

BIND 10 プロジェクト

(の **C++/python DNS** ライブラリについて)

Internet Systems Consortium (ISC)

神明達哉

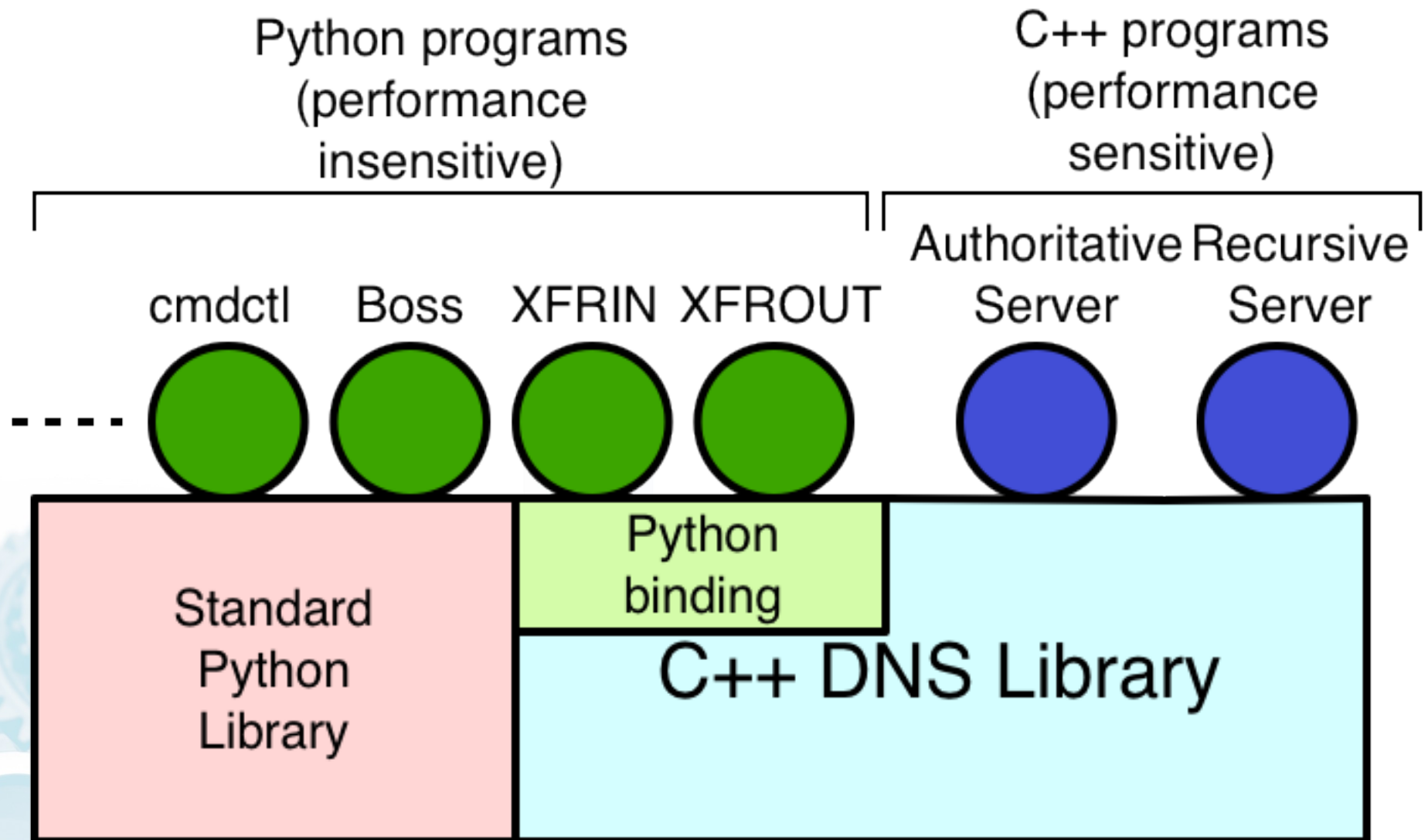
jinmei@isc.org



Sponsors



BIND 10 Software Architecture



C++/python DNS ライブラリ

- DNS のプロトコル要素を扱うフレームワーク
 - ドメイン名、RR type/class, TTL, RRset など
 - 文字列、内部オブジェクト形式、wire format の相互変換
 - DNS Message クラス
 - 受信メッセージの解析、出力メッセージの構築
 - Truncation、compress/decompress

High Level Goals

- 汎用性
 - 依存性を少なく
 - 詳細なドキュメント
- 正確性
 - 徹底したテスト (971 テスト /37 C++ files)
 - 高いコードカバレッジ (90.6%: 2679/2957 行)
- Python Binding
 - " 高速な dnspython"
 - より書きやすく、かつ十分に速いライブラリ

DNS ライブラリ : 現状

- RR types
 - A, NS, CNAME, SOA, PTR, TXT, MX, AAAA, DNAME, OPT, DS, RRSIG, NSEC, DNSKEY, NSEC3, NSEC3PARAM, "Unknown"(RFC3597)
- Standard query の送受信
 - 含 truncation, compress
- EDNS

DNS ライブラリ : 今後

- TSIG(もうすぐ)
- その他の RR types(徐々に)
- DNSSEC の署名と検証 (今年度中)
- 性能チューニング (そのうち)



Python Binding の性能

- おもちゃの DNS サーバによるベンチマーク
 - ハードコードした“ authors.bind” に答えるだけ
 - 11 TXT RRs (299 bytes)
- BIND 9 queryperf で最大 QPS 測定
 - この MacBook Pro 上のループバックテスト :-)



ベンチマーク対象

- BIND 10 python
- その他のスクリプト系 DNS ライブラリ
 - Perl Net::DNS, dnspython, dnsruby
- C++ のみのバージョン
- (参考) ldns python binding
 - 要ハック、機能限定、かつ少しバグっぽい
- ソースコード
 - <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(Message.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(Message.HEADERFLAG_AA)
            msg.set_rcode(Rcode.NOERROR())
            msg.add_rrset(Message.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);
}
```

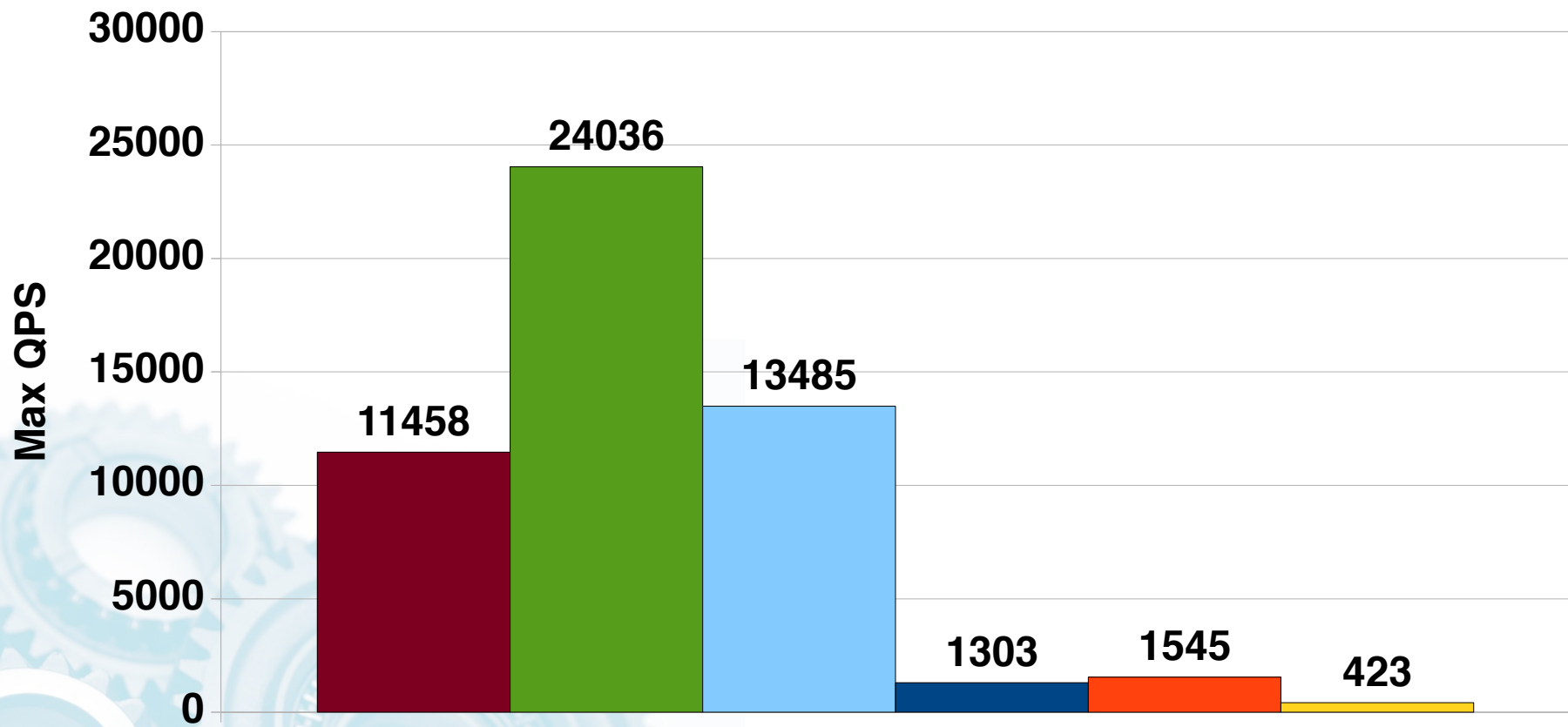
Sample Code: dnsruby

```
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
```

ベンチマーク結果

(注：発表時から一部コードを訂正して結果も更新しています)

■ BIND 10 py ■ BIND 10 C++ ■ Idns-py
■ Net::DNS ■ dnspython ■ dnscruby



ご意見募集中

- BIND 10 サーベイ
 - 現状の BIND に対する要望、不満
 - BIND 10 で実現してほしい機能など
 - <http://www.isc.org/announcement/give-your-input-future-bind>
 - または www.isc.org 内のリンクから
- または jinmei@isc.org にメールで
 - 日本語 OK:-)