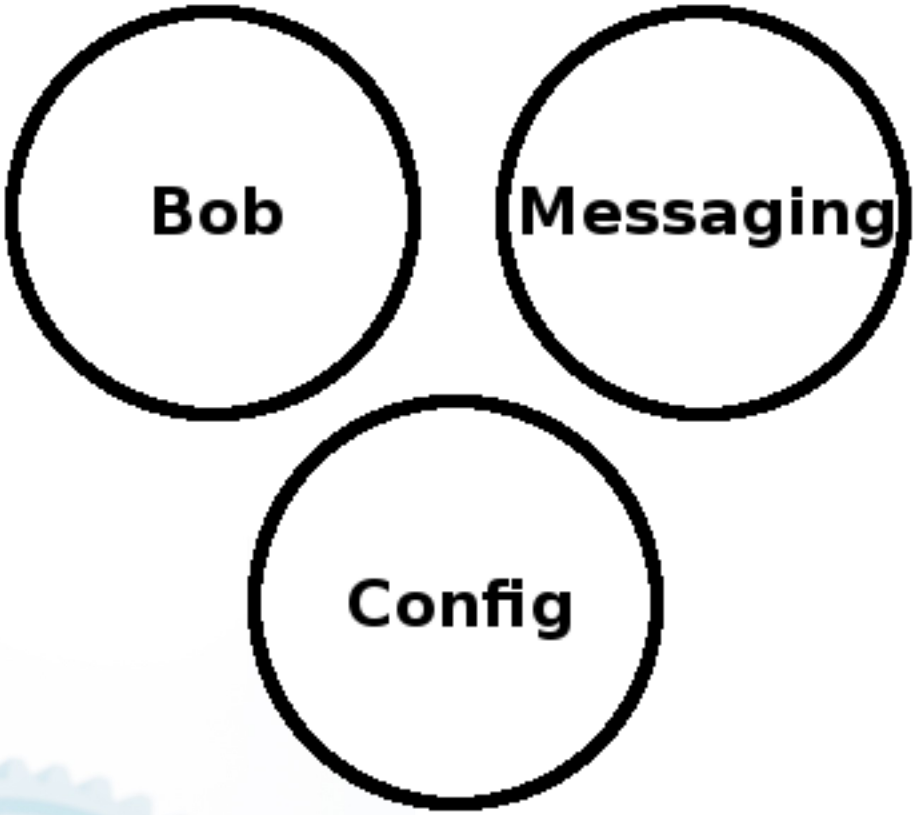
BIND 10 Core



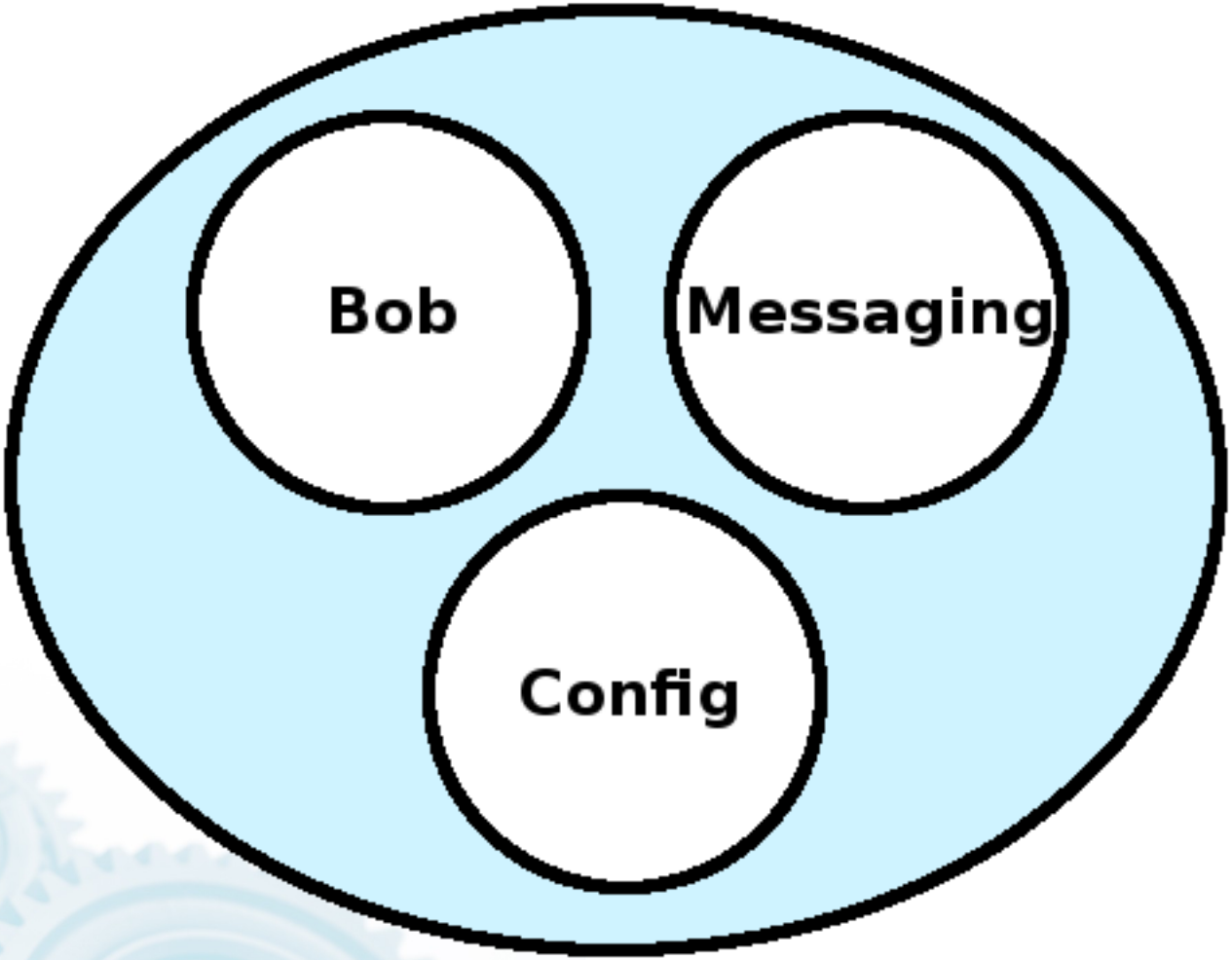
BIND 10 Core



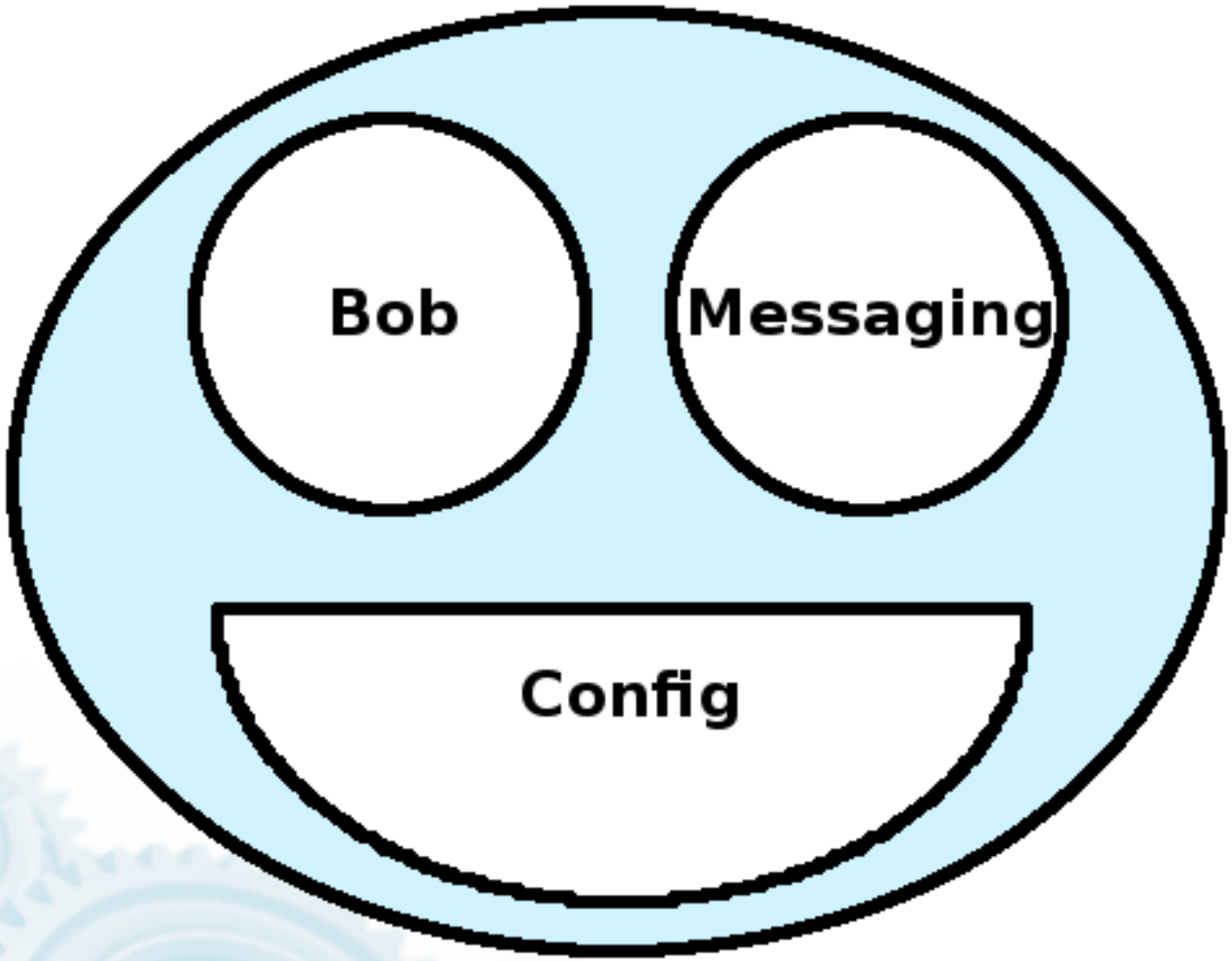
BIND 10 Core



BIND 10 Core

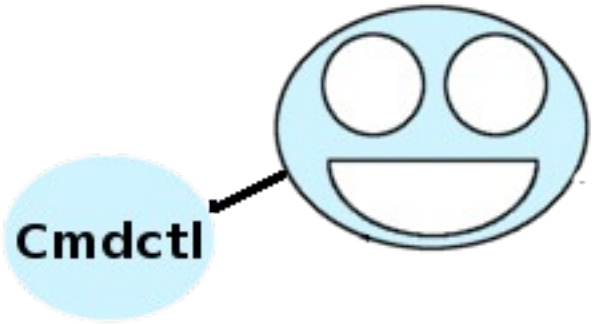


BIND 10 Core

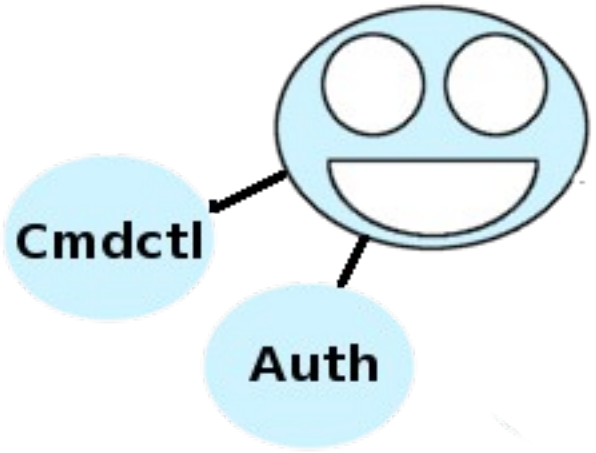


BIND 10 Modules

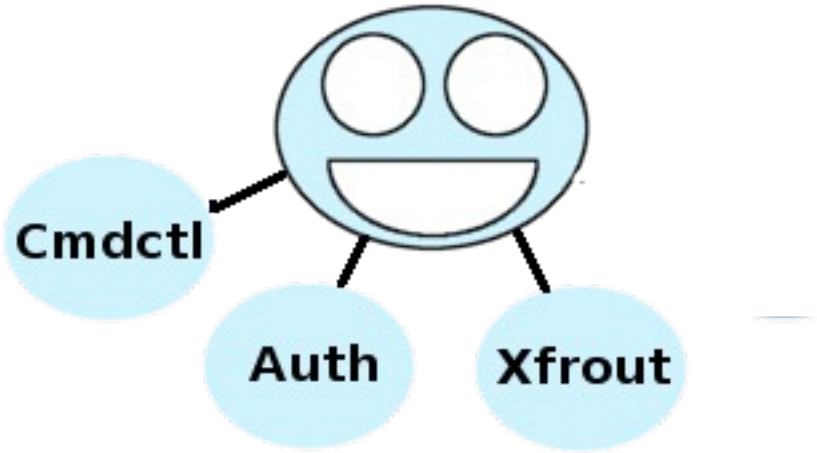
BIND 10 Modules



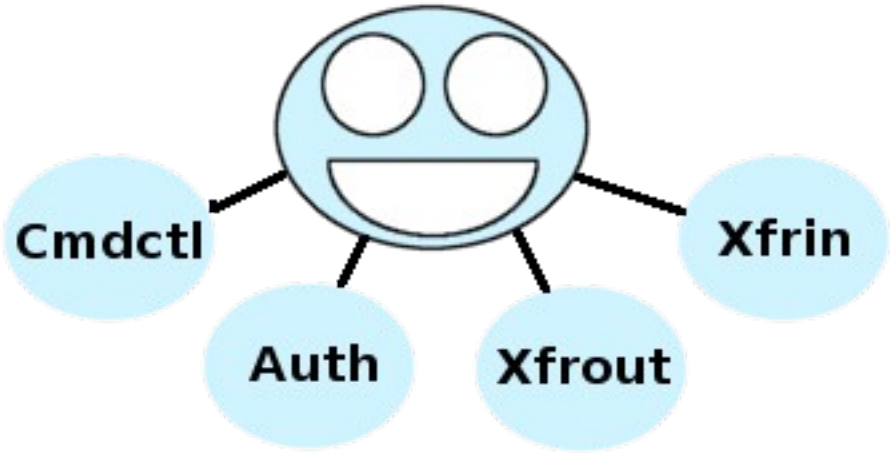
BIND 10 Modules



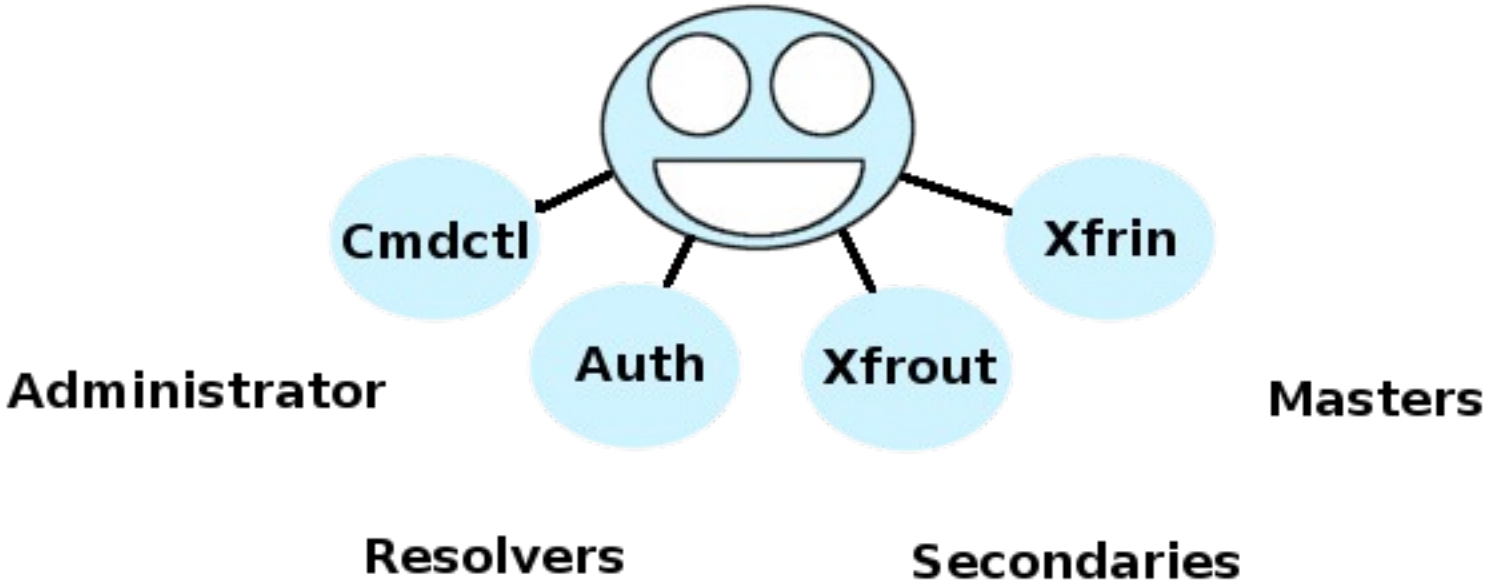
BIND 10 Modules



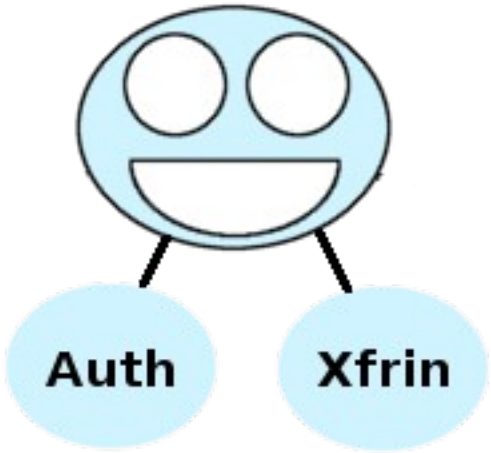
BIND 10 Modules



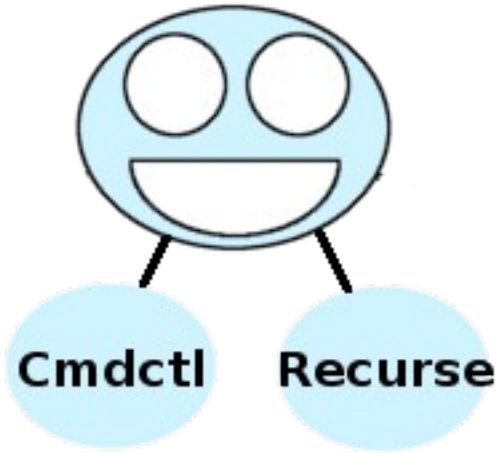
BIND 10 Modules



BIND 10 Modules

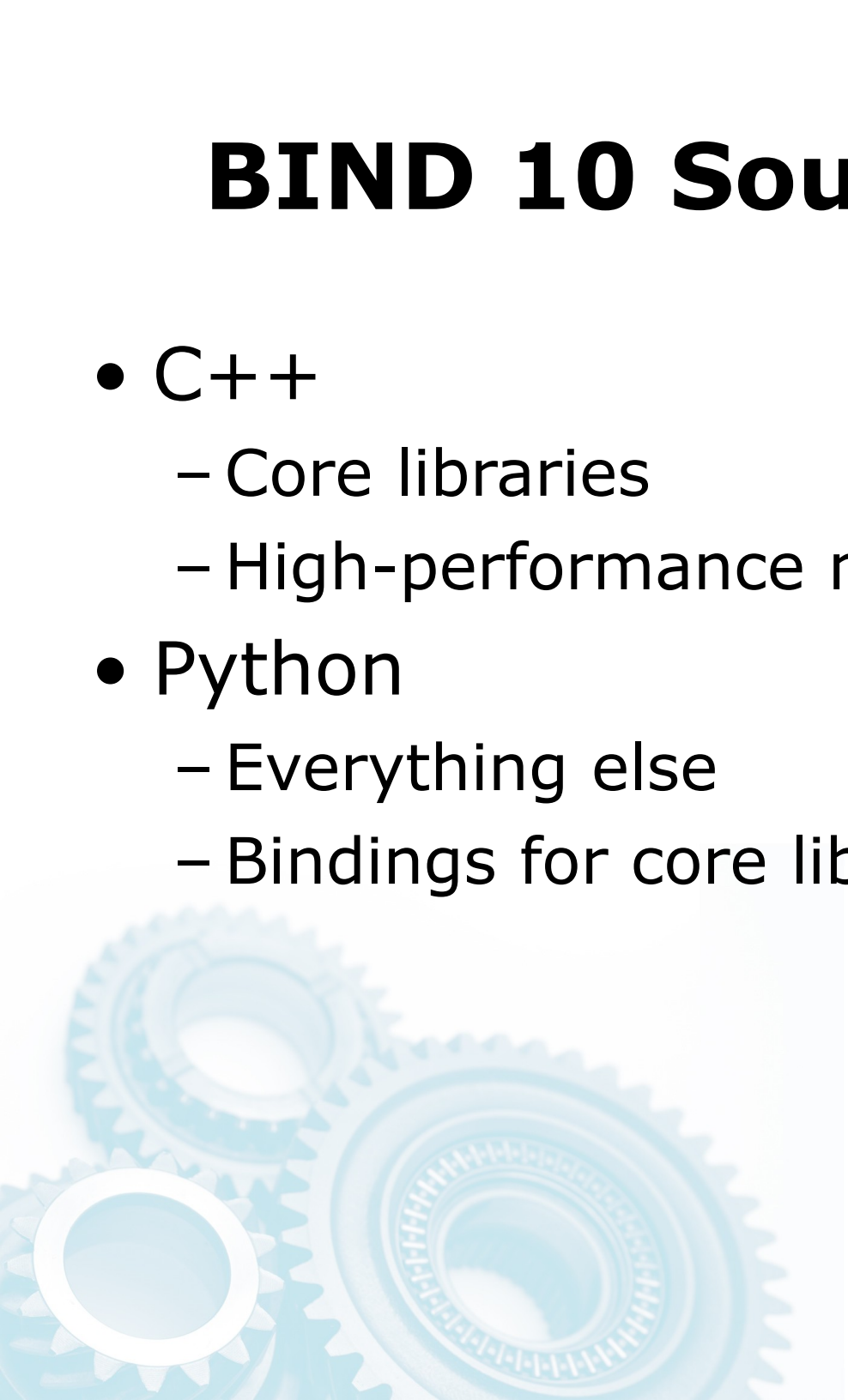


BIND 10 Modules

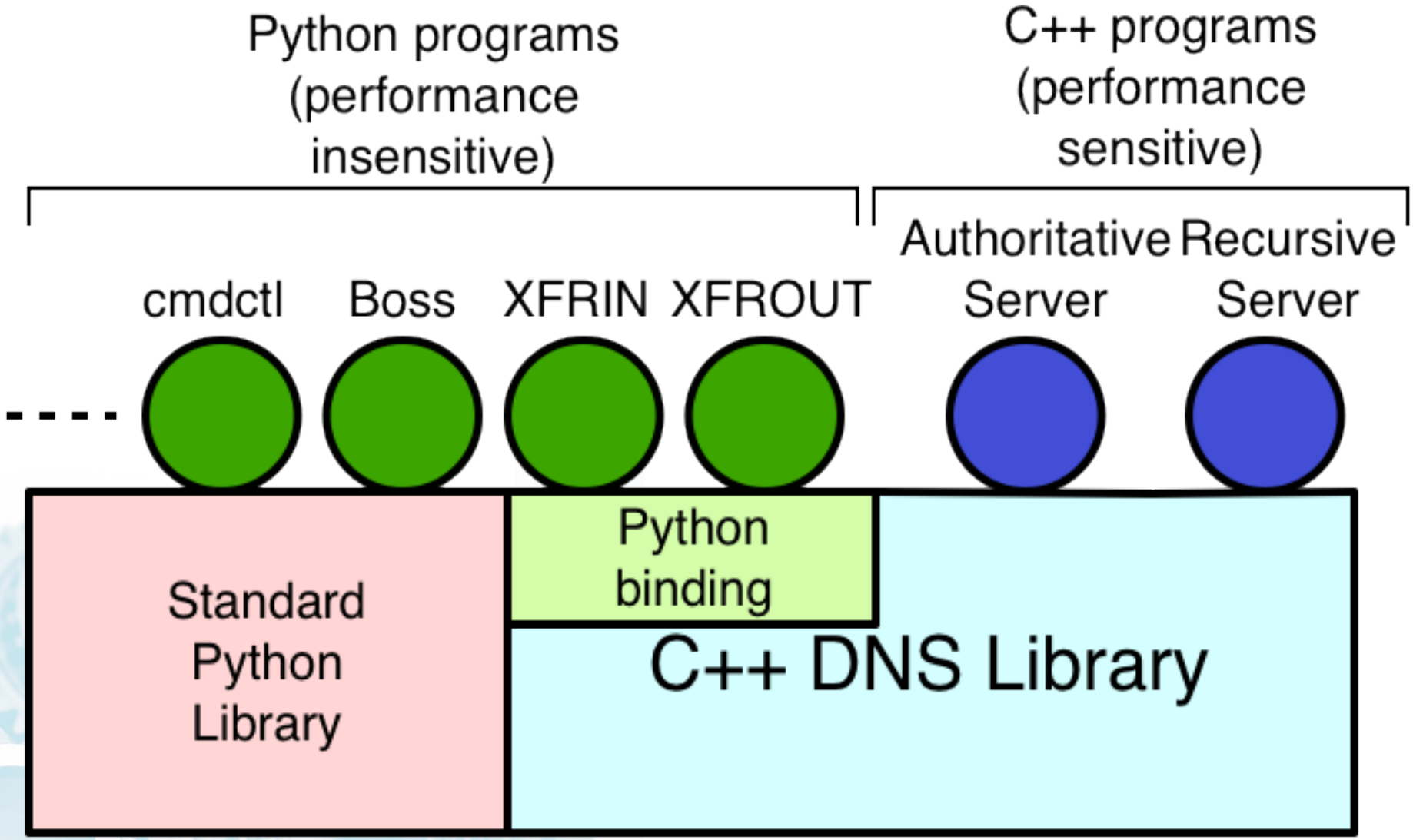


BIND 10 Source Languages

- C++
  - Core libraries
  - High-performance modules
- Python
  - Everything else
  - Bindings for core libraries



BIND 10 Source Languages



Python Binding Performance

- Goal: fast enough for performance insensitive, production quality apps
  - Should be much faster than script-only implementations
  - Can be much slower than optimized server implementations written in C/C++

Python Binding: Lies

- BIND 10 vs 3 popular libraries
  - Perl Net::DNS, dnspython, dnsruby
    - All script-based (full perl, etc)
- A simple benchmark test
  - For a toy DNS server returning pre-populated responses
    - http://bind10.isc.org/~jinmei/authors/

Python Binding: Benchmark

- BIND 10 vs 3 popular libraries
  - Perl Net::DNS, dnspython, dnsruby
    - All script-based (full perl, etc)
- A simple benchmark test
  - For a toy DNS server returning pre-populated responses
    - http://bind10.isc.org/~jinmei/authors/

Sample Code: BIND10 Python

```
while True:
    (data, fromaddr) = s.recvfrom(4096)
    msg.clear(Message.PARSE)
    Message.from_wire(msg, data)
    msg.make_response()
    if msg.get_rr_count(Section.QUESTION()) != 1:
        msg.set_rcode(Rcode.FORMERR())
    else:
        question = msg.get_question()[0]
        if question.get_name() == authors_name and \
                question.get_class() == authors_rrclass and \
                question.get_type() == authors_rrtype:
            msg.set_header_flag(MessageFlag.AA())
            msg.set_rcode(Rcode.NOERROR())
            msg.add_rrset(Section.ANSWER(), authors_rrset)
        else:
            msg.set_rcode(Rcode.NXDOMAIN())
    renderer = MessageRenderer()
    msg.to_wire(renderer)
    s.sendto(renderer.get_data(), fromaddr)
```

Sample Code: dnspython

```
while True:
    (data, fromaddr) = s.recvfrom(4096)
    request = message.from_wire(data)
    response = message.make_response(request)
    if len(request.question) != 1:
        response.set_rcode(rcode.FORMERR)
    else:
        question = request.question[0]
        if question.name == authors_name and \
                question.rdclass == authors_rrclass and \
                question.rdtype == authors_rrtype:
            response.set_rcode(rcode.NOERROR)
            response.flags = response.flags | flags.AA
            response.answer.append(authors_rrset)
        else:
            response.set_rcode(rcode.NXDOMAIN)
    s.sendto(response.to_wire(), fromaddr)
```

Sample Code: Net::DNS

```
for (;;) {
    my $buf;
    $sock->recv($buf, 4096);
    my ($packet, $err) = new Net::DNS::Packet(\$buf, 0);
    my @questions = $packet->question;
    $packet->header->qr(1);
    if (@questions != 1) {
        $packet->header->rcode("FORMERR");
    } else {
        $packet->header->aa(1);
        if ($questions[0]->qname eq $authors_name &&
            $questions[0]->qtype eq $authors_rrtype &&
            $questions[0]->qclass eq $authors_rrclass) {
           foreach my $answer_rr (@authors_rrs) {
              $packet->push("answer", $answer_rr);
           }
        } else {
           $packet->header->rcode("NXDOMAIN");
        }
    }
    $sock->send($packet->data);
}
```

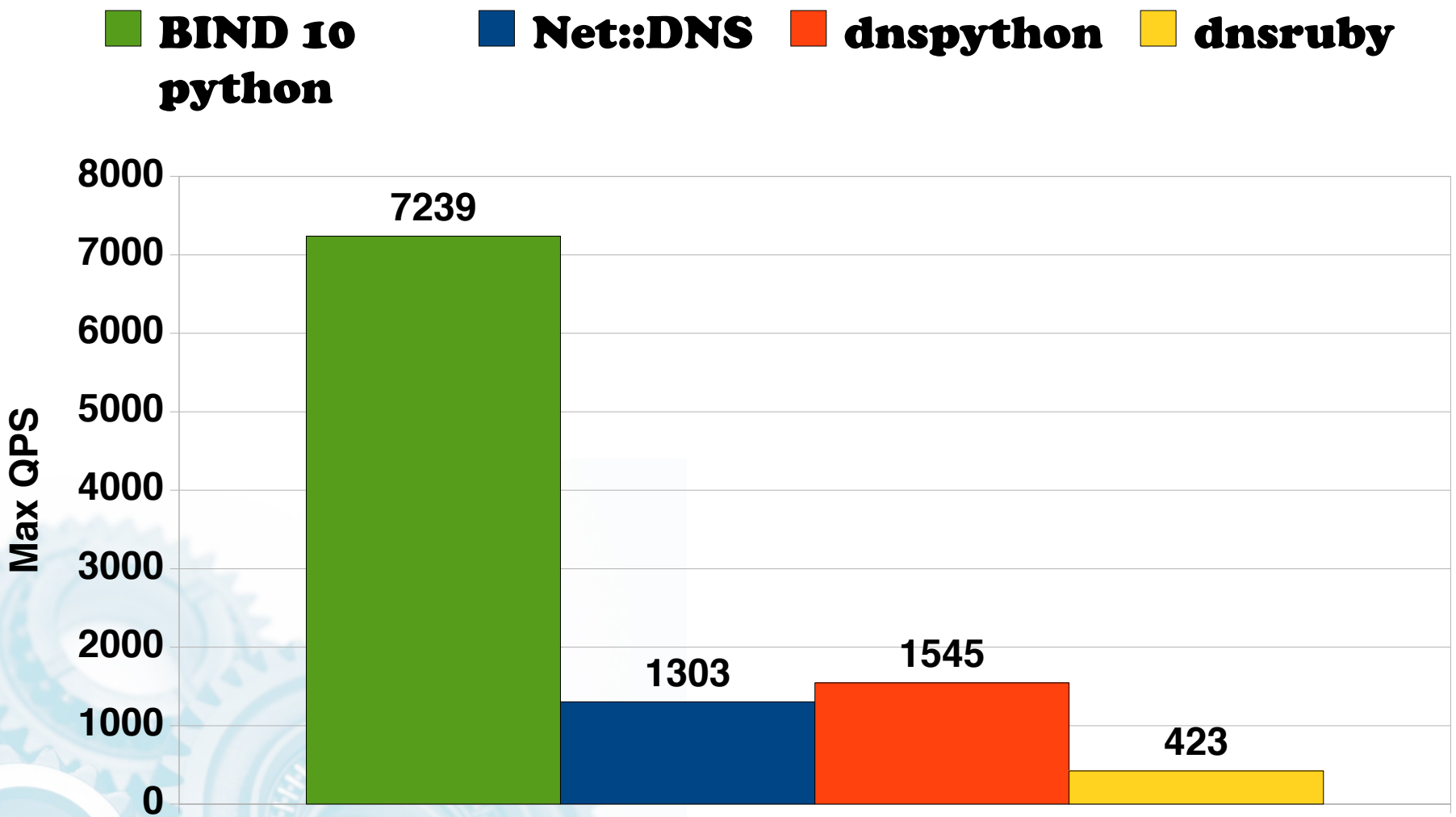
Sample Code: dnsruby

```
loop do
  data, fromhost = s.recvfrom(4096)
  msg = Dnsruby::Message::decode(data)
  if msg.header.qdcount != 1
    msg.header.rcode = Dnsruby::RCode.FORMERR
  else
    question = msg.question[0]
    if question.qname == authors_name and
        question.qclass == authors_rrclass and
        question.qtype == authors_rrtype
      msg.header.rcode = Dnsruby::RCode.NOERROR
      msg.header.qr = 1
      authors_rrs.each do |rr|
        msg.add_answer(rr)
      end
    else
      msg.header.rcode = Dnsruby::RCode.NXDOMAIN
    end
  end
  s.send(msg.encode(), 0, fromhost[3], fromhost[1])
end
```

Benchmark Setups

- Measured max possible QPS of each implementation
  - By sending a high rate of queries until the target drops them
  - Run on Jinmei's laptop:-)

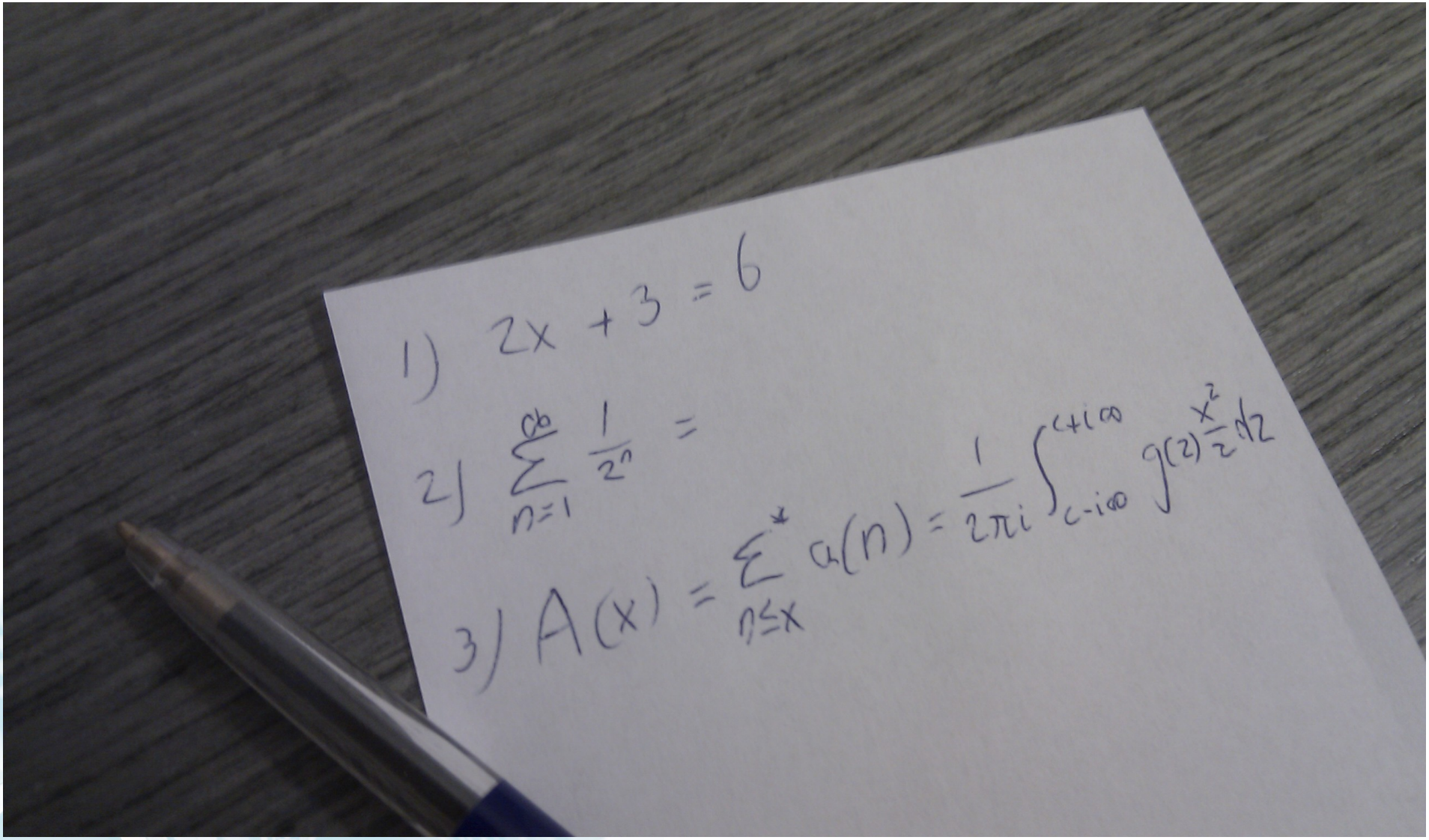
Benchmark Results



Python Binding Performance

- Goal: fast enough for performance insensitive, production quality apps
  - Should be much faster than script-only implementations
  - Can be much slower than optimized server implementations written in C/C++

Homework


1) $2x + 3 = 6$
2) $\sum_{n=1}^{\infty} \frac{1}{2^n} =$
3) $A(x) = \sum_{n \le x}^{*} a_1(n) = \frac{1}{2\pi i} \int_{c-i\infty}^{c+i\infty} g(z) \frac{x^z}{z} dz$

What do YOU want?

BIND 10 Survey

http://www.isc.org/announcement/give-your-input-future-bind

or follow the link at isc.org

and come talk to me!

I'll be here all week

BIND 10 Survey

http://www.isc.org/announcement/give-your-input-future-bind

or follow the link at isc.org

and come talk to me!

I'll be here the rest of the week

BIND 10 Survey

http://www.isc.org/announcement/give-your-input-future-bind

or follow the link at isc.org

and come talk to me!

I'll be here the rest of the day

Thank you!

http://www.isc.org/announcement/give-your-input-future-bind

or follow the link at isc.org

and drop me an e-mail!

jelte@isc.org