# PADS/ML

## A Functional Data Description Language

Yitzhak Mandelbaum
Princeton University, AT&T

With: Kathleen Fisher, David Walker,
Mary Fernandez, Artem Gleyzer

# Meet Stan

Stan works for a large networking company, managing their routers.

Every day, Stan sends and receives commands to and from his routers.

# Meet Stan

Stan works for a large networking company, managing their routers.

Every day, Stan sends and receives commands to and from his routers.

```
Router(config)# ip cef distributed
Router(config)# tag-switching tsp-tunnels
Router(config)# interface e0/1
Router(config-if)# tag-switching tsp-tunnels
Router(config-if)# interface e0/2
Router(config-if)# tag-switching tsp-tunnels
Router(config-if)# exit

Router(config)# interface tunnel 2003
Router(config-if)# tunnel mode tag-switching
Router(config-if)# tunnel tsp-hop 1
10.10.0.12
Router(config-if)# tunnel tsp-hop 2
10.50.0.24 lasthop
Router(config-if)# exit

Router(config)# interface tunnel 2003
Router(config-if)# no tunnel tsp-hop 2
Router(config-if)# tunnel tsp-hop 1
10.10.0.12 last
Router(config-if)# exit
```

# Stan's New Hope

- One day, Stan decides to automate!
- Stan will request entire config from router and process it externally.
- But router configuration files can contain 10s of thousands of commands each; stored in a complex and evolving, ad-hoc format.

# Meet Stan's Configs

```
Current configuration : 1682365 bytes
version 12.0
no service pad
service tcp-keepalives-in
service password-encryption
service compress-config
service upgrade all
!
hostname fromage
!
redundancy
 mode rpr-plus
logging buffered 2000000 debugging
logging console informational
!
username wallace password 5 1240AF334003
username grommit password 5 1A644AC55FF7
!
hw-module slot 1 provision dynamic
!
!
ip subnet-zero
ip name-server 12.127.17.83
ip flow-cache timeout active 1
ip cef table hardware resource-failure
action punt
no ip finger
```

```
ip vrf 8904
 description MPLS-PNT| BOOTKEY| LES-MPLS|
 route-target export 123413:1492
 maximum routes 220 40
!
ip vrf 1023
 description ANTI-PESTO S.W.A.T. TEAM|
 export map To_NY_VPN
 route-target 100:3
 maximum routes 150 80
!
ip receive access-list 2660
!
!
policy-map mis_policy_90:0:0:100_output_12288K
  class rt_class
    priority
   police cir percent 90 conform-action transmit
exceed-action drop
  class class-default
    bandwidth remaining percent 100
    queue-limit 500 ms
...
```

# Meet Stan's Configs

```
Current configuration : 1682365 bytes
version 12.0
no service pad
service tcp-keepalive
service password-encr
service compress-conf
service upgrade all
!
hostname fromage
!
redundancy
 mode rpr-plus
logging buffered 2000000 debugging
logging console informational
!
username wallace password 5 1240AF334003
username grommit password 5 1A644AC55FF7
!
hw-module slot 1 provision dynamic
!
!
ip subnet-zero
ip name-server 12.127.17.83
ip flow-cache timeout active 1
ip cef table hardware resource-failure
action punt
no ip finger
```

```
ip vrf 8904
```


```
username wallace password 5 1240AF334003
username gromit password 5 1A644AC55FF7
!
```


```
 description ANTI-PESTO S.W.A.T. TEAM|
 export map To_NY_VPN
 route-target 100:3
 maximum routes 150 80
!
ip receive access-list 2660
!
!
policy-map mis_policy_90:0:0:100_output_12288K
  class rt_class
    priority
   police cir percent 90 conform-action transmit
exceed-action drop
  class class-default
    bandwidth remaining percent 100
    queue-limit 500 ms
...
```

# Meet Stan's Configs

```
Current configuration : 1682365 bytes
version 12.0
no service pad
service tcp-keepalives-in
service password-encryption
service compress-config
service upgrade all
!
hostname fromage
!
redundancy
 mode rpr-plus
logging buffered 2000000 debugging
logging console informational
!
username wallace password 5 1240AF334003
username grommit password 5 1A644AC55FF7
!
hw-module slot 1 provision dynamic
!
!
ip subnet-zero
ip name-server 12.127.17.83
ip flow-cache timeout active 1
ip cef table hardware resource-failure
action punt
no ip finger
```

```
ip vrf 8904
 description MPLS-PNT| BOOTKEY| LES-MPLS|
 route-target export 123413:1492
 maximum routes 220 40
!
ip vrf 1023
 description ANTI-PESTO S.W.A.T. TEAM|
 export map To_NY_VPN
 route-target 100:3
 maximum routes 150 80
!
ip receive access-list 2660
!
!
policy-map mis_policy_90:0:0:100_output_12288K
  class rt_class
    priority
   police cir percent 90 conform-action transmit
exceed-action drop
  class class-default
    bandwidth remaining percent 100
    queue-limit 500 ms
...
```

# Meet Stan's Configs

```
Current configuration : 1682365 bytes
version 12.0
no service pad
service tcp-keepalives-in
service password-encryption
service compress-config
service upgrade all
!
hostname fromage
!
redundancy
 mode rpr-plus
logging buffered 2000000 debugging
lo
!
us
us
!
hw
!
!
ip
ip name-server 12.127.17.83
ip flow-cache timeout active 1
ip cef table hardware resource-failure
action punt
no ip finger
```

```
ip vrf 8904
 description MPLS-PNT| BOOTKEY| LES-MPLS|
 route-target export 123413:1492
 maximum routes 220 40
!
ip vrf 1023
 description ANTI-PESTO S.W.A.T. TEAM|
 export map To_NY_VPN
 route-target 100:3
 maximum routes 150 80
!
```

```
ip vrf 1023
 description ANTI-PESTO S.W.A.T. TEAM|
 export map To_NY_VPN
 route-target 100:3
 maximum routes 150 80
```

```
3K
smit
bandwidth remaining percent 100
queue-limit 500 ms
...
```

# Stan's World

| Goals | Requirements |
| --- | --- |
| | Parsing |
| Audit | Analysis |
| Communicate | XML Conversion |
| Configure | Transformation |
| | Printing |

# Stan's World

| Goals | Requirements |
| --- | --- |
| | Parsing |
| Audit | Analysis |
| Communicate | XML Conversion |
| Configure | Transformation |
| | Printing |

# Stan's World

| Goals | Requirements |
| --- | --- |
| | Parsing |
| Audit | Analysis |
| Communicate | XML Conversion |
| Configure | Transformation |
| | Printing |

# Stan's World

| Goals | Requirements |
| --- | --- |
| | Parsing |
| Audit | Analysis |
| Communicate | XML Conversion |
| Configure | Transformation |
| | Printing |

# Stan's World

| Goals | Requirements |
| --- | --- |
| | Parsing ← |
| Audit | Analysis |
| → Communicate | XML Conversion ← |
| Configure | Transformation |
| | Printing |

# Stan's World

| Goals | Requirements |
| --- | --- |
| | Parsing |
| Audit | Analysis |
| Communicate | XML Conversion |
| Configure | Transformation |
| | Printing |

# Stan's World

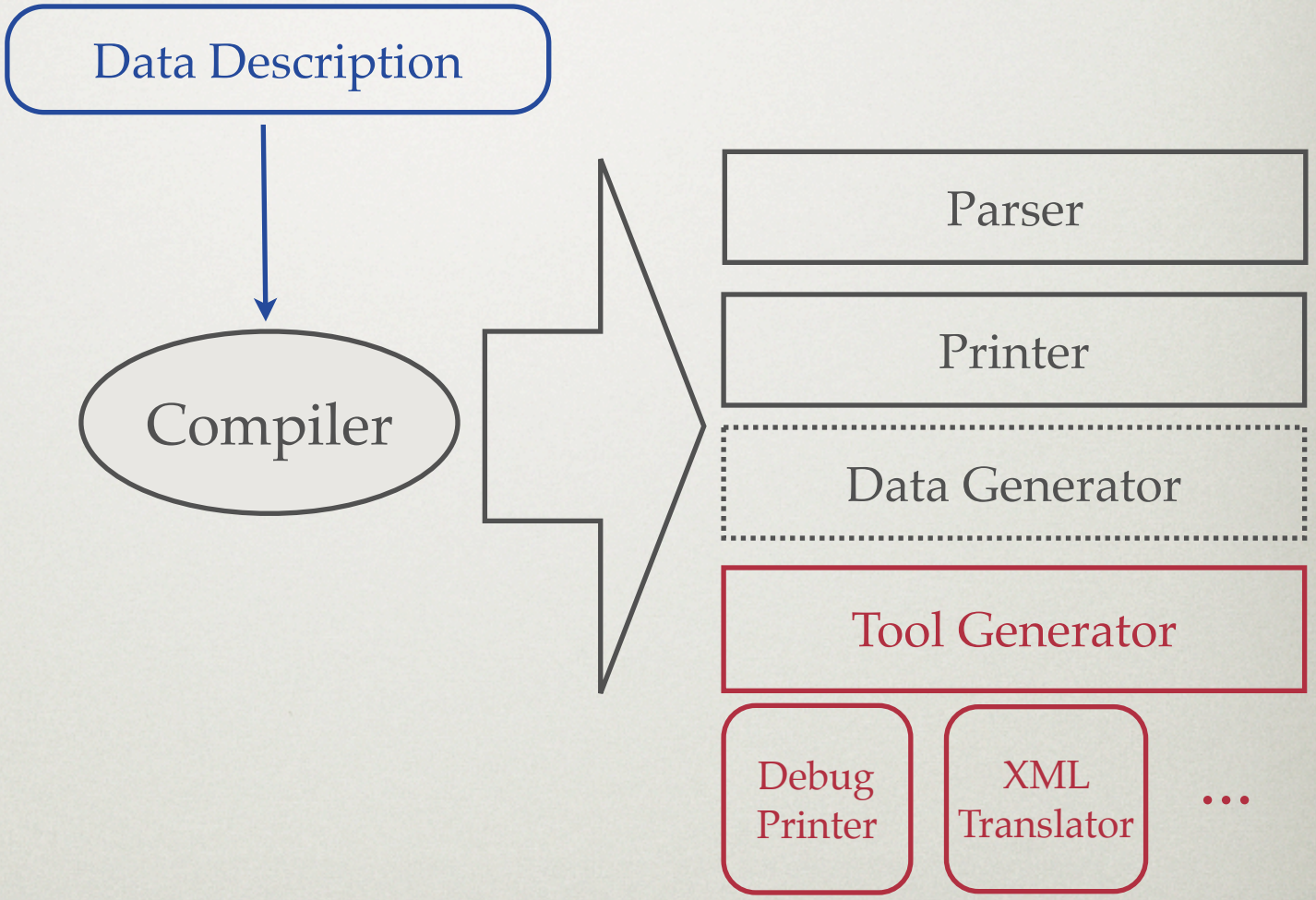
| Goals | Requirements |
| --- | --- |
| | Parsing |
| Audit | Analysis |
| Communicate | XML Conversion |
| Configure | Transformation |
| | Printing |

# Enter: PADS/ML

Description Language

Compiler

Generic Tool Development Framework



# Based on ML Types

We can describe data like we describe data structures.

# Based on ML Types

We can describe data like we describe data structures.

Base types describe atomic portions of data

| Data | PADS/ML |
|---|---|
| wallace | pstring_ws |
| 53.5637 | pfloat |
| 100 | pint_FW(3) |

# Based on ML Types

We can describe data like we describe data structures.

Base types describe atomic portions of data

| Data | PADS/ML |
|---|---|
| wallace | pstring_ws |
| 53.5637 | pfloat |
| 100 | pint_FW(3) |

Familiar type constructors describe complex data:

tuples, records, datatypes, ...

```
ip vrf 1023
 description ANTI-PESTO S.W.A.T. TEAM|
 export map To_NY_VPN
 route-target 100:3
 max routes 150 80
```

```
ptype ip_vrf_command =
  Description  of “description ”  * pstring(‘|’) * ‘|’
| Export       of “export map ”   * pstring(‘\n’)
| Route_target of “route-target ” * pint * ‘:’ * pint
| Max_routes   of “max routes “   * pint * ‘ ’ * pint

ptype ip_vrf = {
  header : “ip vrf “ * pint * ‘\n’;
  commands : ip_vrf_command plist(‘\n’)
}
```

```
ip vrf 1023
 description ANTI-PESTO S.W.A.T. TEAM|
 export map To_NY_VPN
 route-target 100:3
 max routes 150 80
```

```
ptype ip_vrf_command =
  Description  of "description "  * pstring('|') * '|'
| Export       of "export map "   * pstring('\n')
| Route_target of "route-target " * pint * ':' * pint
| Max_routes   of "max routes "   * pint * ' ' * pint

ptype ip_vrf = {
  header : "ip vrf " * pint * '\n';
  commands : ip_vrf_command plist('\n')
}
```

```
ip vrf 1023
 description ANTI-PESTO S.W.A.T. TEAM|
 export map To_NY_VPN
 route-target 100:3
 max routes 150 80
```

```
ptype ip_vrf_command =
  Description  of "description "  * pstring('|') * '|'
| Export       of "export map "   * pstring('\n')
| Route_target of "route-target " * pint * ':' * pint
| Max_routes   of "max routes "   * pint * ' ' * pint

ptype ip_vrf = {
  header : "ip vrf " * pint * '\n';
  commands : ip_vrf_command plist('\n')
}
```

```
ip vrf 1023
  description ANTI-PESTO S.W.A.T. TEAM|
  export map To_NY_VPN
  route-target 100:3
  max routes 150 80
```

```
ptype ip_vrf_command =
  Description  of "description "  * pstring('|') * '|'
| Export       of "export map "   * pstring('\n')
| Route_target of "route-target " * pint * ':' * pint
| Max_routes   of "max routes "   * pint * ' ' * pint

ptype ip_vrf = {
  header : "ip vrf " * pint * '\n';
  commands : ip_vrf_command plist('\n')
}
```

```
ip vrf 1023
 description ANTI-PESTO S.W.A.T. TEAM|
 export map To_NY_VPN
 route-target 100:3
 max routes 150 80
```

```
ptype ip_vrf_command =
  Description  of "description "  * pstring('|') * '|'
| Export       of "export map "   * pstring('\n')
| Route_target of "route-target " * pint * ':' * pint
| Max_routes   of "max routes "   * pint * ' ' * pint

ptype ip_vrf = {
  header : "ip vrf " * pint * '\n';
  commands : ip_vrf_command plist('\n')
}
```

# Language Features

Records for data dependency

Constraints for additional properties

Datatypes for variation

Matches for tagged data

Recursion for nested structure

Polymorphism for code reuse

Lists, speculation, early commit, and more ...

# Types as Modules

Compiler converts PADS/ML types into O'Caml modules.

| PADS/ML | O'Caml |
|---|---|
| Monomorphic type | Module |
| Recursive type | Recursive Module |
| Polymorphic type | Functor (module → ...) |

# Generic Tool Framework

Developer writes tool in type-directed manner.

Compiler generates a single-pass traversal.

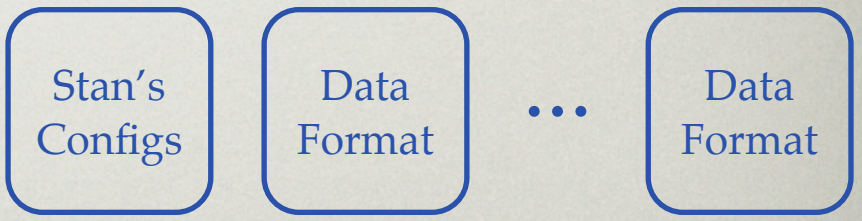
User easily applies any tool to any data source.

# Generic Tool Framework

Developer writes tool in type-directed manner.

Compiler generates a single-pass traversal.

User easily applies any tool to any data source.

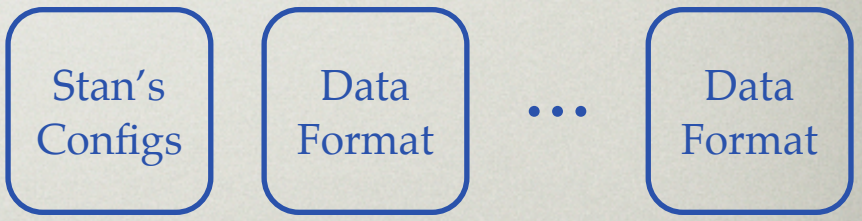


# Generic Tool Framework

Developer writes tool in type-directed manner.


Compiler generates a single-pass traversal.

User easily applies any tool to any data source.

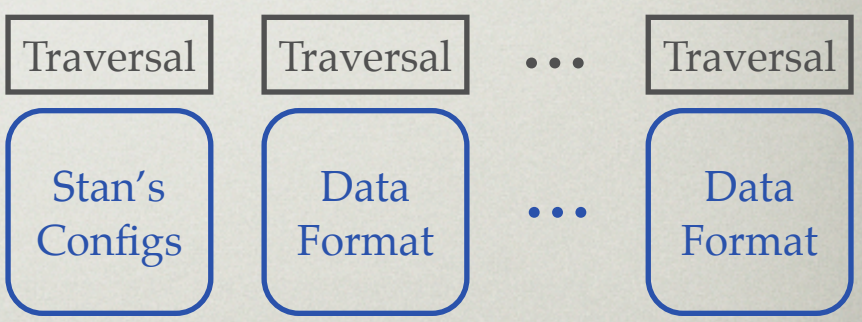


# Generic Tool Framework

Developer writes tool in type-directed manner.


Compiler generates a single-pass traversal.

User easily applies any tool to any data source.

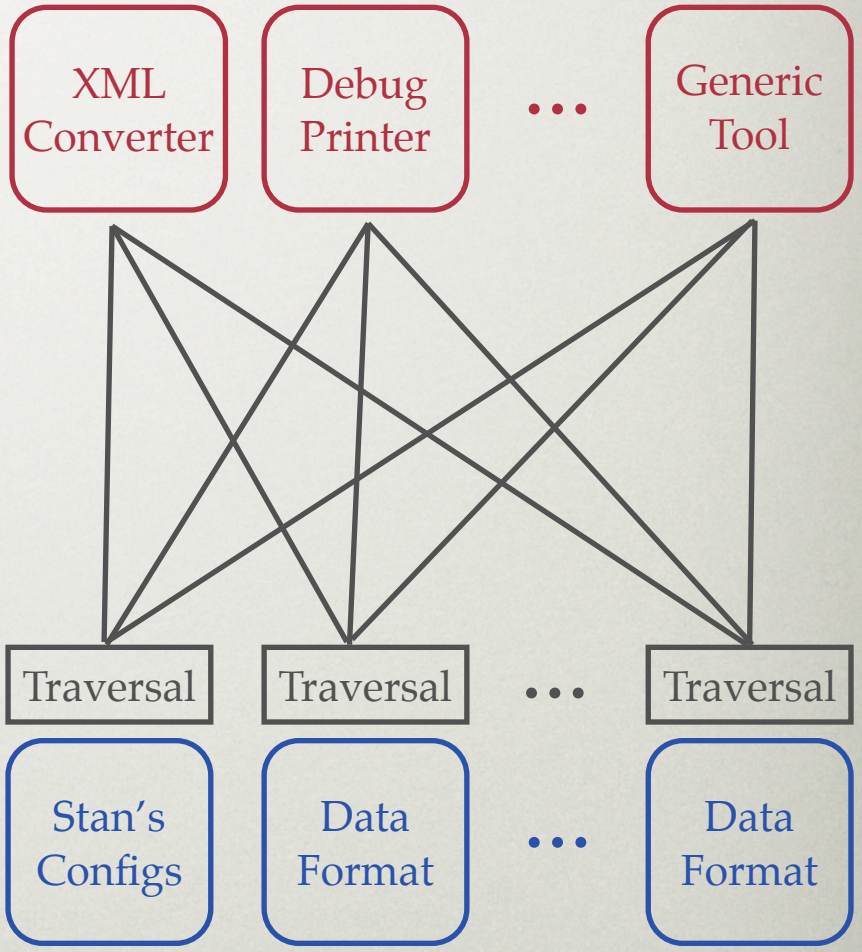


# Generic Tool Framework

Developer writes tool in type-directed manner.

Compiler generates a single-pass traversal.

User easily applies any tool to any data source.



# Summary

- A functional description language for data analysts.
- A practical framework for rapidly developing new data processing tools.
- Based on solid theoretical foundation.
  - DDC + polymorphic types [POPL '06, Thesis].

# Describes real formats

- Stan's config files
- Newick tree-structured data
- AT&T billing data
- Web server logs
- ...

# Download

www.padsproj.org

# The End