



# Lessons learned from a smooth migration

DDI User Group June 2023  
2023-06-15



### Jens Hoffrichter

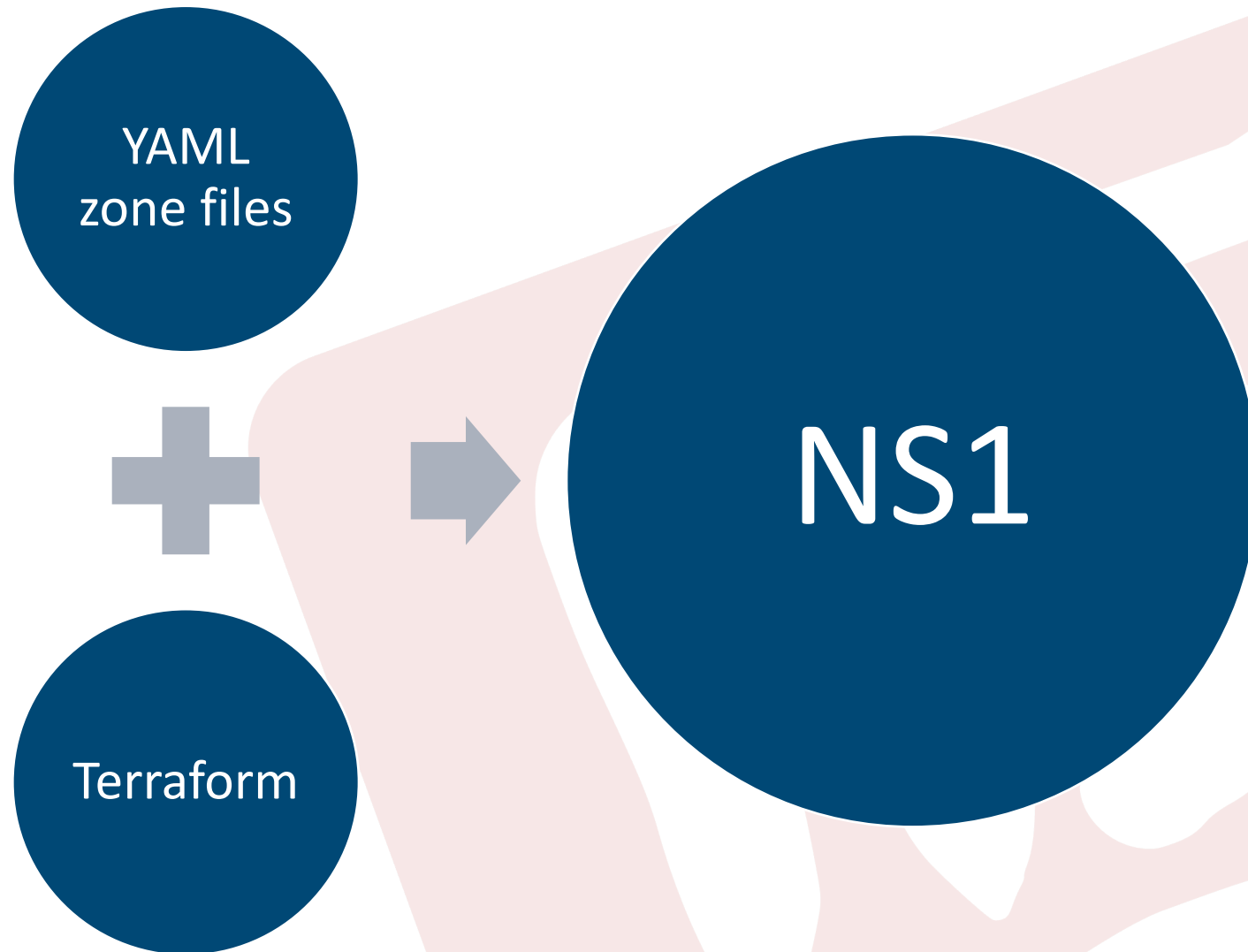
- Dipl.-Ing. (BA) in information technology, specialization in network and media technology
- One of two managing directors of p-square GmbH
- Operations Manager SMTP & DNS for a large company in the automotive sector

### p-square

- Small specialized consulting and operations team for managed services and infrastructure
- Mostly in the automotive and sports management



- Cloud migration of DNS service of the automotive company
- Four authoritative DNS server in two geo locations
- Administration in bind zone files, with git and ansible pushed to servers
- Multi master setup without replication
  
- 5500 domains over a very diverse set of TLDs
- 1000 zone configurations
- 25.000 resource records
- 100 high profile zones
- 20 high profile, large number of records, high change frequency zones
- 100 records with geo dependent answers



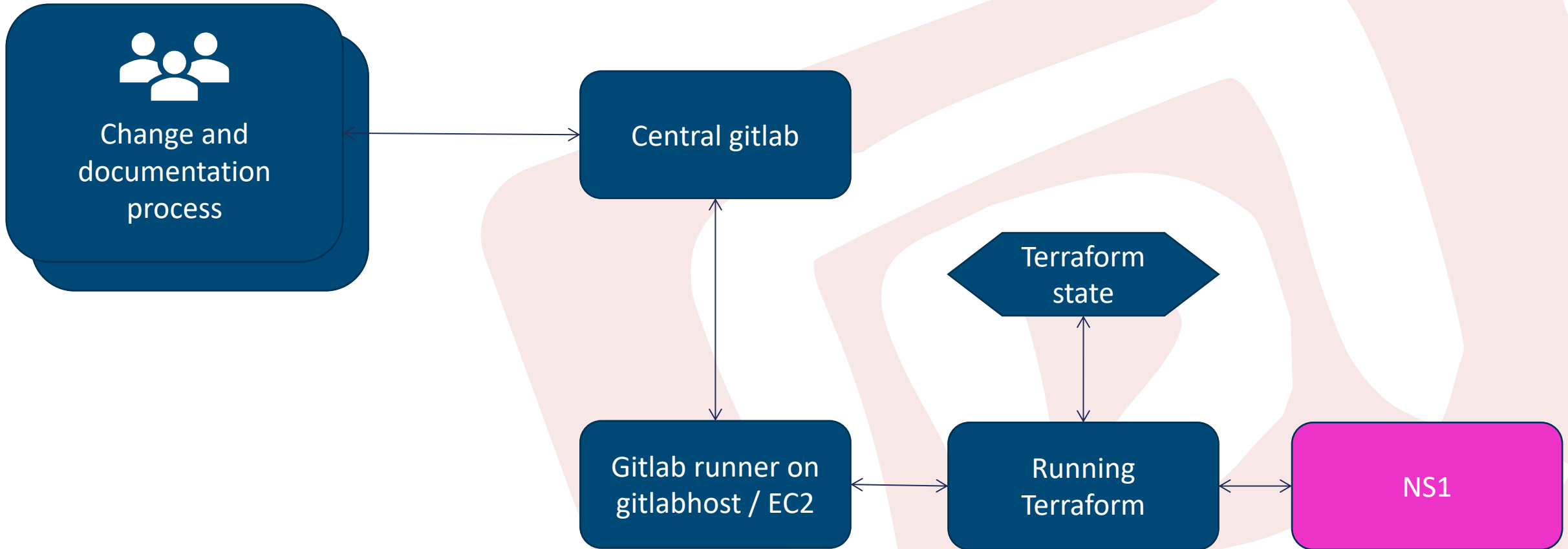


## Advantages

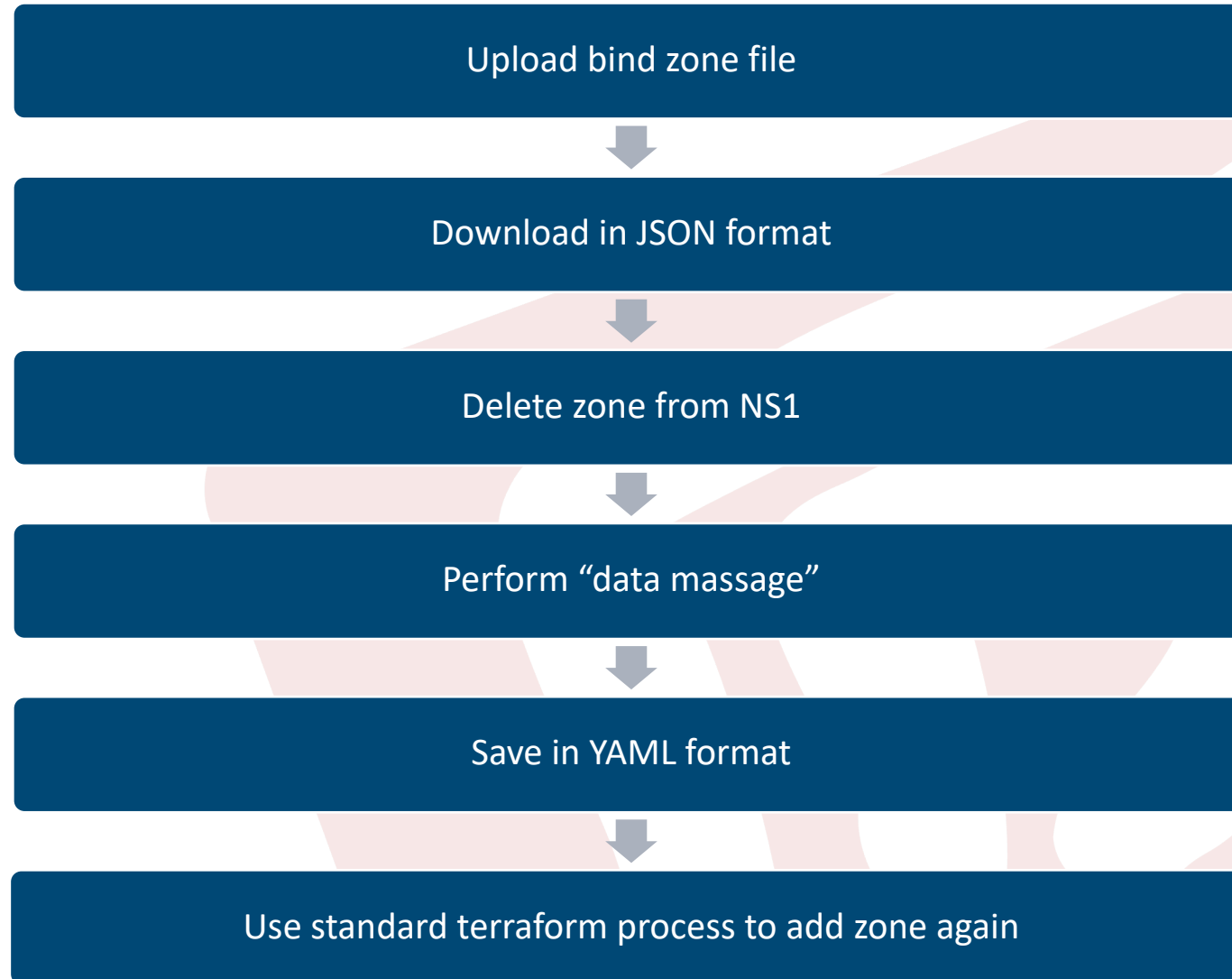
- Standard tool
- Thriving ecosystem
- Knowledge in the team
- Describe the wanted state, terraform makes it how it should be
- Good integration in the automation tool at the customer

## Disadvantages

- Extra layer of complexity
- Need support from the vendor
- Needs to refresh the state regularly
- Does not care about anything not defined in the plan / state











Early in the migration

Migrate batches of 5-200 domains (all TLDs with bulk updates of delegation)



Change NS records to NS1 DNS server records (dns[1-4].p01.nsone.net)



Change delegation at domain vendor

Bulk migration

(Soft) Change freeze for DNS changes



Convert large batches of zones at once (~500)



Change original DNS server records to point to NS1 servers



```
automotivednstest.de:
  dnssec: false
  ttl: 60
  ns: ns1
  zone:
  - automotivednstest.de
  records:
  - label: ''
    type: A
    answers:
    - 169.168.2.15
```



```
resource "ns1_zone" "automotivednstest_de" {
  zone = "automotivednstest.de"
  ttl = 60
  dnssec = false
}
resource "ns1_record" "ns1___A_automotivednstest_de" {
  zone = "automotivednstest.de"
  domain = "automotivednstest.de"
  type = "A"
  answers {
    answer = " 169.168.2.15"
  }
  depends_on = [ ns1_zone.automotivednstest_de ]
}
```



- Stable resource names without collisions
- Structure of the terraform provider leads to a very big state file
- State refreshes are not possible in real time -> Triggered task at night
- YAML format not as easily human readable as a zone file



- 4 DNS servers under dns.automotive.(de|com|ch|asia)
- Moved the zones automotive.(de|com|ch|asia) to DNS names of NS1
- Changed delegation, so no more glue records necessary
- Left most of the other domains under the old DNS server names
- Updated A records in the zone for the DNS servers
  
- After a couple of days after the main migration, there were still queries towards the legacy DNS server
  
- Verisign DNS server handed out dns.automotive.(de|com|ch|asia) as NS set, with IP hint for dns.automotive.com towards the old IP address
- Orphaned glue record set at Verisign, admin of our domain vendor needed to log in manually on the website and delete those



- A working cloud dns solution, with version control in gitlab, automated rollout via CI/CD and a lot of scripting possibilities
- Stable connection via terraform to NS1
- No downtimes during migration
- Change freeze for about 2 days
- Only a couple of hiccups for some domains, where new changes were not seen everywhere
- Generally, very satisfied customer :D
- Few improvement possibilities, although not high priority right now



- A good support from the vendor is worth its weight in gold
- Not all problems can be found with test data
- Automate as much as you can, to remove the possibility of human error
  
- Would I go for terraform again? – Probably yes, but structure the project a bit differently to have more flexibility

Thank you!



## Jens Hoffrichter



**p-square GmbH**

[jens.hoffrichter@p-square.de](mailto:jens.hoffrichter@p-square.de)

<https://p-square.de/>