Marcus.Grando@corp.terra.com.br
Rodrigo.Broilo@corp.terra.com.br

Implementando DNS Anycast para uso em cache distribuido no Terra

Agenda

- Motivação?
- Rede
- Sistemas
- Scripts e RTT
- Resultados

Motivação

- Alta Disponibilidade.
- Experiência do usuário.
- Porque é legal:
  - Tecnologia nova.
  - Fazer algo diferente.
  - Dificuldade.

Rede

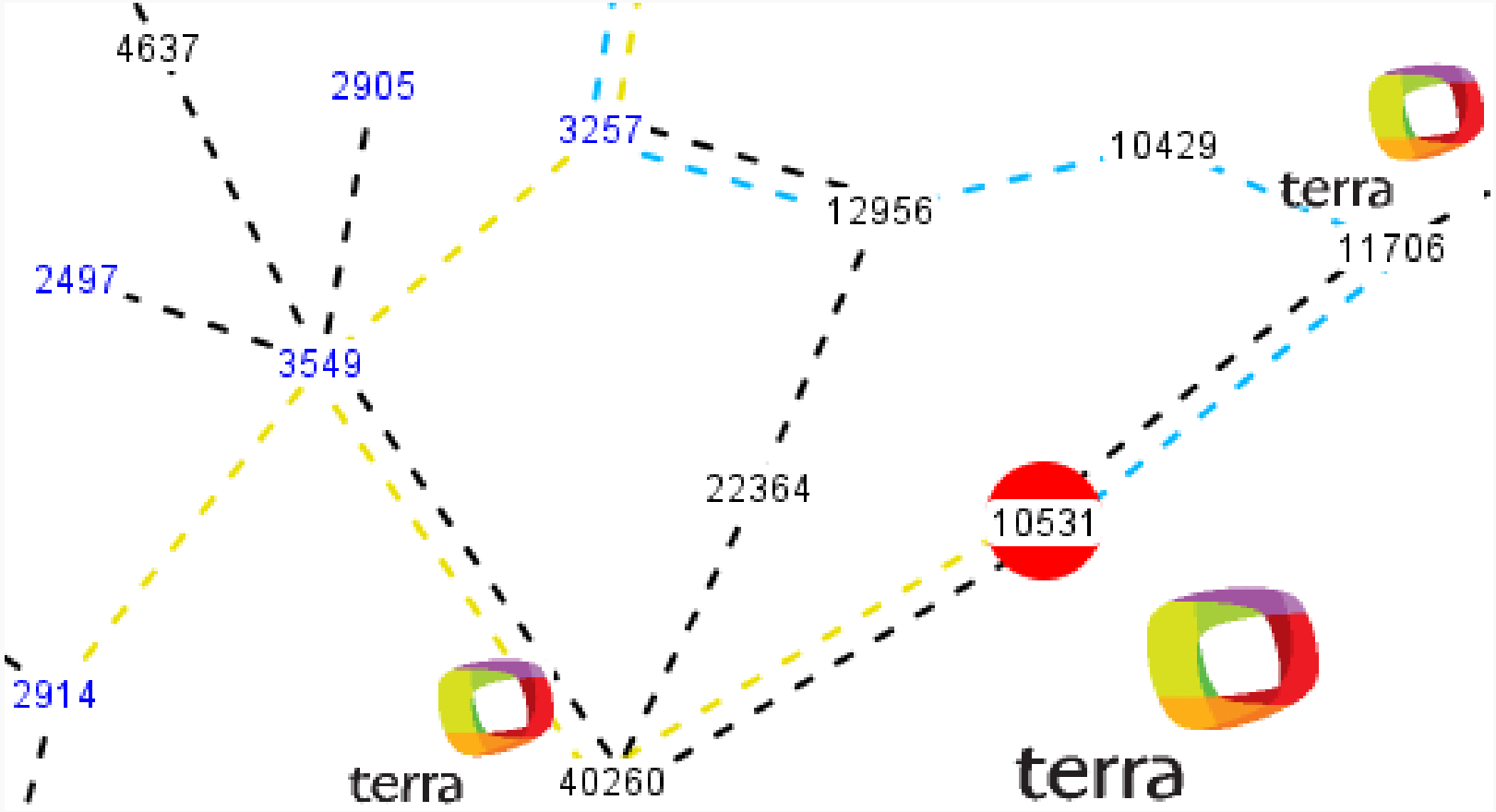
- Routers
  - Adequação Router Reflectors
  - Adequação dos Routers de Borda
  - Liberação Peering e Trânsito
- BGP (Cisco 6K)
  - Keep-alive: 4s
  - Hold interval: 20s
  - Community “anycast”

Rede

- OSPF ou IS-IS possuem convergência melhor, porque não utilizá-los?
- ASN 10531
  - 200.215.192.0/20
    - 200.215.193.0/24 (DNS Primário)
    - 200.215.194.0/24 (DNS Secundário)
    - 200.215.195.0/24 (DNS GSLB)

ASN

- **Trânsito via ASN 11706 (BR) e ASN 40260 (US):**



ASN

- **ASN Prepend?**
- **Anúncio com no-export específico a algum carrier?**

Sistemas

- **Centos 5.6.**
- **BGP no servidor com Quagga . Bird parece promissor.**
- **DNS server Bind. Pode ser utilizado outros também.**

- **Configuração Quagga:**
  - **zebra.conf:**

```
lo:193_1 (200.215.193.1)
lo:194_1 (200.215.194.1)
ip route 200.215.192.0/20 Null0
ip forwarding
```

  - **bgpd.conf:**

```
router bgp 10531
      bgp router-id X.X.X.X
      network 200.215.192.0/20
      network 200.215.193.0/24
      network 200.215.194.0/24
      redistribute connected
      timers bgp 4 20
ip prefix-list ANYCAST-IN
ip prefix-list ANYCAST-OUT
```

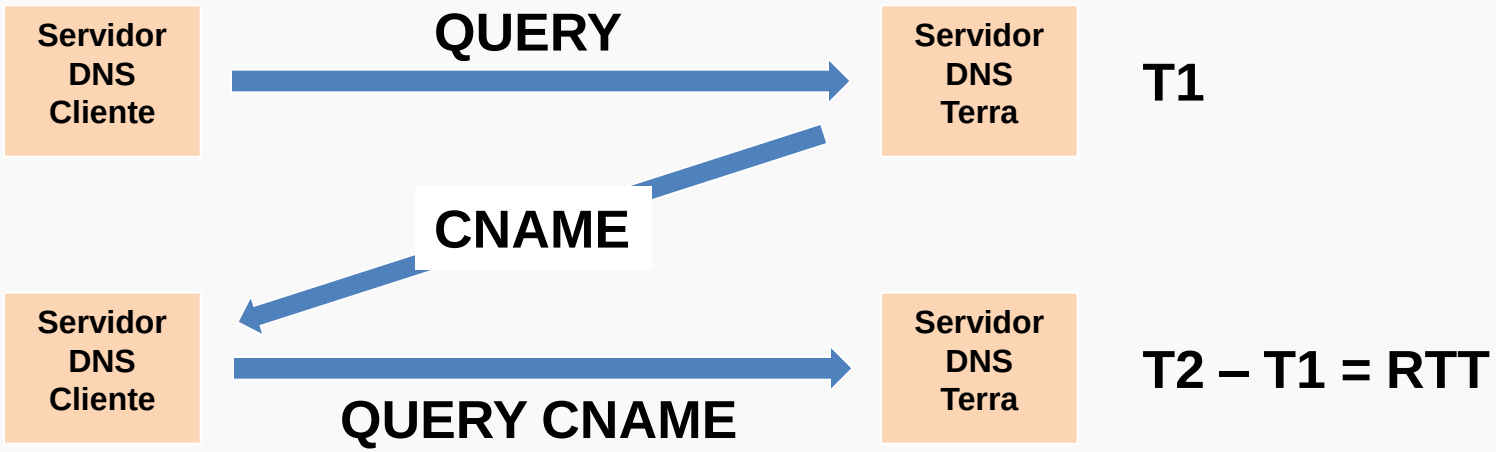
Sistemas

- **Scripts para verificação de funcionamento:**

```
while true:
    if zebra is down:
        startup zebra
    if loopback ips are down
        exit
    if bind is down:
        startup bind
    if dig @ips_anycast hostname.bind != hostname:
        exit
    if bgpd is down:
        startup bgpd
    if bgp sessions and announce are not ok:
        exit
```

RTT

- **Dnscap, tcpdump ou ncap?**
- **Cálculo do RTT:**



QUERY: 1305165969.337861 189.59.5.162 p2.trrsf.com.br
REPLY: 1305165969.337989 189.59.5.162 p2.trrsf.com.br[sdp.trrsf.com.br]
>>> REPLY: 1305165969.337989 189.59.5.162 p2.trrsf.com.br[sdp.trrsf.com.br]
>>> QUERY: 1305165969.339904 189.59.5.162 sdp.trrsf.com.br 0.001915

RTT

- **Script de coleta (python + pcapy + dpkt):**

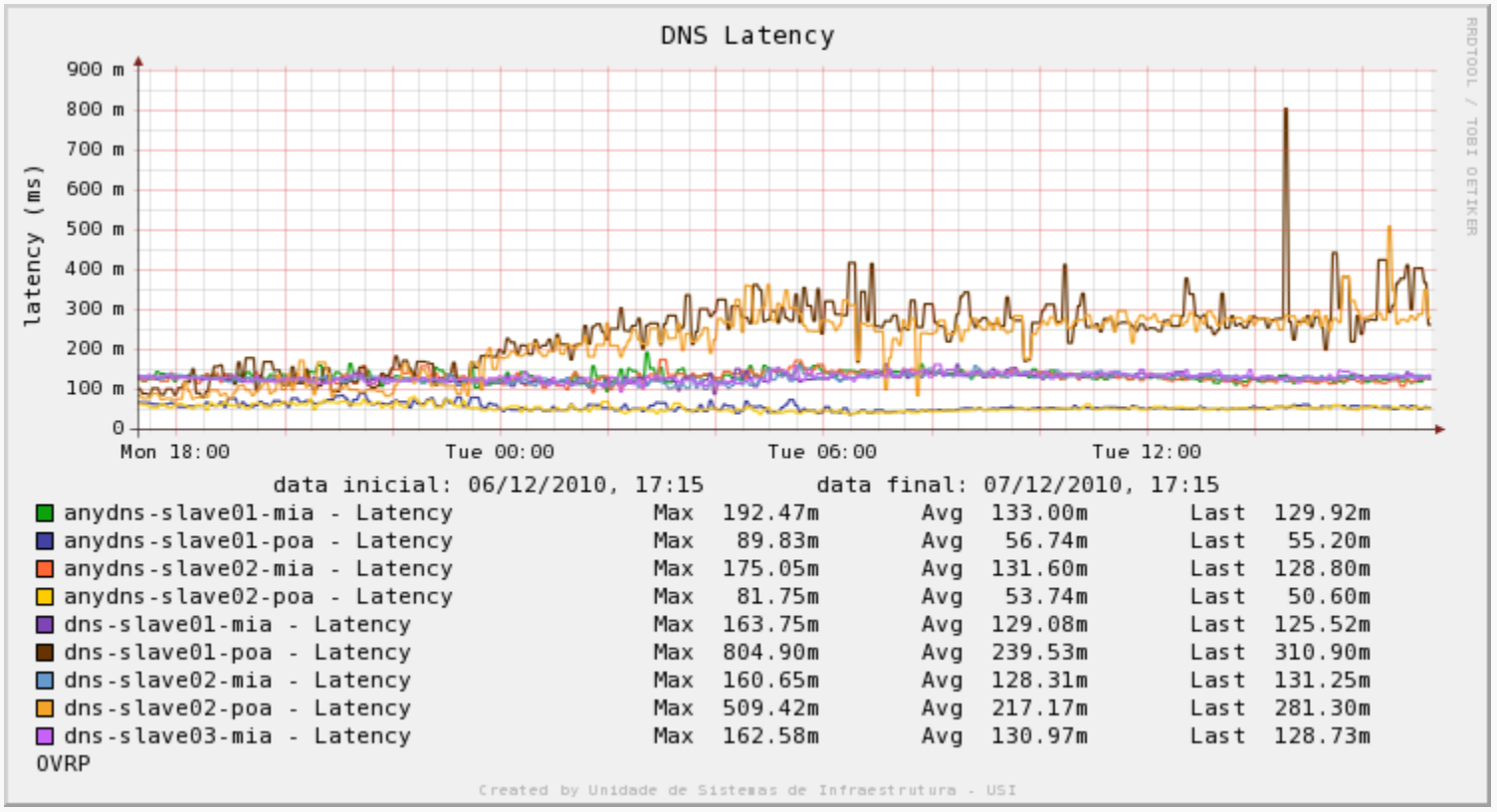
```
#!/usr/bin/python
import dpkt
import pcapy

if ip.data.sport == 53 and len(dns.an) > 1 and dns.an[0].type == 5:
        # REPLY
        tv, usec = hdr.getts()
        t = float("%i.%06i" % (tv, usec))
        replystr = "REPLY: %f %15s %s[%s]" % (t, ipstr, dns.qd[0].name.lower(), dns.an[1].name.lower())
        DATA_LATENCY[ipstr] = {"replytime":t, "expect":dns.an[1].name.lower(), "replystr":replystr}
elif ip.data.dport == 53 and len(dns.qd) > 0:
        # QUERY
        tv, usec = hdr.getts()
        t = float("%i.%06i" % (tv, usec))
        querystr = "QUERY: %f %15s %s" % (t, ipstr, dns.qd[0].name.lower())
        if DATA_LATENCY.has_key(ipstr):
                data_tmp = DATA_LATENCY.pop(ipstr)
                if data_tmp["expect"] == dns.qd[0].name.lower():
                        diff = t - data_tmp["replytime"]
                        VALUES_LATENCY.append(diff)
```

Resultados

- **Alta disponibilidade.**
- **Nenhum usuário afetado.**
- **Facilidade de upgrades.**

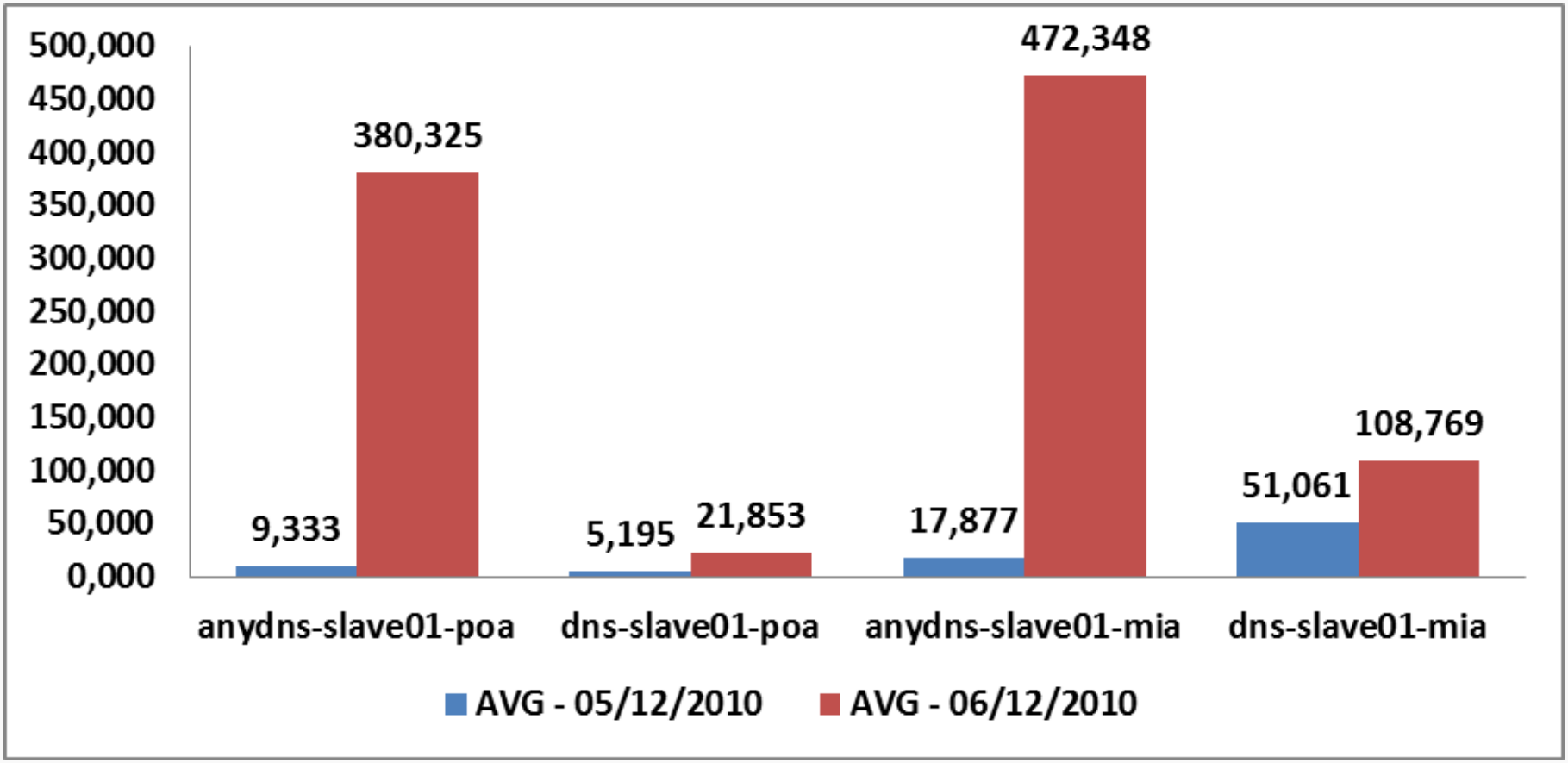
Resultados

- **Tempos de resposta menores:**



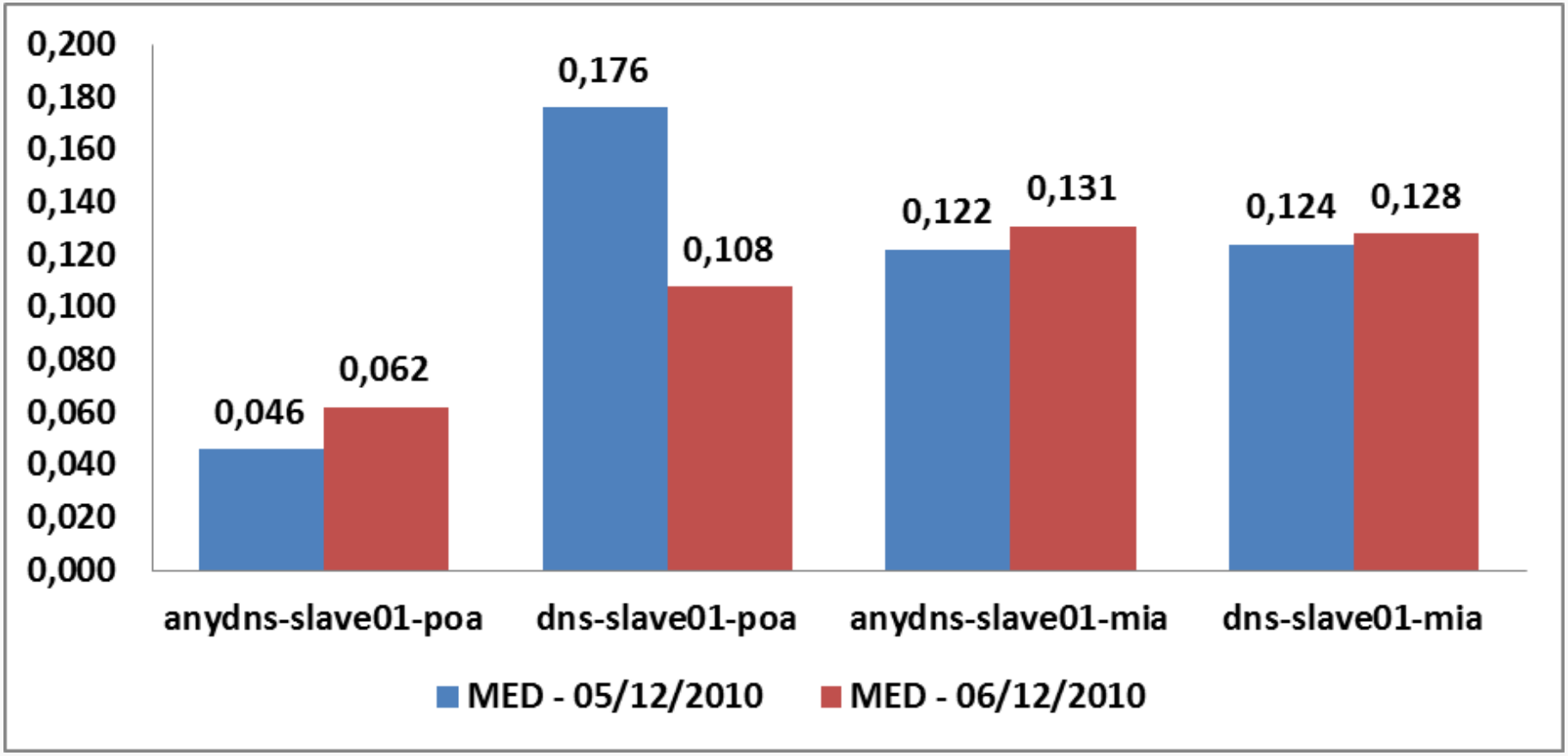
Resultados

- **Média:**



Resultados

- **Mediana:**



OBRIGADO

Marcus.Grando@corp.terra.com.br
Rodrigo.Broilo@corp.terra.com.br