



WE ARE ALL BLESSED

Bruce Tate @redrapids

Dave Thomas @pragdave

DAVE'S STORY



Toby Bailey seated and David Thomas operating an educational analogue computer (L7819/7).





https://upload.wikimedia.org/wikipedia/commons/0/0b/Hand-operated_Card_Punch-2.jpg



https://upload.wikimedia.org/wikipedia/commons/e/e8/IBM_card_punch_029.JPG

ALWAYS LOVED LANGUAGES

Algol, APL, APT, Basic, BCP,
C, C++, COBOL, Delphi,

...

TECO, TCL, TeX, XSLT

comp.lang.misc

BRUCE'S STORY



http://home.arcor.de/andreas.ernaue/images/Tandy_Radio_Shack_TRS-80_Color_Computer_full.jpg



[http://www.trs-80.com/images/hwimage-model-ii-rsc03-\[26-4001\]-\(rs\).png](http://www.trs-80.com/images/hwimage-model-ii-rsc03-[26-4001]-(rs).png)

I REALLY LOVED BASIC.



MOSTLY PASCAL; C, LISP ...

I LIKED PASCAL.



C, C++, JAVA.

I TOLERATED JAVA.

SOLO

LOVE / HATE RELATIONSHIP





https://commons.wikimedia.org/wiki/File:CQ311_switching_tracks.jpg

***SHUT UP AND DO SOMETHING NONTRIVIAL IN
RUBY, AND THEN WE CAN TALK.***

- Dave Thomas

The
Pragmatic
Programmers

From Java to Ruby

Things Every Manager Should Know

Bruce Tate

The Facets



of Ruby Series

The
Pragmatic
Programmers

Seven Languages in Seven Weeks

A Pragmatic
Guide to
Learning
Programming
Languages

Bruce A. Tate

Edited by Jacquelyn Carter





There is a central quality which is the root criterion of life and spirit in a man, a town, a building, or a wilderness. This quality is objective and precise, but it cannot be named.

Christopher Alexander
The Timeless Way of Building



**QUALITY
WITHOUT A
NAME**



QWAN



QWAN

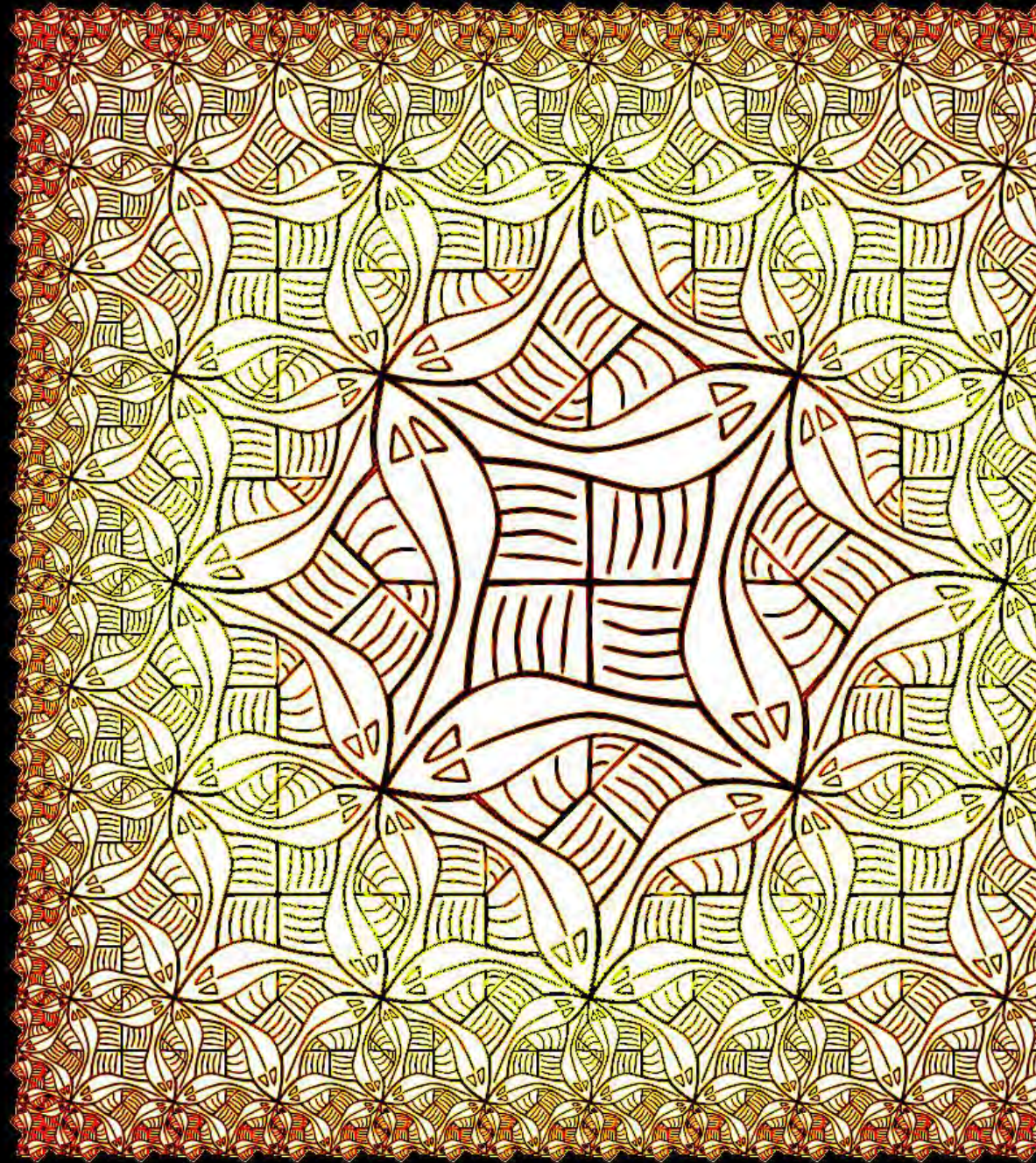
COMPLETENESS



*It seems there is a
positive correlation
between the simplicity
of the rules and the
quality of the algebra
as a description tool.*

Peter Henderson

<https://github.com/shashi/escher-demo>



SOUL



I KNOW IT WHEN I SEE IT



<https://prezi.com/ly46uhwu9vrj/apple-history/>



<http://www.sandyflat.net/digerati/ibmps1/index.htm>

I KNOW IT WHEN I SEE IT



Angus L. Macdonald Bridge, Halifax, Nova Scotia, Canada



Salim Virji

I KNOW IT WHEN I SEE IT

```
int sum = 0;

for (int i = 0;
     i < sizeof(arr)/sizeof(arr[0])
     i++) {
    sum += arr[i];
}
return sum;
```

```
list |> Enum.reduce(&+/2)
```

QWAN IN RUBY

RUN CODE DURING "COMPILATION"

```
class Dictionary  
  WORDS = File.readlines("/usr/dict/words")  
  
  def size  
    WORDS.size  
  end  
  
end
```

METAPROGRAMMING AND DSLS

```
get '/' do  
  "Hello, World!. It's #{Time.now}"  
end
```

METAPROGRAMMING AND DSLS

```
set(:probability) { |value| condition { rand <= value } }  
  
get '/win_a_car', :probability => 0.1 do  
  "You won!"  
end  
  
get '/win_a_car' do  
  "Sorry, you lost."  
end
```

<http://www.sinatrarb.com/intro.html>

DUCKTYPING AND THE OBJECT MODEL

```
test "'not found' is handled" do
  mock_server = "xxx"

  def mock_server.query(*)
    { status: 404, message: "not found" }
  end

  response = @my_app.send_query(mock_server)

  assert_equal :not_found, response
end
```

(YOUR AESTHETICS MAY VARY 😊)

QWAN IN ELIXIR


```
signature = join(sort(to_list(word)))
```

```
signature = word |> to_list |> sort |> join
```

More than a convenience —it reflects and reinforces a key facet of functional programming.

```
signature = word |> to_list |> sort |> join
```

FUNCTIONS TRANSFORM DATA

SPEAKING OF FUNCTIONS

```
case File.open("somedata.csv") do
  {:ok, file} -> process(file)
  {:error, reason} -> report(reason)
end
```

SPEAKING OF FUNCTIONS

```
"somedata.csv"  
|> File.open  
|> case({:ok, file} -> process(file)  
        {:error, reason} -> report(reason))
```

It's all transformations


```
defmodule LogWriter do

  def log(msg) do
    if Application.get_env(:app, :debug) do
      IO.puts msg
    end
  end
end
```

How many reserved words?

```

%C %R %S %W %c %r %s %w && ! |> || .. :: <>
= @ ^ __CALLER__ __DIR__ __ENV__ __MODULE__
  __aliases__ __block__ alias! alias and
binding callback case cond def defdelegate
  defexception defimpl defmodule
  defoverridable defp defprotocol defstruct
destructure for get_and_update_in if import
  in is_nil match? or put_in quote raise
receive require reraise super to_char_list
to_string try unless unquote update_in use
  var! with

```

THE LANGUAGE **IS** THE LANGUAGE

```
%C %R %S %W %c %r %s %w && ! |> || .. :: <> = @ ^ __CALLER__  
__DIR__ __ENV__ __MODULE__ __aliases__ __block__ alias! alias  
and binding callback case cond def defdelegate defexception  
defimpl defmodule defoverridable defp defprotocol defstruct  
destructure for get_and_update_in if import in is_nil match? or  
put_in quote raise receive require reraise super to_char_list  
to_string try unless unquote update_in use var! with
```

h File.open

```
def open(path, modes \\ [])
```

he given `path` according to the given list of modes.

r to write and read files, one must use the functions in the `IO` module. ult, a file is opened in binary mode, which requires the functions `read/2` and `IO.binwrite/2` to interact with the file. A developer may pass s an option when opening the file and then all other functions from `IO` ilable, since they work directly with Unicode data.

owed modes:

`read` - the file, which must exist, is opened for reading.

`write` - the file is opened for writing. It is created if it does not st.

the file does exists, and if write is combined with read, the file l be truncated.

`append` - the file will be opened for writing, and it will be created it does not exist. Every write operation to a file opened with append l take place at the end of the file.

`exclusive` - the file, when opened for writing, is created if it does exist. If the file exists, open will return `{:error, :eexist}`.

`char_list` - when this term is given, read operations on the file will urn char lists rather than binaries.

`compressed` - makes it possible to read or write gzip compressed files.

compressed option must be combined with either read or write, but not h. Note that the file size obtained with `stat/1` will most probably not ch the number of bytes that can be read from a compressed file.

`utf8` - this option denotes how data is actually stored in the disk

e and makes the file perform automatic translation of characters to and

UTF-8

.../erlang-factory-sf-2016/*t<2>* Term : char run

```

%%%-----
%%% @author some name me@hostname.local
%%% @copyright (C) 2013, some name
%%% @doc
%%%
%%% @end
%%% Created : 26 May 2013 by some name me@hostname.local
%%%-----
-module().

-behaviour(gen_server).

%% API
-export([start_link/0]).

%% gen_server callbacks
-export([init/1, handle_call/3, handle_cast/2, handle_info/2,
terminate/2, code_change/3]).

-define(SERVER, ?MODULE).

-record(state, {}).

%%%-----
%%% API
%%%-----

%%-----
%% @doc
%% Starts the server
%%
%% @spec start_link() -> {ok, Pid} | ignore | {error, Error}
%% @end
%%-----
start_link() ->
gen_server:start_link({local, ?SERVER}, ?MODULE, [], []).

%%%-----
%%% gen_server callbacks
%%%-----

%%-----
%% @private
%% @doc
%% Initializes the server
%%

```

- 128 lines
- 22 lines of code

```
defmodule MyServer do
  use GenServer

  # ...
end
```



```
defmodule Agent.Server do
  use GenServer

  def init(fun) do
    initial_call(fun)
    {:ok, run(fun, [])}
  end

  def handle_call({:get, fun}, _from, state) do
    {:reply, run(fun, [state]), state}
  end

  def handle_call({:get_and_update, fun}, _from, state) do
    case run(fun, [state]) do
      {reply, state} -> {:reply, reply, state}
      other          -> {:stop, {:bad_return_value, other}, state}
    end
  end
end
```

THE QWAN GOES ON

- Testing
- Protocols
- Rationalized library APIs
- Mix
- Hex
- exrm
- People
- ...



© David Merrett, CCA. <https://www.flickr.com/photos/davehamster/364659502/in/pool-cathedral/>


A low-angle, upward-looking shot of a highly ornate, octagonal dome ceiling. The ceiling is covered in intricate carvings and features several rectangular skylights that allow bright light to stream in, creating a dramatic effect. Between the skylights are panels of colorful frescoes depicting various scenes. The overall architecture is classical and grand.

QWAN IS TWO WAY

WE ARE INSPIRED IN THE PRESENCE OF QUALITY



**PIPELINES AND PROTOCOLS INSPIRE US TO
CODE IN TERMS OF TRANSFORMATIONS**



**BEHAVIOURS AND MACROS INSPIRE US TO
CODE IN THE DOMAIN, NOT THE
IMPLEMENTATION**



**GREAT DOCUMENTATION INSPIRES US TO
WRITE MORE THAN CODE**
(WHICH MAKES US THINK ABOUT OUR WORK DIFFERENTLY)



**MODULARITY AND EXRM INSPIRE US TO
WRITE TINY APPS AND DEPLOY OFTEN**



**GREAT LEADERS INSPIRE US
TO EMULATE THEM**



BUT THIS IS DIFFICULT...

<http://www.mountainsoftravelphotos.com/Everest/Best/slides/Gokyo%205%20Sagarmaths%20View%208%20Everest%20Close%20up.jpg>

A black and white photograph of two men in flight suits, smiling and looking towards the camera. They are wearing goggles on their foreheads. The man on the left is wearing a flight suit with a zipper and a strap. The man on the right is wearing a flight suit with a strap and a corrugated tube. The background is a plain, light color.

LANGUAGE MAKING IS SEEKING QWAN

INTEREST SUMMIT
metres

CAMP 4 SOUTH COL
8,000 metres

DEATH ZONE:
The lack of oxygen ab
8000 metres can be fa
to climbers.

8,000 m

CAMP 3
7,162 metres

CAMP 2
6,400 metres

CAMP 1
5,333 metres

BASE CAMP
5,334 metres

**LANGUAGE MAKING
IS SOMETIMES HARD**



**LANGUAGE MAKING
IS SOMETIMES MESSY**



HOW TO REACH THE TOP?

SHERPA

SHR - pə

A people living on the border of Nepal and Tibet. Sherpas live at a high altitude. They are capable of functioning at a much higher level at the high elevations found on Everest.



<http://www.nbcnews.com/news/asia/earthquake/nepal-katmandu-summit-everest-summit-attempt-n44534>

An aerial photograph showing a large number of colorful tents (blue, yellow, red, green) pitched on a steep, rocky mountain slope. The terrain is covered with light-colored rocks and patches of snow. The tents are arranged in a somewhat organized manner, following the contours of the slope. The text "SHERPAS MAKE CAMP" is overlaid in large, white, bold letters across the center of the image.

SHERPAS MAKE CAMP

An aerial photograph of a village built on a steep, rocky hillside. The houses are painted in various bright colors, including red, blue, yellow, and green, which stand out against the grey and white rocky terrain. The houses are clustered together, following the contours of the hill. The overall scene suggests a high-altitude or mountainous region.

THE ERLANG TEAM

JOSÉ'S GREAT DECISION

2012



<http://1of7.org/blog/13983880> (1 of 7 blog)

A photograph of two Sherpas carrying heavy loads on a mountain trail. The Sherpa in the foreground is carrying a large blue barrel on his back, with a green barrel on top of it. A long wooden pole is balanced horizontally across the barrels. The Sherpa is wearing a grey t-shirt and dark pants. The second Sherpa is partially visible behind him, also carrying a large barrel. The background shows a steep, rocky mountain slope with sparse vegetation and a cloudy sky.

SHERPAS TAKE THE HEAVIEST LOADS

A man in a grey t-shirt and dark pants is carrying a large blue barrel on his back, secured with ropes. A long, dark pipe is balanced horizontally across the barrel. He is walking on a dirt path in a mountainous, forested area. Another person is visible in the background, also carrying a large barrel. The word "PLUG" is overlaid in large, white, sans-serif capital letters in the center of the image.

PLUG



<https://2012everestbasecamp.files.wordpress.com/2012/01/everest11.jpg?860.jpg>

A wide-angle photograph of a snowy mountain peak. Three Sherpas are visible, each carrying a large backpack and using ropes to secure a series of ladders that span a deep crevasse. The Sherpas are positioned at different points along the ladder line, moving from left to right. The snow is bright white, and the sky is a pale, hazy blue. The overall scene conveys a sense of high-altitude adventure and teamwork.

SHERPAS PREPARE THE WAY



ELIXIR AND ELM

A DARK HISTORY

National Geographic

https://www.nationalgeographic.com/remote-video-import/s3.amazonaws.com/NGC/video/900006448_4a04-1937-afdb-da3e5ff60010-why-sherpas-climb.jpeg



WE ARE ALL SHERPAS























WHAT YOU DO...

WILL AFFECT WHAT THEY

WILL DO



WE CAN SHARE THE LOAD



NOT BECAUSE WE HAVE TO



**BUT BECAUSE
WE ARE BLESSED.**



**BUT BECAUSE
WE ARE BLESSED!**