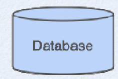
Typing AD Hoc Data

Kathleen Fisher
AT&T Labs Research

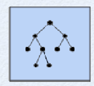
Data, Data, everywherE!

ble amounts of data stored in well-behaved f

Databases:



XML:



Tools

- Schema
- Browsers
- Query Languages
- Standards
- Libraries
- Books, documentation
- Training courses
- Conversion tools
- Vendor support
- Consultants...

We're not always so lucky!

Vast amounts of chaotic *ad hoc* data:

Tools

- Perl
- Awk
- C
- ...

Government stats

```
"MSN","YYYYMM","Publication Value","Publication Unit","Column Order"
"TEAJBUS",197313,-0.456483,Quadrillion Btu,4
"TEAJBUS",197413,-0.482265,Quadrillion Btu,4
"TEAJBUS",197513,-1.066511,Quadrillion Btu,4
"TEAJBUS",197613,-0.177807,Quadrillion Btu,4
"TEAJBUS",197713,-1.948233,Quadrillion Btu,4
"TEAJBUS",197813,-0.336538,Quadrillion Btu,4
"TEAJBUS",197913,-1.649302,Quadrillion Btu,4
"TEAJBUS",198013,-1.0537,Quadrillion Btu,4
```

Train Stations

```
Southern California Regional Railroad Authority,"Los Angeles, CA",
U,45,46,46,47,49,51,U,45,46,46,47,49,51
Connecticut Department of Transportation ,"New Haven, CT",
U,U,U,U,U,U,8,U,U,U,U,U,U,8
Tri-County Commuter Rail Authority ,"Miami, FL",
U,U,U,U,U,U,18,U,U,U,U,U,U,18
Northeast Illinois Regional Commuter Railroad Corporation,"Chicago,
IL",226,226,226,227,227,227,227,91,104,104,111,115,125,131
Northern Indiana Commuter Transportation District,"Chicago,
IL",18,18,18,18,18,18,20,7,7,7,7,7,7,11
Massachusetts Bay Transportation Authority,"Boston, MA",
U,U,117,119,120,121,124,U,U,67,69,74,75,78
Mass Transit Administration - Maryland DOT ,"Baltimore, MD",
U,U,U,U,U,U,42,U,U,U,U,U,U,22
New Jersey Transit Corporation ,"New York,
NY",158,158,158,162,162,162,167,22,22,41,46,46,46,51
```

Web logs

```
207.136.97.49 - - [15/Oct/2006:18:46:51 -0700] "GET /turkey/amnty1.gif HTTP/1.0" 200 3013
207.136.97.49 - - [15/Oct/2006:18:46:51 -0700] "GET /turkey/clear.gif HTTP/1.0" 200 76
207.136.97.49 - - [15/Oct/2006:18:46:52 -0700] "GET /turkey/back.gif HTTP/1.0" 200 224
207.136.97.49 - - [15/Oct/2006:18:46:52 -0700] "GET /turkey/women.html HTTP/1.0" 200 17534
208.196.124.26 - Dbuser [15/Oct/2006:18:46:55 -0700] "GET /candatop.html HTTP/1.0" 200 -
208.196.124.26 - - [15/Oct/2006:18:46:57 -0700] "GET /images/done.gif HTTP/1.0" 200 4785
www.att.com - - [15/Oct/2006:18:47:01 -0700] "GET /images/reddash2.gif HTTP/1.0" 200 237
208.196.124.26 - - [15/Oct/2006:18:47:02 -0700] "POST /images/refrun1.gif HTTP/1.0" 200 836
208.196.124.26 - - [15/Oct/2006:18:47:05 -0700] "GET /images/hasene2.gif HTTP/1.0" 200 8833
www.cnn.com - - [15/Oct/2006:18:47:08 -0700] "GET /images/candalog.gif HTTP/1.0" 200 -
208.196.124.26 - - [15/Oct/2006:18:47:09 -0700] "GET /images/nigpost1.gif HTTP/1.0" 200 4429
208.196.124.26 - - [15/Oct/2006:18:47:09 -0700] "GET /images/rally4.jpg HTTP/1.0" 200 7352
128.200.68.71 - - [15/Oct/2006:18:47:11 -0700] "GET /amnesty/usalinks.html HTTP/1.0" 143 10329
208.196.124.26 - - [15/Oct/2006:18:47:11 -0700] "GET /images/reyes.gif HTTP/1.0" 200 10859
```

Genetic data

```
((raccoon:19.19959,bear:6.80041):0.84600,((sea_lion:11.997
00,seal:12.00300):7.52973,((monkey:100.85930,cat:47.14069)
:20.59201,weasel:18.87953):2.09460):3.87382,dog:25.46154);
(Bovine:0.69395,(Gibbon:0.36079,(Orang:0.33636,(Gorilla:0.
17147,(Chimp:0.19268,Human:0.11927):0.08386):0.06124):0.15
057):0.54939,Mouse:1.21460):0.10;
(Bovine:0.69395,(Hylobates:0.36079,(Pongo:0.33636,(G._Gori
lla:0.17147,(P._paniscus:0.19268,H._sapiens:0.11927):0.083
86):0.06124):0.15057):0.54939,Rodent:1.21460);
```

Haskell HI files

```
00000000: 0001 face 0000 0073 0400 0000 3600 0000  .......s....6...
00000010: 3000 0000 3500 0000 3000 0000 0000 0000  0...5...0.......
00000020: 0001 0000 0000 0100 0000 0043 0001 0000  ...........C....
00000030: 0002 0200 0000 0200 0000 0300 0000 0200  ................
00000040: 0000 0400 0000 4800 0100 0000 0200 0000  ......H.........
00000050: 0502 0000 0000 0006 0000 0000 0007 0000  ................
00000060: 0001 0000 0000 6800 0000 0000 006f 0000  ......h......o..
00000070: 0000 0100 0000 0800 0000 0968 6173 6b65  ...........haske
00000080: 6c6c 3938 0000 0007 4350 5554 696d 6500  ll98....CPUTime.
00000090: 0000 0462 6173 6500 0000 0847 4843 2e42  ...base....GHC.B
000000a0: 6173 6500 0000 0e47 4843 2e46 6f72 6569  ase....GHC.Forei
000000b0: 676e 5074 7200 0000 0e53 7973 7465 6d2e  gnPtr....System.
000000c0: 4350 5554 696d 6500 0000 0a67 6574 4350  CPUTime....getCP
000000d0: 5554 696d 6500 0000 1063 7075 5469 6d65  UTime....cpuTime
000000e0: 5072 6563 6973 696f 6e                   Precision
```

Ad hoc data from AT&T

Name & Use	Representation	Size
Web server logs (CLF): Measure web workloads	Fixed-column ASCII records	≤ 12 GB/week
Sirius data: Monitor service activation	Variable-width ASCII records	2.2GB/week
Call detail: Detect fraud	Fixed-width binary records	~7GB/day
Altair data: Track billing process	Various Cobol data formats	~4000 files/day
Regulus data: Monitor IP network	ASCII	≥ 15 sources, ~15 GB/day
Netflow: Monitor IP network	Data-dependent number of fixed-width binary records	>1Gigabit/second

And many others...

- Gene ontology data
- Cosmology data
- Financial trading data
- Telecom billing data
- Router config files
- System logs
- Call detail data
- Netflow packets
- DNS packets
- Java JAR files
- Jazz recording info
- ...

Technical challenges

- Data arrives "as is."
- Documentation is often out-of-date or nonexistent.
  - Hijacked fields.
  - Undocumented "missing value" representations.
- Data is buggy.
  - Missing data, human error, malfunctioning machines, race conditions on log entries, "extra" data, …
  - Processing must detect *relevant* errors and respond in *application-specific* ways.
  - Errors are sometimes the *most* interesting portion of the data.
- Data sources often have high volume.
  - Data may not fit into main memory.

Existing approaches

- Lex/Yacc
  - No one we have encountered uses them for ad hoc data.
- Perl/C
  - Code brittle with respect to changes in input format.
  - Analysis often ends up interwoven with parsing, precluding reuse.
  - Error code, if written, swamps main-line computation. If not written, errors can corrupt "good" data.
  - Everything has to be coded by hand.
- Data description languages (PacketTypes, Datascript)
  - Binary data

types to the rescue!

Relational and XML data are relatively easy to manage (partly) because schema exist to describe the data.

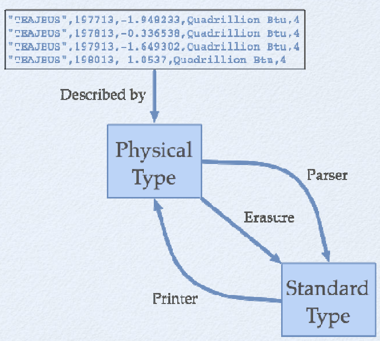
Relational Data	Relational Schema
XML	XML Schema
Ad Hoc Data	***Physical Types***

Thesis: Types can facilitate ad hoc data management, and *the types developed for in-memory values are suited to the task.*

Our approach: PADS

- Data expert writes declarative description of data source:
  - Physical format information
  - Semantic constraints
- Many data consumers use description and generated parser.
- Description serves as living documentation.
- Parser exhaustively detects errors without cluttering user code.
- From declarative specification, we generate auxiliary tools.

Typing ad hoc data



Base types

"TEAJBUS",197313,-0.456483,Quadrillion Btu,4
"TEAJBUS",197413,-0.482265,Quadrillion Btu,4

String, Int, Float

Tuple types

"TEAJBUS",197313,-0.456483,Quadrillion Btu,4
"TEAJBUS",197413,-0.482265,Quadrillion Btu,4

String * Int * Float * String * Int

Singleton types

"TEAJBUS",197313,-0.456483,Quadrillion Btu,4
"TEAJBUS",197413,-0.482265,Quadrillion Btu,4

```
'\"' * String * '\"' * ','
     * Int           * ','
     * Float         * ','
     * String        * ','
     * Int
```

Where we write ',' for S(',').

Simple dependent types

```
"TEAJBUS",197313,-0.456483,Quadrillion Btu,4
"TEAJBUS",197413,-0.482265,Quadrillion Btu,4
```

```
'\"' * String('\"') * '\"' * ','
     * Int                  * ','
     * Float                * ','
     * String(',')          * ','
     * Int
```

Records

```
"TEAJBUS",197313,-0.456483,Quadrillion Btu,4
"TEAJBUS",197413,-0.482265,Quadrillion Btu,4
```

```
{                                  "\"",
  source: String('\"'),            "\",",
  date:   Int,                     ",",
  measurement: Float,              ",",
  units:  String(',')              ",",
  order:  Int
}
```

Unions

```
Southern California Regional Railroad Authority,"Los Angeles, CA",
U,45,46,46,47,49,51,U,45,46,46,47,49,51
Connecticut Department of Transportation ,"New Haven, CT",
U,U,U,U,U,U,8,U,U,U,U,U,U,8
Tri-County Commuter Rail Authority ,"Miami, FL",
U,U,U,U,U,U,18,U,U,U,U,U,U,18
```

Anonymous:

```
'U' + Int
```

Named:

```
type OptInt = unavailable of 'U'
            |   available of Int
```

Arrays/Lists

```
Southern California Regional Railroad Authority,"Los Angeles, CA",
U,45,46,46,47,49,51,U,45,46,46,47,49,51
Connecticut Department of Transportation ,"New Haven, CT",
U,U,U,U,U,U,8,U,U,U,U,U,U,8
Tri-County Commuter Rail Authority ,"Miami, FL",
U,U,U,U,U,U,18,U,U,U,U,U,U,18
```

```
type OptInt = unavailable of 'U'
            |   available of Int

type counts = OptInt[14] sep(',')
                         term(eor)
```

Dependent Types

```
sdw-01.ab.ca - - [16/12/06] "GET /images/fish.gif HTTP/1.0" 200 8552
sdw-01.ab.ca - DBUser [16/12/06] "GET /images/bug.gif HTTP/1.0" 200 1357
64.233.161.99 - - [16/12/26] "GET /images/plex.gif HTTP/1.0" 304 -
69.30.123.195 - - [16/12/2006] "GET /images/adjoint.gif HTTP/1.0" 304 -
```

```
type responseCode = { x : Int | 99 < x < 600}
```

Dependent Types

```
sdw-01.ab.ca - - [16/12/06] "GET /images/fish.gif HTTP/1.0" 200 8552
sdw-01.ab.ca - DBUser [16/12/06] "GET /images/bug.gif HTTP/1.0" 200 1357
64.233.161.99 - - [16/12/26] "GET /images/plex.gif HTTP/1.0" 304 -
69.30.123.195 - - [16/12/2006] "GET /images/adjoint.gif HTTP/1.0" 304 -
```

```
type method = GET | POST | LINK | UNLINK | ...

fun  check(method, major, minor) = ...

type request =
   { method : method,        ' ',
     url    : String(' '), " HTTP/",
     major  : Int,           '.',
     minor  : Int
   } where check(method, major, minor)
```

Value Abstraction

```
sdw-01.ab.ca - DBUser [16/12/06] "GET /images/bug.gif HTTP/1.0" 200 1357
64.233.161.99 - - [16/12/26] "GET /images/plex.gif HTTP/1.0" 304 -
```

```
Connecticut Department of Transportation ,"New Haven, CT",
U,U,U,U,U,U,8,U,U,U,U,U,U,8
```

Two representations for missing integer: 'U' and '-'. We can use value abstraction to reduce redundancy:

```
type OptInt  = λx.unavailable of x
             |   available of Int
type OptIntU = OptInt 'U'
type OptIntD = OptInt '-'
```

Type Abstraction

```
sdw-01.ab.ca - DBUser [16/12/06] "GET /images/bug.gif HTTP/1.0" 200 1357
64.233.161.99 - - [16/12/26] "GET /images/plex.gif HTTP/1.0" 304 -
```

```
Connecticut Department of Transportation ,"New Haven, CT",
U,U,U,U,U,U,8,U,U,U,U,U,U,8
```

Two different types can be missing: Int and String. We can use type abstraction to reduce redundancy:

```
type Opt = Λt.λx.unavailable of x
                |  available of t
type OptIntU = Opt Int 'U'
type OptIntD = Opt Int '-'
type OptStringD = λy.Opt (String y)'-'
```

Recursive types

```
(Bovine:0.69395,(Gibbon:0.36079,(Orang:0.33636,(Gorilla:0.
17147,(Chimp:0.19268,Human:0.11927):0.08386):0.06124):0.15
057):0.54939,Mouse:1.21460):0.10
```

```
type Entry = {name : String(':'), ':',
              dist : Float}
type Tree  =
    Leaf of Entry
  | Interior of
    '(' * Tree[] (',', ')') * ')' * ':' * Float
```

Pointers?

```
00000000: 0001 face 0000 0073 0400 0000 3600 0000  .......s....6...
00000010: 3000 0000 3500 0000 3000 0000 0000 0000  0...5...0.......
00000020: 0001 0000 0000 0100 0000 0043 0001 0000  ...........C....
...
00000070: 0000 0100 0000 0800 0000 0968 6173 6b65  ...........haske
00000080: 6c6c 3938 0000 0007 4350 5554 696d 6500  ll98....CPUTime.
00000090: 0000 0462 6173 6500 0000 0847 4843 2e42  ...base....GHC.B
000000a0: 6173 6500 0000 0e47 4843 2e46 6f72 6569  ase....GHC.Forei
000000b0: 676e 5074 7200 0000 0e53 7973 7465 6d2e  gnPtr....System.
000000c0: 4350 5554 696d 6500 0000 0a67 6574 4350  CPUTime....getCP
000000d0: 5554 696d 6500 0000 1063 7075 5469 6d65  UTime....cpuTime
000000e0: 5072 6563 6973 696f 6e                   Precision
```

```
type Dictionary = {count : SBH_uint32(4),
                   ids   : Hstring[count]}
...
type Hi = { id   : B_uint32(4) where checkId(id),
            dict : Dictionary Pointer(4), ...}
```

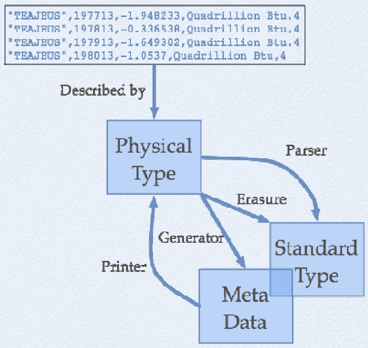
Type summary

- Base types
- Tuples
- Singleton types
- Records
- Unions
- Lists/Arrays
- Value abstraction
- Type abstraction
- Dependent types
- Recursive types
- Pointers
- ???

Differences

- Data layout *is not* under the control of the type system.
  - Physical types need some extra information: separators, terminators.
  - Many physical types map to the same internal type: String(' '), String(':'), SBH_uint32, B_uint32 ...
- Dependent types much more important for physical types:
  - Missing value representations, value-level constraints, embedded array lengths, union tags.
- We should not assume data conforms 100% to description.

Meta data



Leverage

- Convert PADS description into a collection of tools:
  - Accumulators
  - Histograms
  - Clustering tool
  - Formatters
  - Translator into XML, with corresponding XML Schema.
  - XQueries using Galax's data interface
  - …
- **Long term goal**: Provide a compelling suite of tools to overcome inertia of a new language and system.

Accumulators

- Statistical profile of "leaves" in a data source:

```
<top>.length : uint32
good: 53544   bad: 3824    pcnt-bad: 6.666
min: 35  max: 248591  avg: 4090.234
top 10 values out of 1000 distinct values:
tracked 99.552% of values
 val:  3082 count:  1254  %-of-good:  2.342
 val:   170 count:  1148  %-of-good:  2.144
                  . . .

. . . . . . . . . . . . . . . . . . . . . . .
 SUMMING    count:  9655  %-of-good: 18.032
```

Not all lengths were legal!

- Suggested by AT&T user to get "bird's eye" view of their 4000 daily feeds.
- Used at AT&T for vetting data (and for debugging PADS descriptions).

Pretty printer

- Customizable program to reformat data:

```
207.136.97.49 - - [15/Oct/1997:18:46:51 -0700] "GET /tk/p.txt HTTP/1.0" 200 30
tj62.aol.com - - [16/Oct/1997:14:32:22 -0700] "POST /scpt/dd@grp.org/confirm HTTP/1.0" 200 941
```

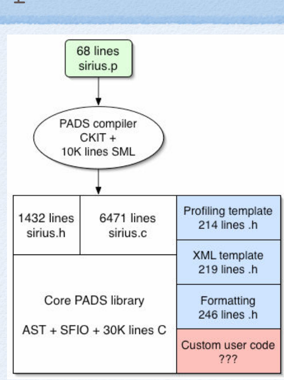
Normalize time zones
Normalize delimiters

Drop unnecessary values
Filter/repair errors

```
207.136.97.49|-|-|10/16/97:01:46:51|GET|/tk/p.txt|1|0|200|30
tj62.aol.com|-|-|10/16/97:21:32:22|POST|/scpt/dd@grp.org/confirm|1|0|200|941
```

- Users can override printing on a per type basis.
- Used at AT&T to normalize monitoring data before loading into a relational database.

Implementation



Why a domain specific language?

- Dramatically **shorter code** (68 lines versus ~7.9K lines).
- Description is short enough to **serve as documentation**.
- **Safer**: error code inserted automatically and completely (we just have to get the compiler right…).
- We can leverage the declarative specification to produce **value-added tools**.

Future research directions

- Design
  - How can we specify error-aware data transformations?
  - Can we infer a data transformation between two descriptions?
  - How can we express application-specific information?
  - How can we generate template programs for arbitrary data source?
- Implementation
  - How can we specialize generated libraries to incorporate application-specific information?
  - How can we optimize streaming XQueries?
- Theory
  - How do we precisely specify the semantics of PADS? (…tomorrow…)
  - What is the expressiveness of PADS vs. context free grammars?
- Engineering
  - How do we build the system to make it easy to add new base types?
  - New libraries and tools? New language bindings?

Contributors

- Kathleen Fisher (AT&T)
- Robert Gruber (Google)
- Mary Fernandez (AT&T)
- Joel Gottlieb (AT&T)
- Yitzhak Mandelbaum (Princeton → AT&T)
- Martin Strauss (University of Michigan)
- David Walker (Princeton)
- Xuan Zheng (University of Michigan)

Try it!

- Available for download with an open source license (CPL 1.0).
- Demo of accumulators, format program, and XML conversion.
- Send us feedback!

www.padsproj.org