

Why Elixir Matters

A Genealogy of Functional Programming

Osa Gaius
github.com/osagaius
@osagaius

Who am I?

Engineer at Mailchimp

Organizer - Atlanta Elixir Meetup



Outline

1. Introduction
2. Genealogy
3. History of Functional Programming
4. Why Elixir Matters
5. Moving Forward
6. Conclusion

Introduction

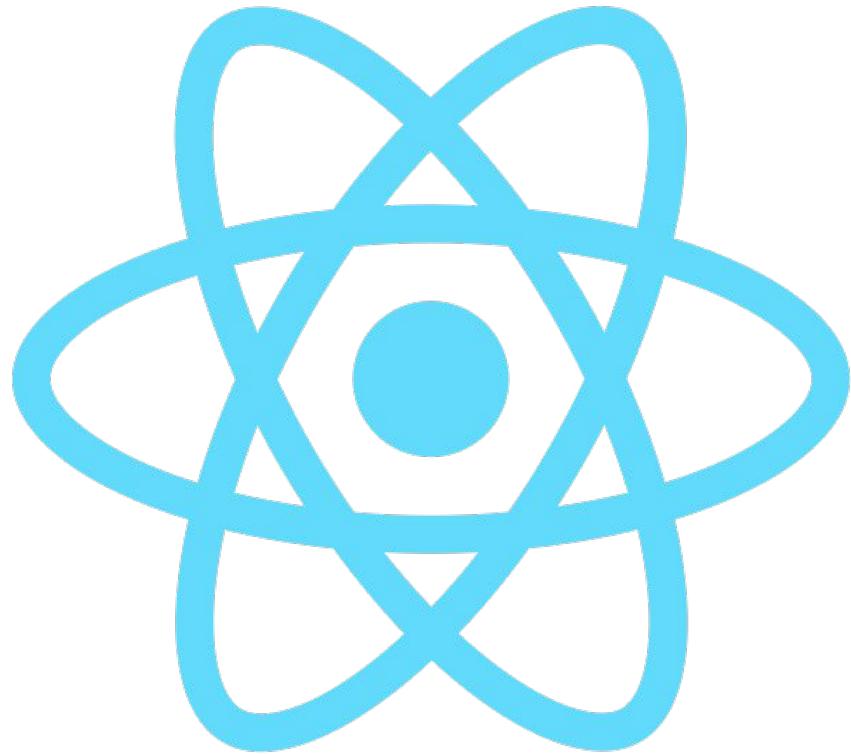


WHY DO YOU LIKE FUNCTIONAL
PROGRAMMING SO MUCH? WHAT
DOES IT ACTUALLY GET YOU?

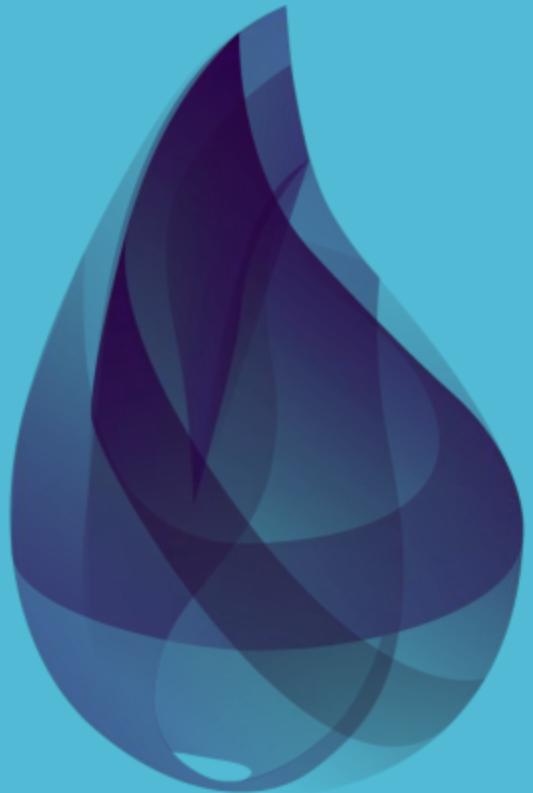
TAIL RECURSION IS
ITS OWN REWARD.



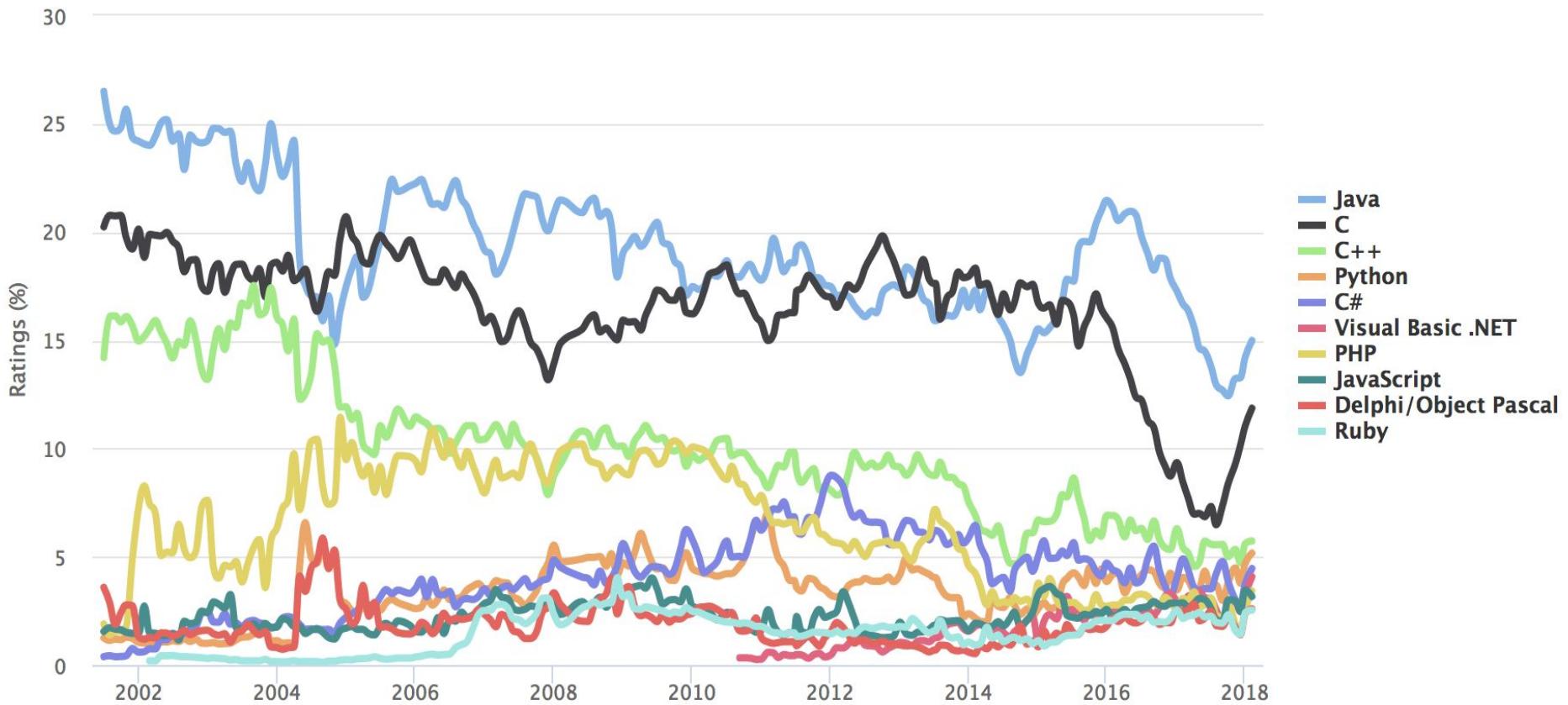






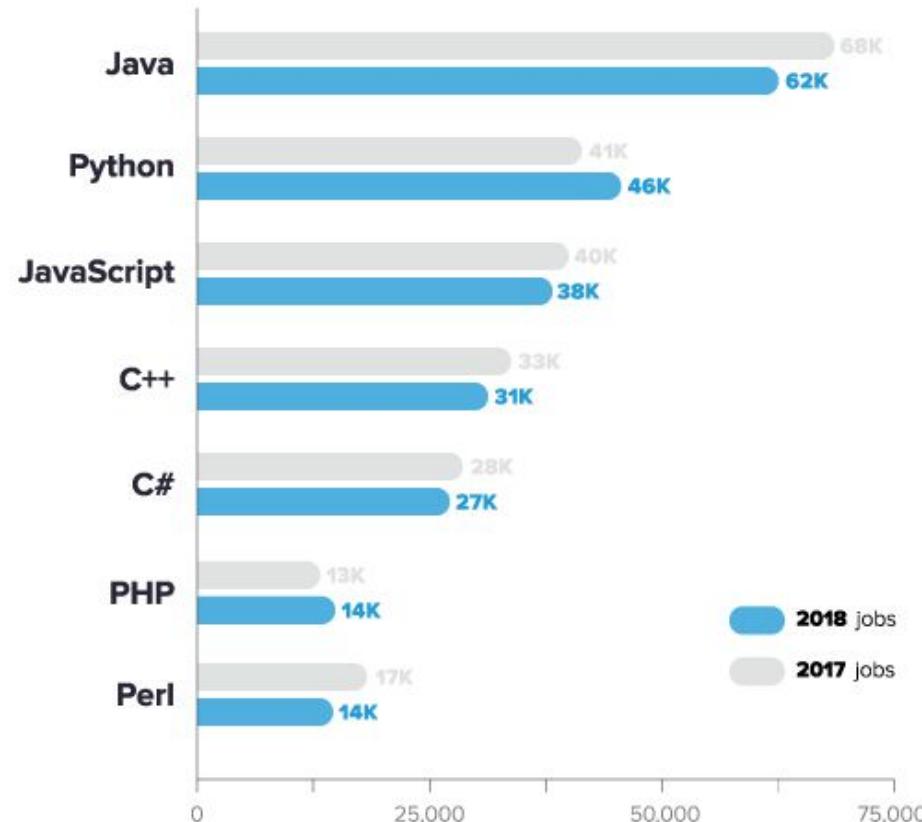


Language Popularity (TIOBE Index)



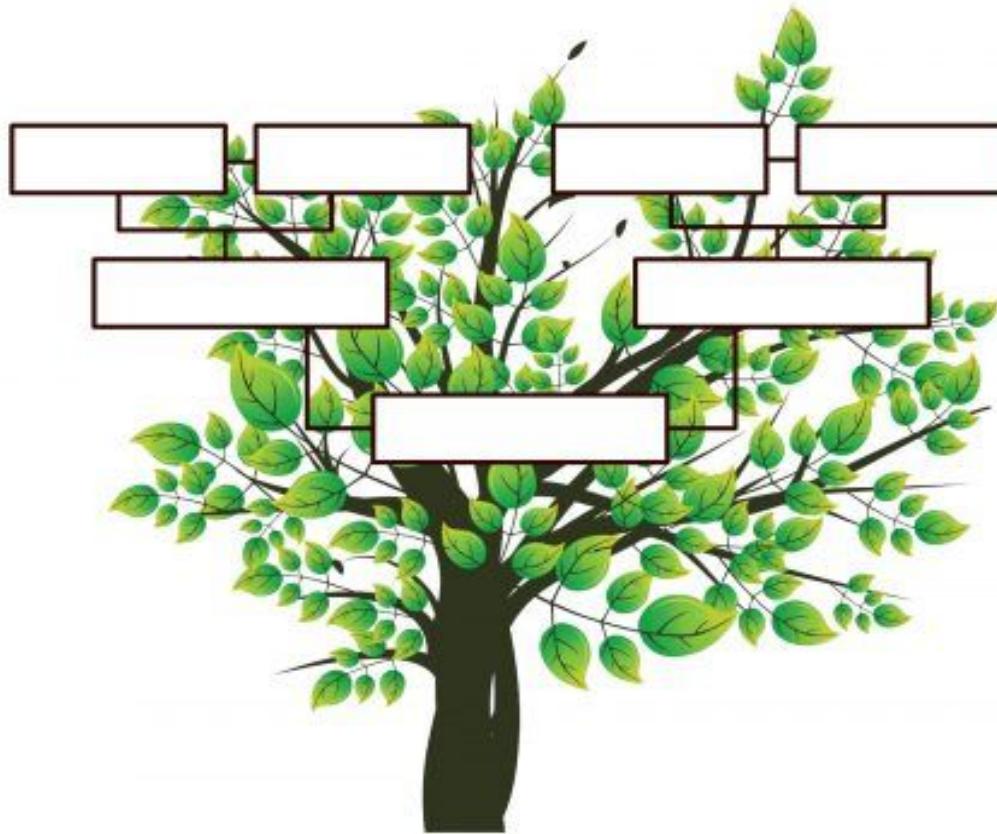
Job postings containing top languages

Indeed.com - November, 17th 2017



Genealogy

jee-nee-ol-uh-jee



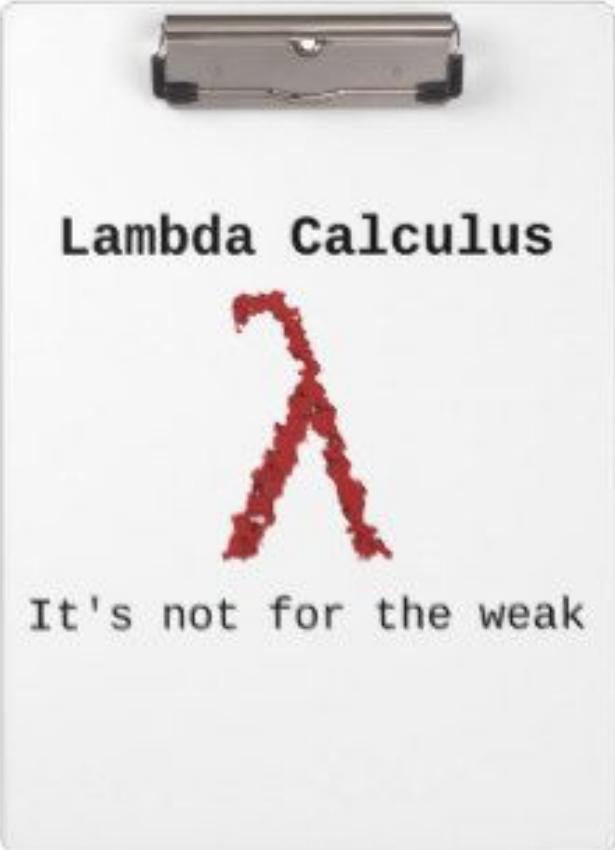


History of Functional Programming





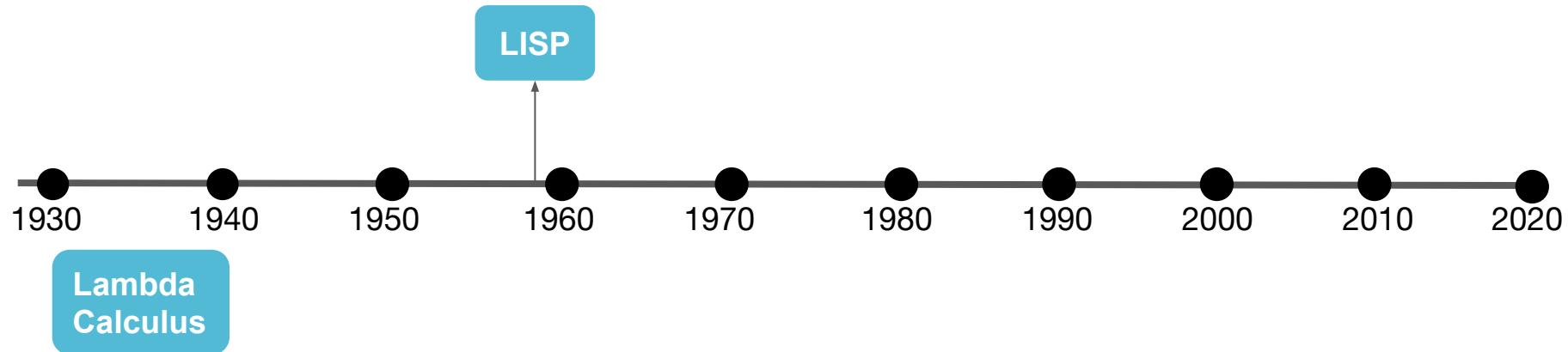
Lambda
Calculus

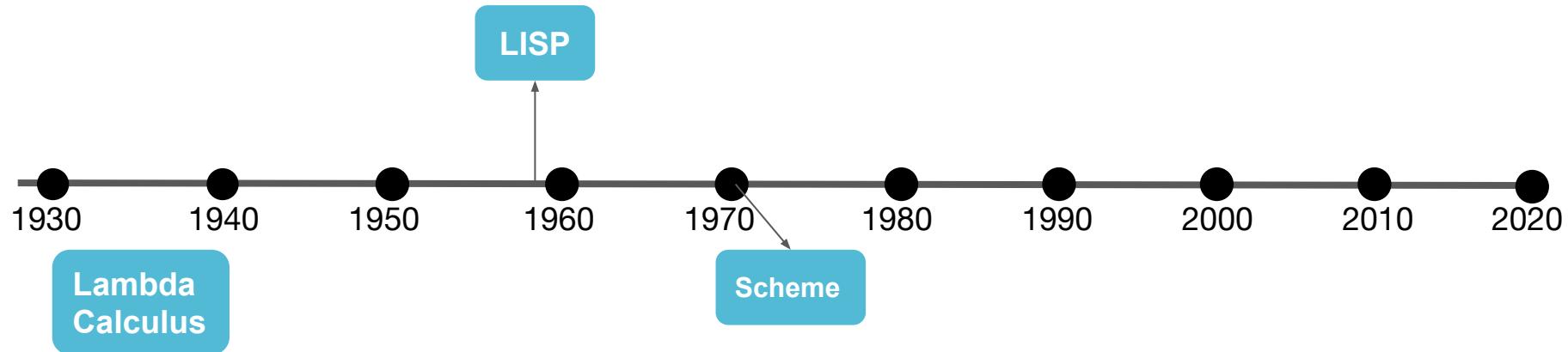


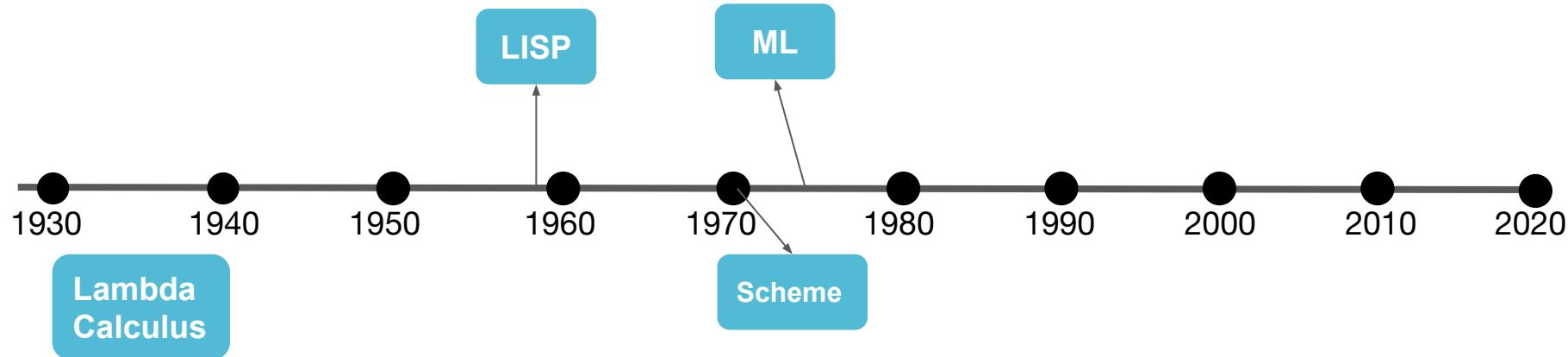
Lambda Calculus

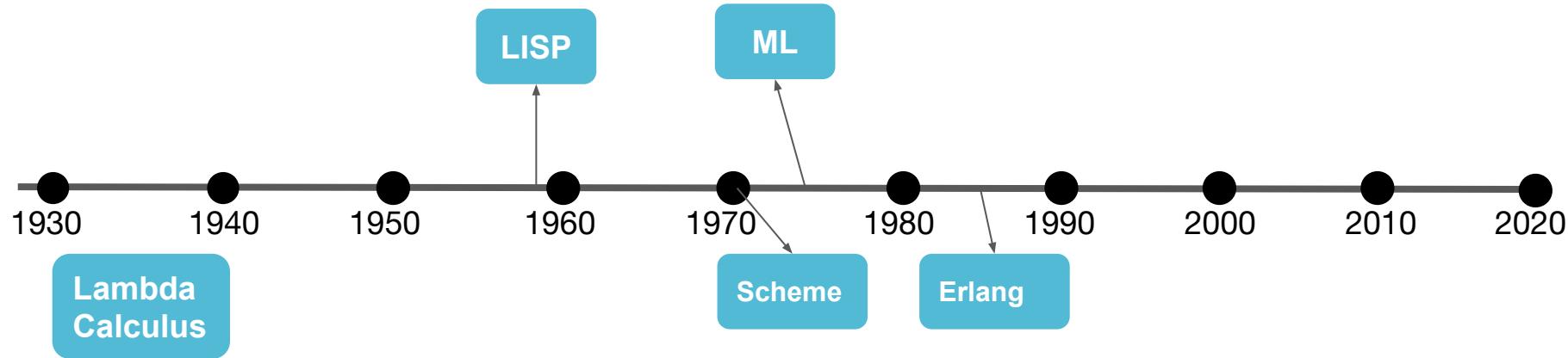


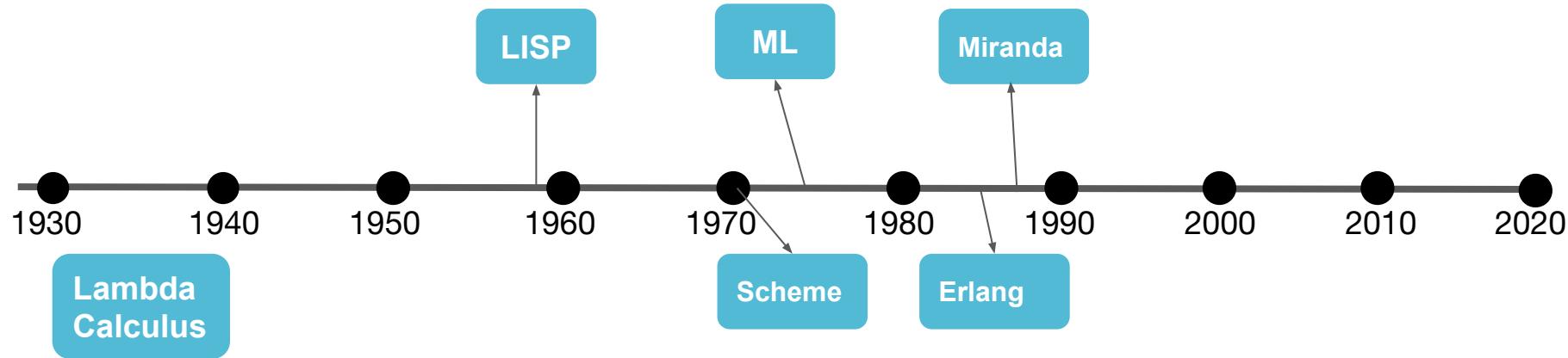
It's not for the weak

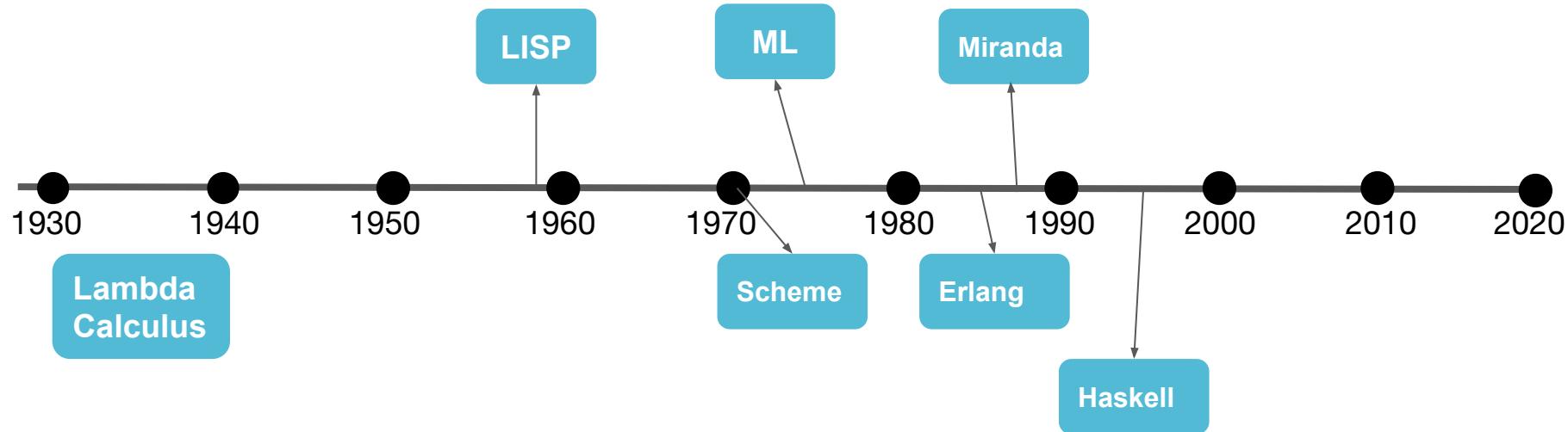


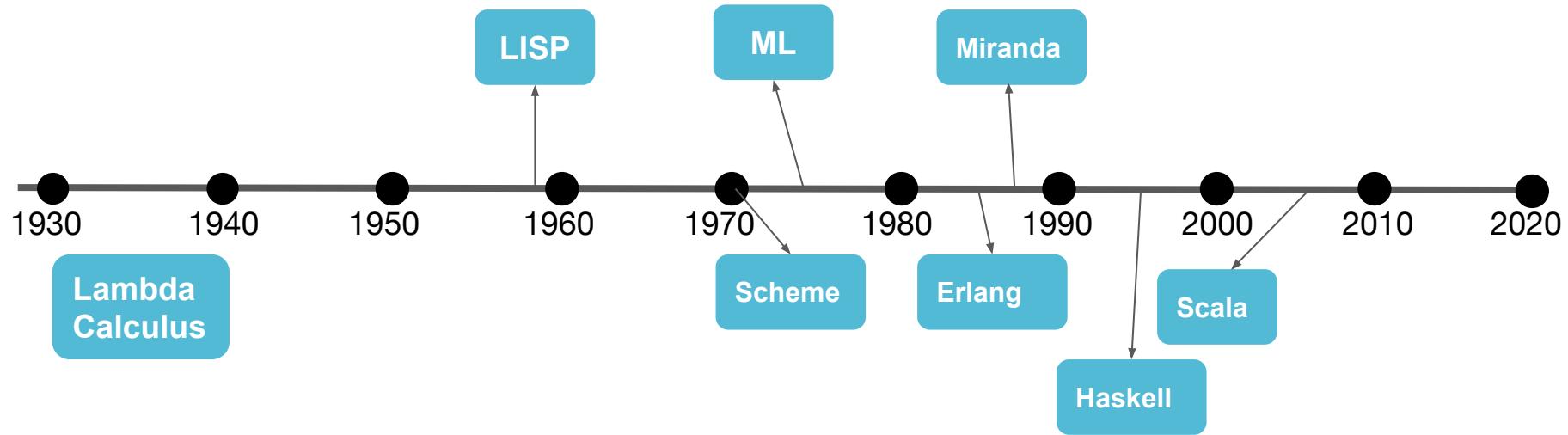


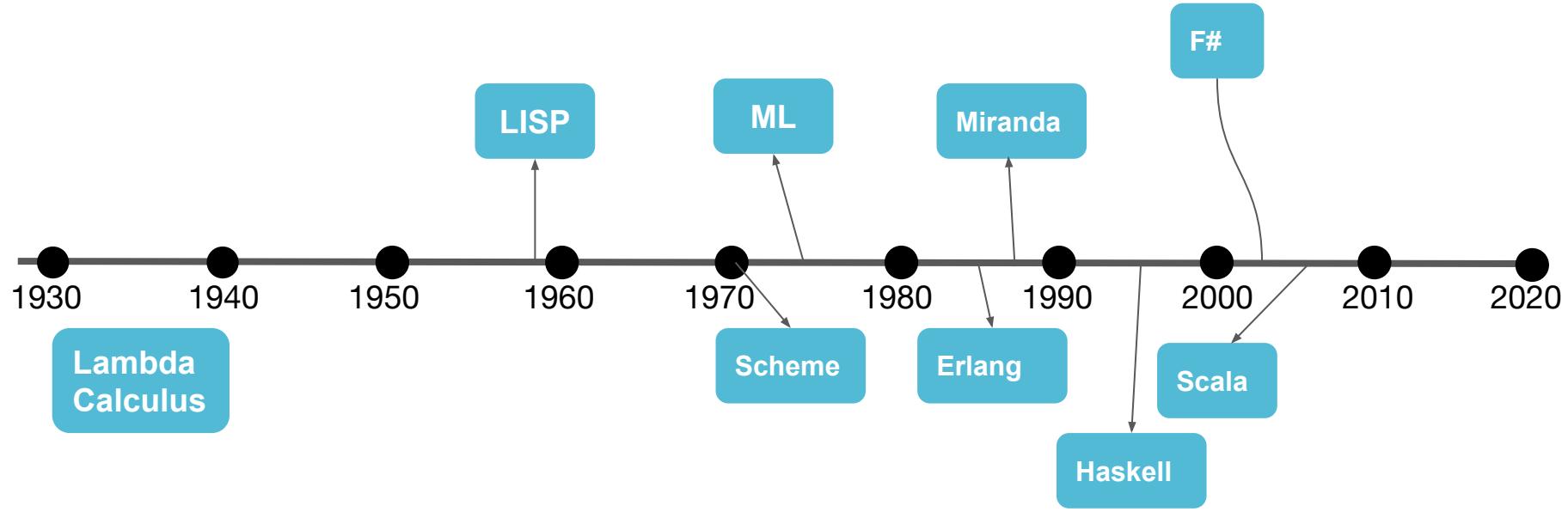


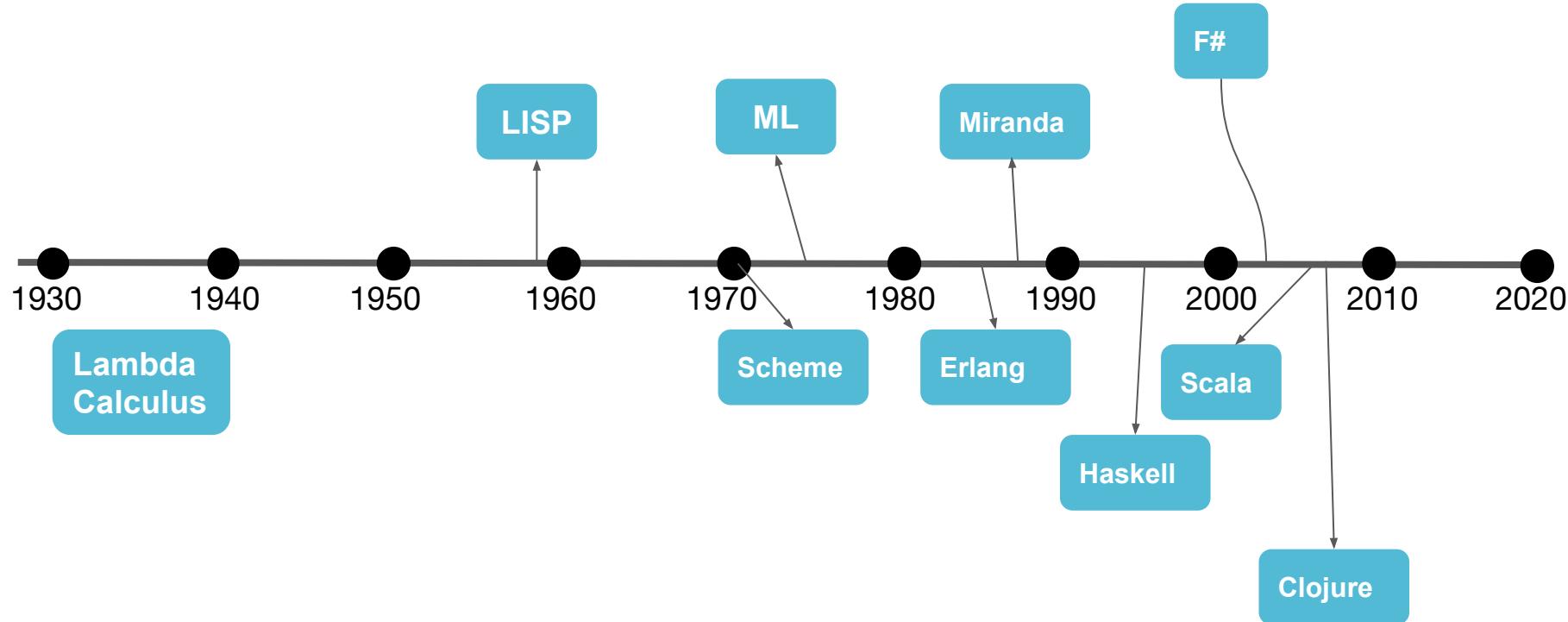


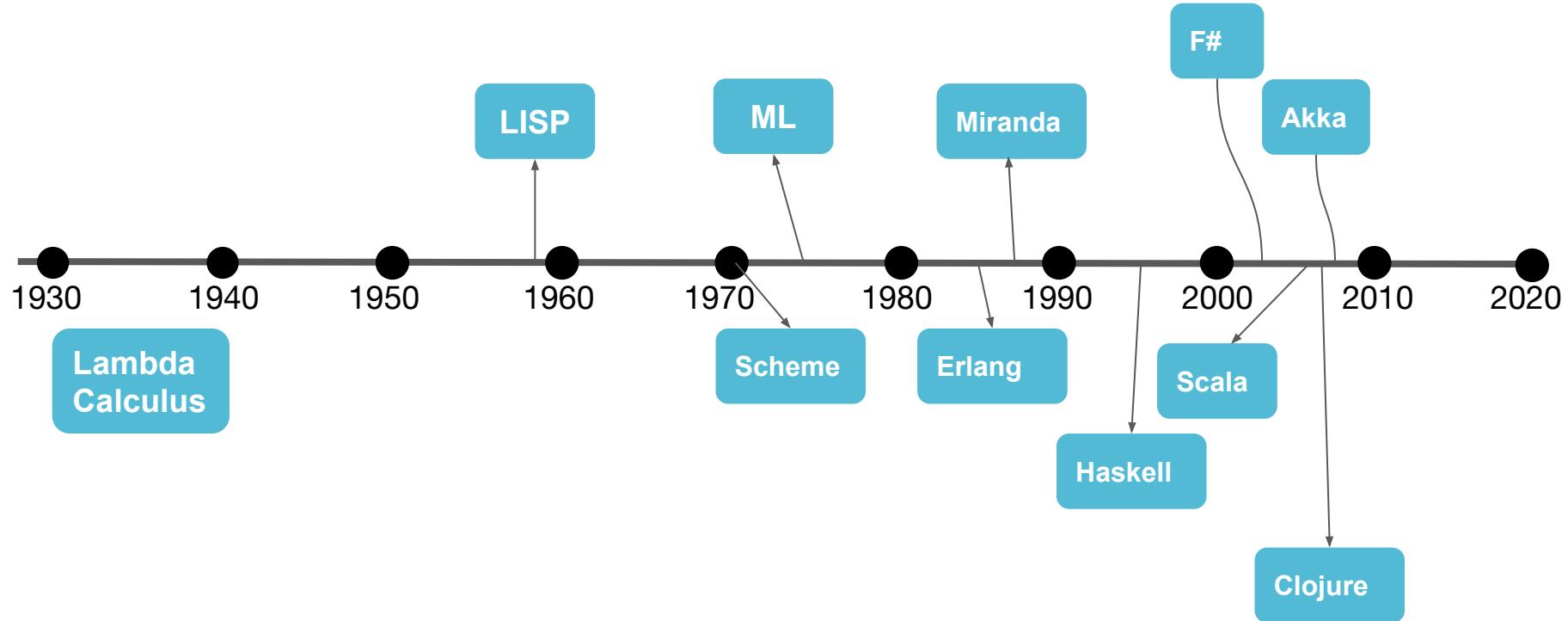


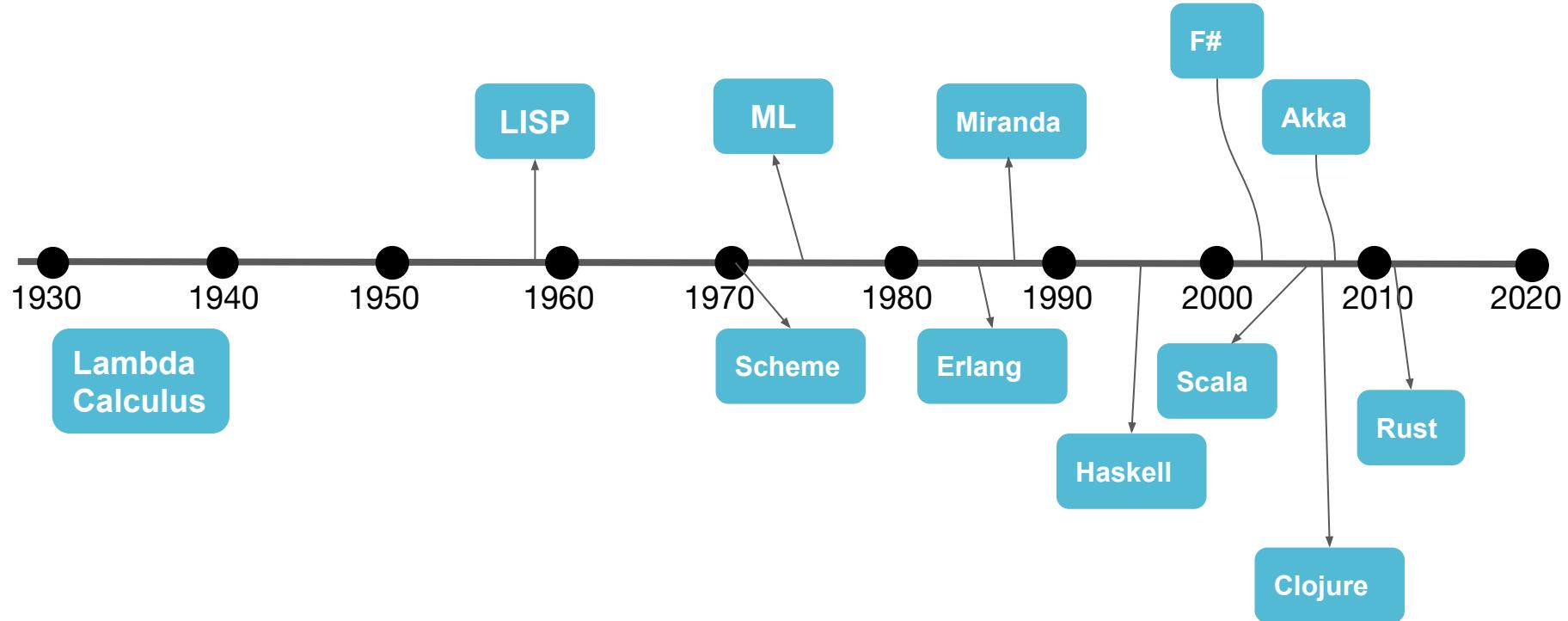


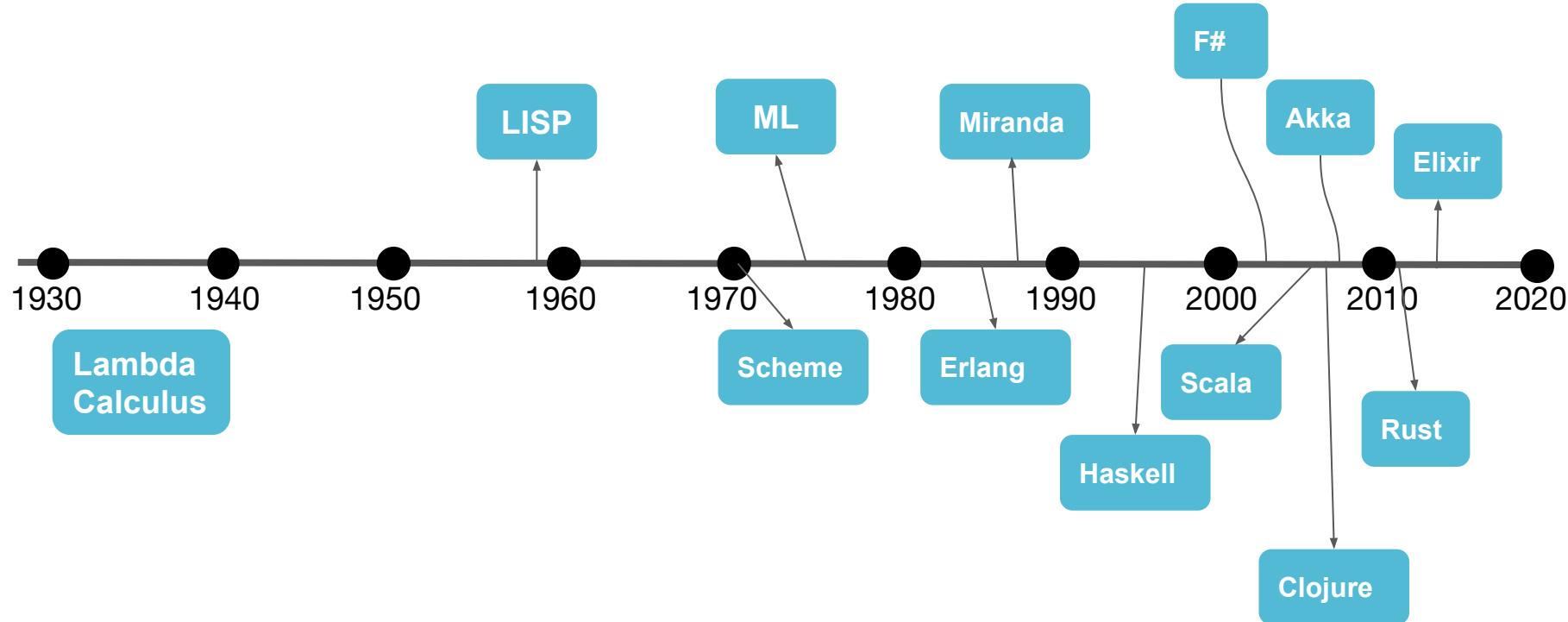


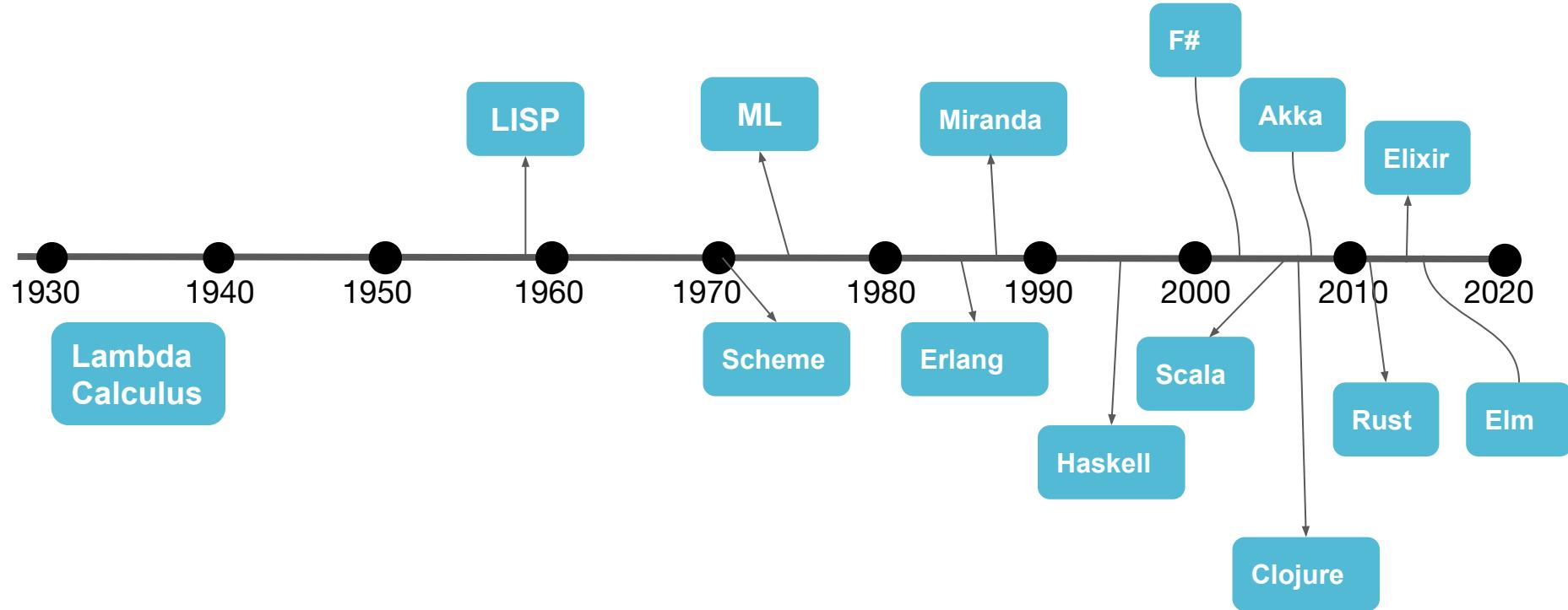












Why Elixir Matters

*“The rise in popularity of the Internet
and the need for non-interrupted
availability of services has extended
the class of problems that Erlang can
solve.”*

Joe Armstrong

Erlang x Elixir

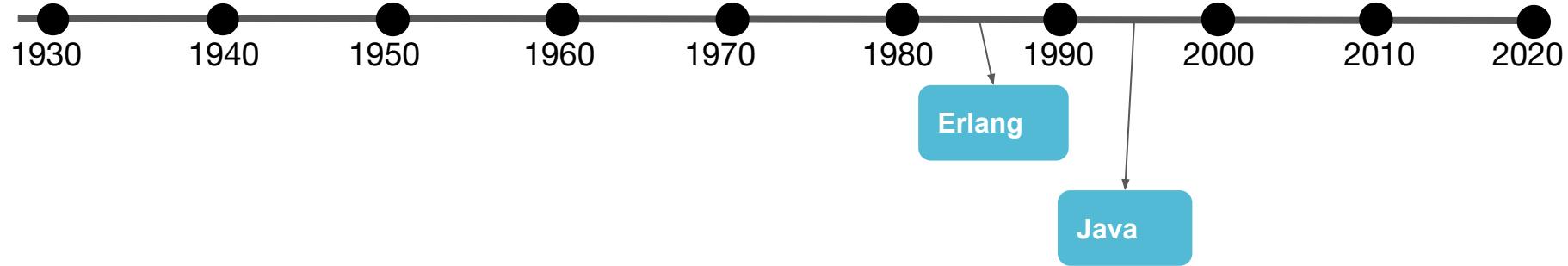
- Syntax
- The Web
- Evangelism

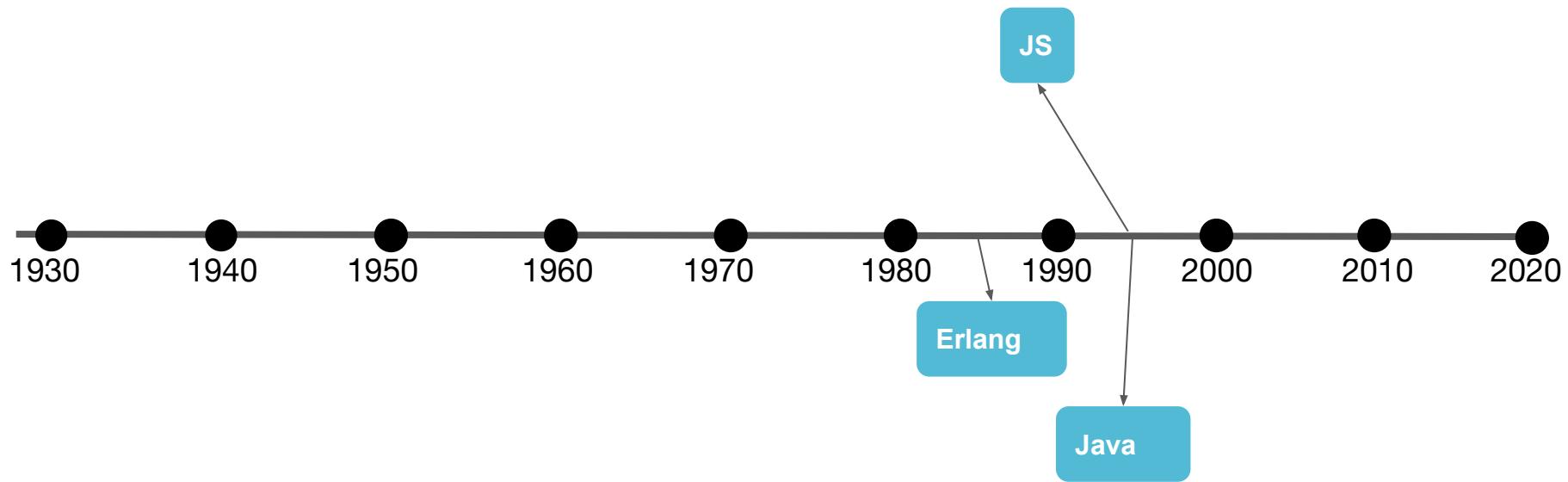
Erlang x Elixir

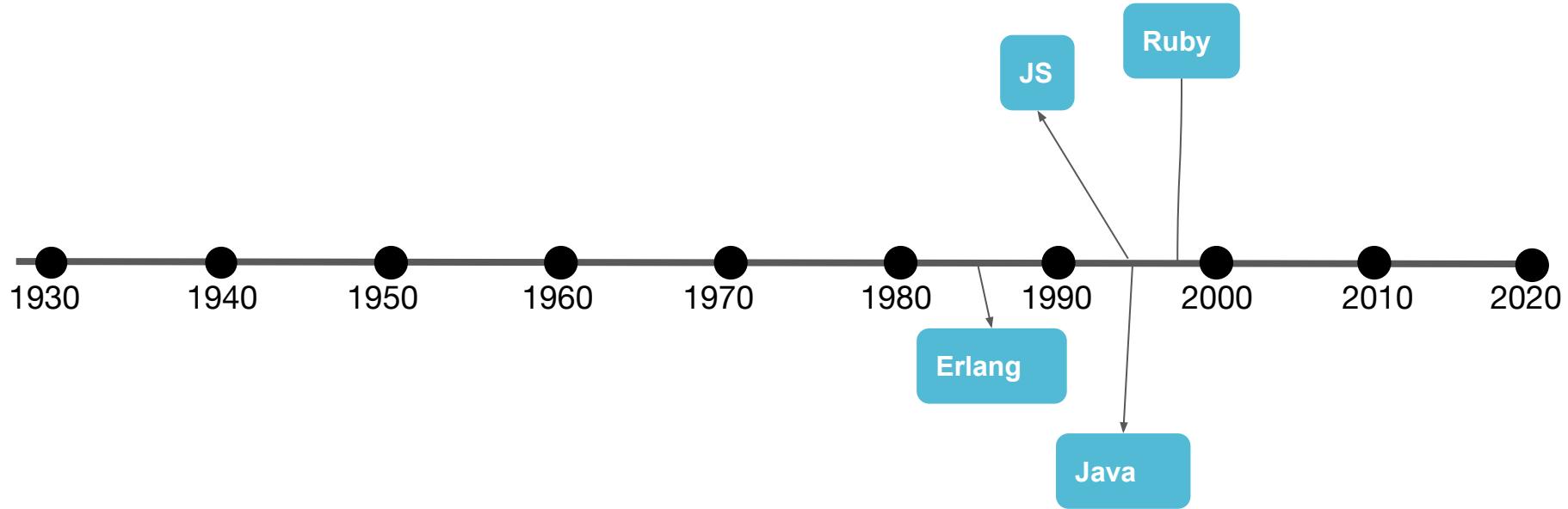
- Syntax
- **The Web**
- Evangelism

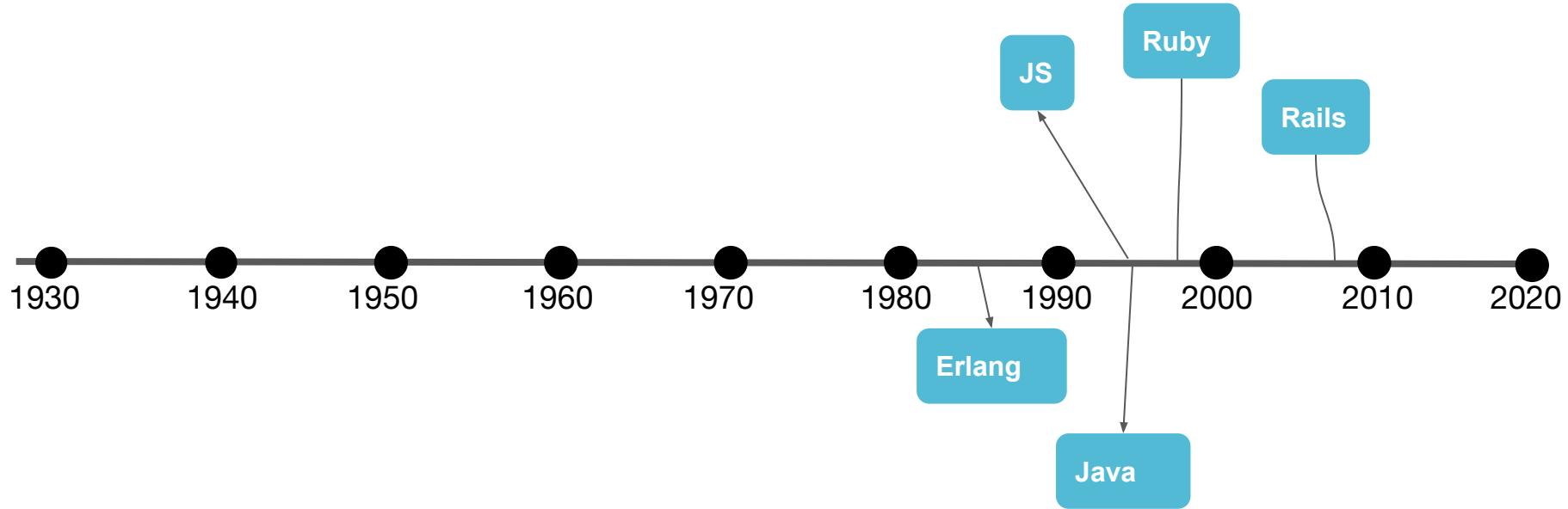


Erlang









Erlang x Elixir

- Syntax
- The Web
- **Evangelism**

*“The rise in popularity of the Internet
and the need for non-interrupted
availability of services has extended
the class of problems that Erlang can
solve.”*

Joe Armstrong

Moving Forward

“Unpopular languages are niche languages”

Meyerovich and Rabkin

Moving Forward

- **Breadth vs Depth**
- Evangelism

Breadth vs Depth

- Libraries (aka Gems)
- Domains

Moving Forward

- Breadth vs Depth
- **Evangelism**

Evangelism

- Marketing
- Consultation
- Fan Out

Conclusion

Gracias!

Questions?

**github.com/osagaius
@osagaius**