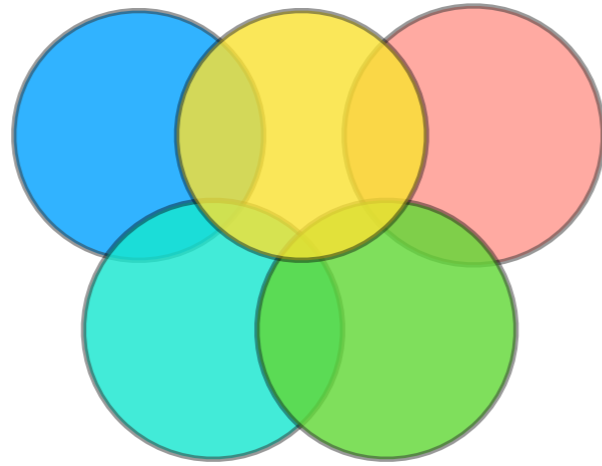




State of the OCaml Platform

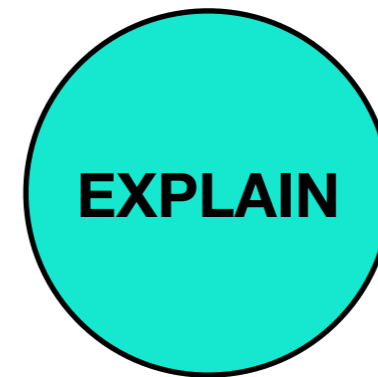
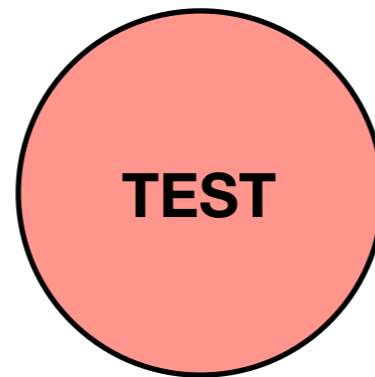
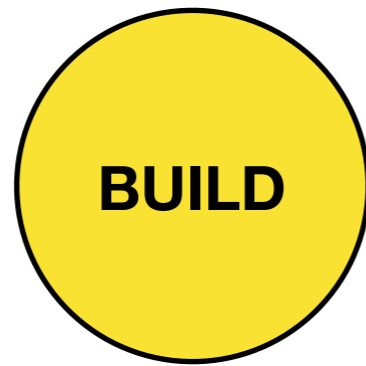
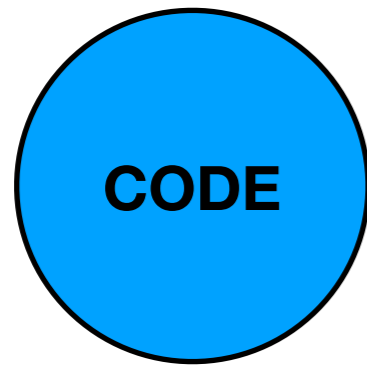
David Allsopp, Stephen Dolan,
Louis Gesbert, Gemma Gordon,
Anil Madhavapeddy (speaker),
KC Sivaramakrishnan
September 2017

With contributions from Nicolas Assouad, Frédéric Bour, Daniel Buenzli, Romain Calascibetta, Amir Chaudhry, Enguerrand Decorne, Jeremie Dimino, Thomas Gazagnaire, Louis Gesbert, Armaël Guéneau, Daniel Hillerström, David Kaloper-Meršinjak, Théo Laurent, Qi Li, Jon Ludlam, Thomas Leonard, Maxime Lesourd, Hannes Mehnert, Richard Mortier, Olivier Nicole, Thomas Refis, David Scott, David Sheets, Mark Shinwell, Magnus Skjægstad, Liang Wang, Leo White, Jeremy Yallop,
and members of the INRIA Galium team, OCamlPro and Jane Street.



tl;dr

- OCaml Labs is 5 years old today!
- This work is based on a huge collaborative effort
- First some **perspective on the past five years**
- Then a **status update on 2017's developments**
- And finally a **sneak preview of 2018 work**

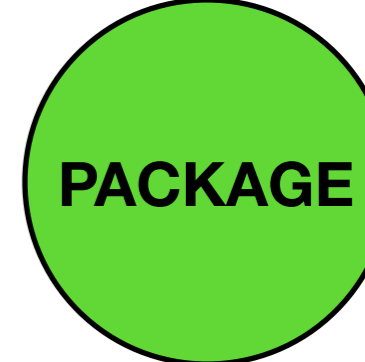
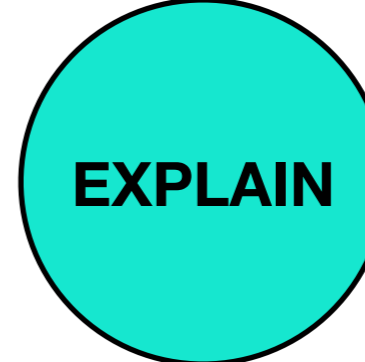
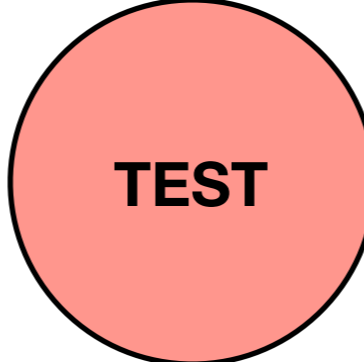
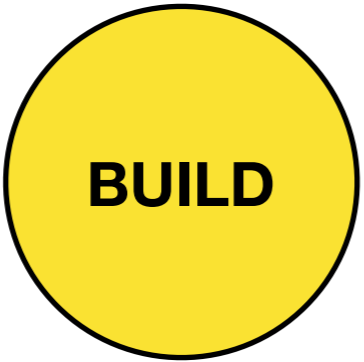
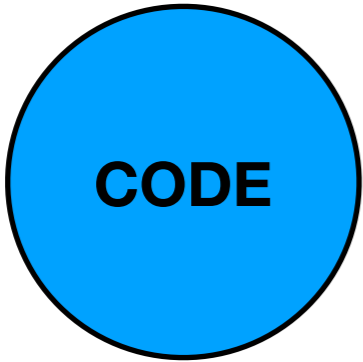


What is the Platform?

The OCaml Platform combines the OCaml compiler toolchain with a coherent set of tools for build, documentation, testing and IDE integration.

The project is a collaborative effort across the OCaml community, tied together by the OCaml Labs group in Cambridge, and OCamlPro in Paris.

The requirements of the Platform are guided by large industrial users such as Jane Street, Citrix, Docker, Facebook, Microsoft and LexiFi, as well as accrued feedback from the opam project.

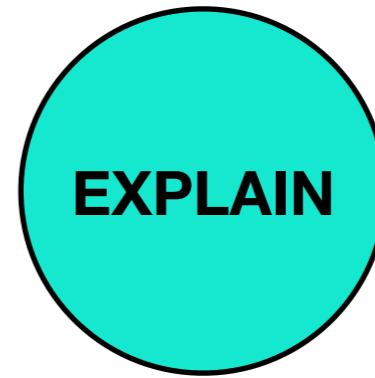
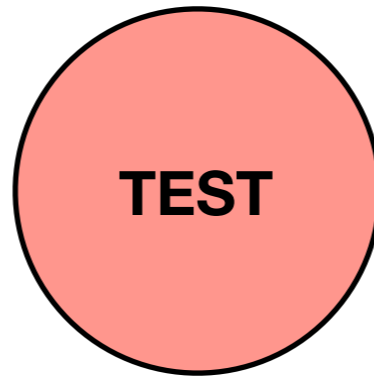
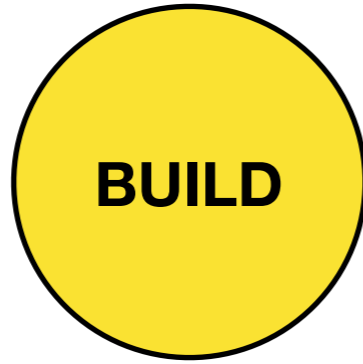
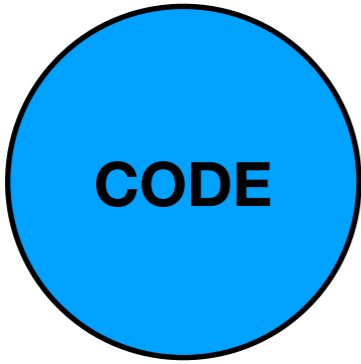


	CODE	BUILD	TEST	EXPLAIN	PACKAGE
<2008	ocamlfind	omake		ocamldoc	godi
2009			ounit		odb
2010		oasis			
2011	mirage				
2012					

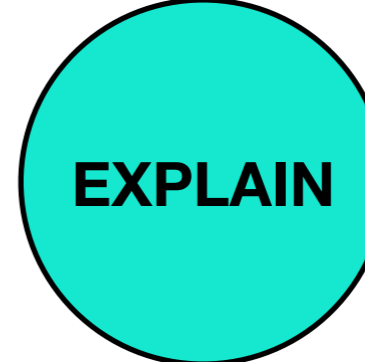
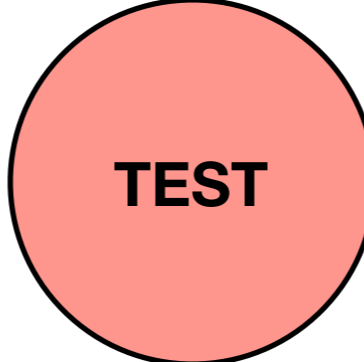
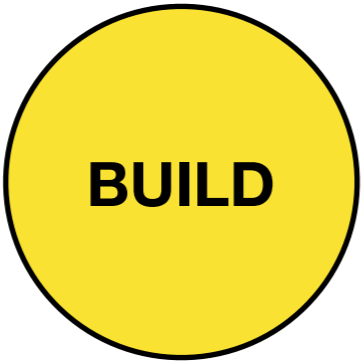
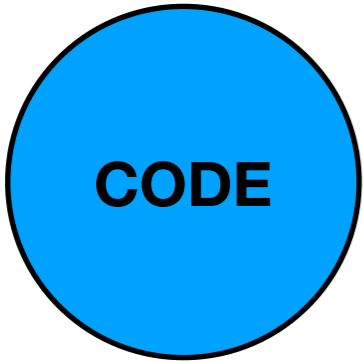
The wilderness years

Every project an island

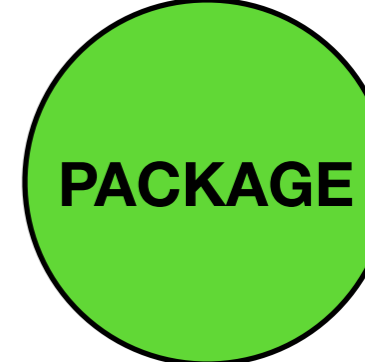
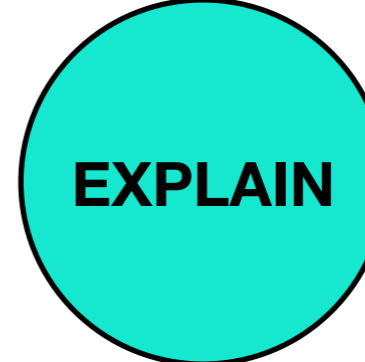
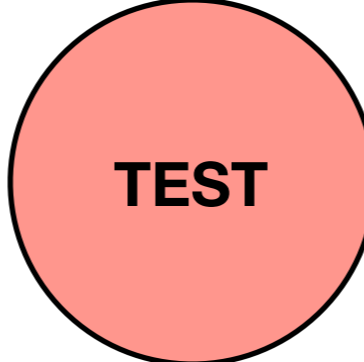
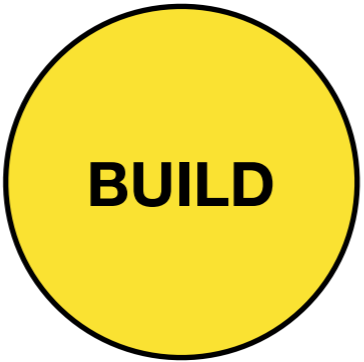
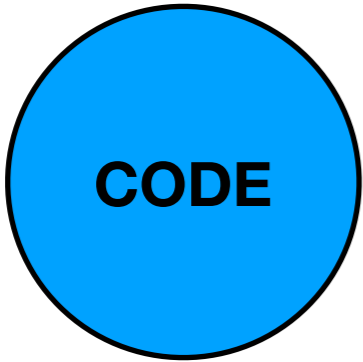
3 hour CUFP "tutorial"



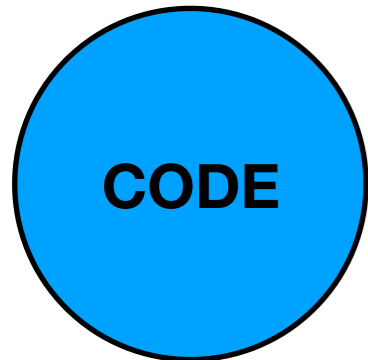
	CODE	BUILD	TEST	EXPLAIN	PACKAGE
<2008	ocamlfind	omake		ocamldoc	godi
2009			ounit		
2010	OCaml Labs founded		A surge of development		Shift to smaller libs
2011					
2012					
2013	merlin		ocamlot	RWO codoc	opam 1.0
2014	irmin ctypes	assemblage		ocaml.org	
2015			docker	AFP course	
2016	git		datakit-ci	ocamlabs.io	topkg
2017		jbuilder	crowbar	odoc odig	opam 2.0



<2008	ocamlfind	omak	Disaggregation & coevolution	amldoc	godi
2009					
2010		oasis			
2011	cmt[i] files		break out components		distribute via opam
2012					
2013	merlin		ocamlot	RWO codoc	opam 1.0
2014	irmin ctypes	assemblage		ocaml.org	
2015		ocamlbuild	docker	AFP course	
2016	git		datakit-ci	ocamlabs.io	topkg
2017		jbuilder	crowbar	odoc odig	opam 2.0



	CODE	BUILD	TEST	EXPLAIN	PACKAGE
<2008	ocamlfind	omake		ocamldoc	godit
2009			ounit		
2010		oasis			
2011	mirage				
2012					
2013	merlin		ocamlot	RWO codoc	opam 1.0
2014	"One in one out"	assemb	Sustainable ecosystem	o	Consistent interfaces
2015				AF	
2016			ocaml-ci	ocamlabs.io	topkg
2017	ppx	jbuilder	crowbar	odoc odig	opam 2.0



camlp4 -> PPX

- PPX are extension points and annotations for OCaml.
- **Key PPX libraries now hosted in one place:**
 - github.com/ocaml-ppx
 - more robust to OCaml compiler versions (ocaml-migrate-parsetree)
- **Getting rid of camlp4 is taking years.**
 - Steady education process required for community.
 - Migration guides on discuss.ocaml.org would accelerate process.
 - e.g. pa_ulex -> sedlex

CODE

Merlin 3.0

- **Major Merlin 3.0 developer tool release**
 - Scalable protocol to communicate with IDEs
 - Robust Windows support.
 - Now promoted to **<https://github.com/ocaml/merlin>**
- **Community now using it as a standard for IDEs**
 - Visual Studio Code, Atom, Sublime Text
 - Facebook Reason syntax support also.
 - More sophisticated short paths algorithm than upstream.



EXPLAIN

Documentation

odoc: generate HTML for a group of libraries, with cross referencing.

By name

A B C D E F G H I J L M N O P R S T U V W X Y Z

[alcotest](#) 0.8.0

[angstrom](#) 0.6.0

[anycache](#) 0.6.0

[arp](#) 0.2.0

[asetmap](#) 0.8.1

[asl](#) 0.11

[asn1-combinators](#) 0.1.3

[astring](#) 0.8.3

[async](#) v0.9.0

[async_extra](#) v0.9.0

[async_find](#) v0.9.0

<http://docs.mirage.io>

```
$ opam install odoc odig
$ opam install mirage # and anything
$ odig odoc
```

EXPLAIN

Documentation

odoc: generate HTML for a group of libraries, with cross referencing.

By name

A B C D E F G H I J L M N O P R S T U V W X Y Z

[alcotest](#) 0.8.0

[angstrom](#) 0.6.0

[anycache](#) 0.6.0

[arp](#) 0.2.0

[asetmap](#) 0.8.1

[asl](#) 0.11

[asn1-combinators](#) 0.1.3

[astring](#) 0.8.3

[async](#) v0.9.0

[async_extra](#) v0.9.0

[async_find](#) v0.9.0

<http://docs.mirage.io>

Now working on docs.ocaml.org - can we build this at a large scale for every package?

EXPLAIN

Documentation

[Top]
Quick start
Tutorial
Execution model
Library guide
Fundamentals
 Promises
 Resolving
 Callbacks
Rejection
Concurrency
 Multiple wait
 Racing
Cancellation
Convenience
 Callback helpers
 Infix operators
 Unallocated promises

Lwt API Reference

`module Lwt`

Asynchronous programming with promises.

Promises are placeholders for values that take a long time to compute. Promises are similar to `refs` – they can store one value. Here is how they differ from `refs`:

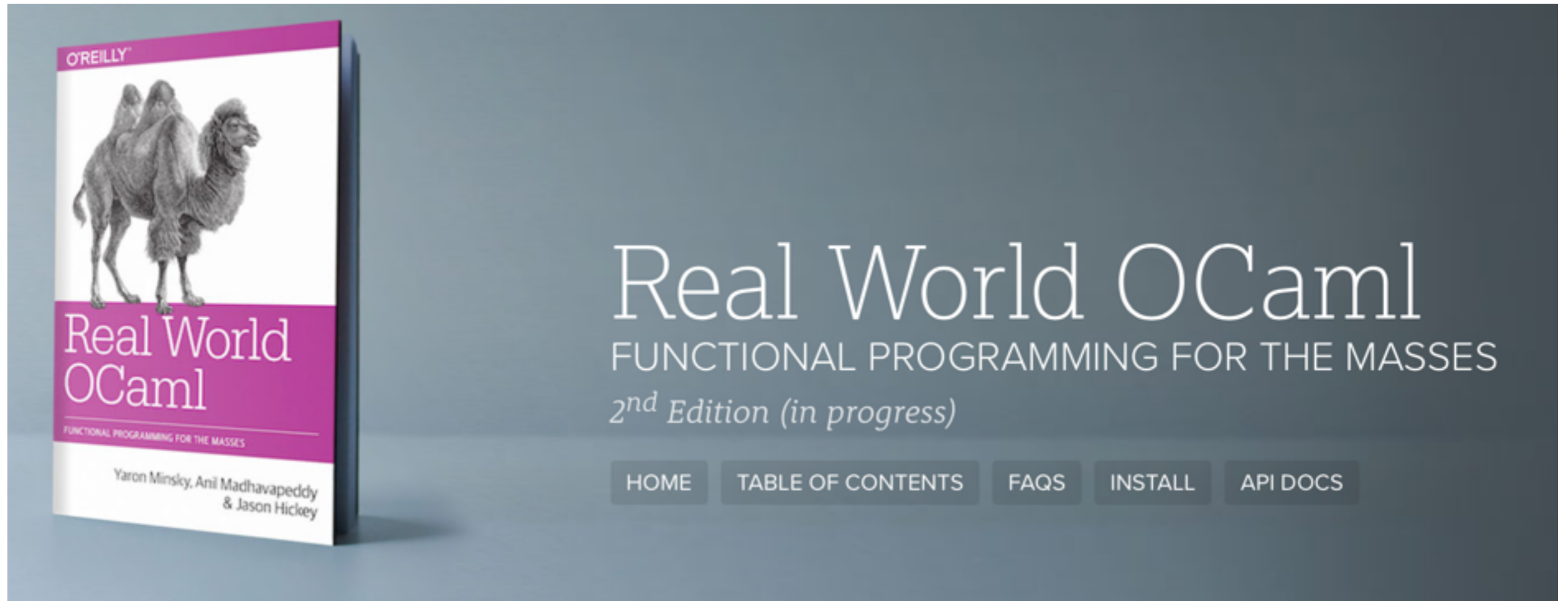
- A promise might not have a value yet. This is called a *pending* promise.
- Writing a value into a promise is called *resolving* it.
- It's possible to attach **callbacks** to a pending promise. They will be run when the promise gets resolved.
- A promise can be resolved only once. Once a promise has a value, it's **immutable**.

<https://ocsigen.github.io/lwt/manual-draft/Lwt.html>

A few remaining issues to fully deprecate
ocaml-doc: porting complex sites like Lwt

EXPLAIN

Documentation



Ongoing refresh at dev.realworldocaml.org



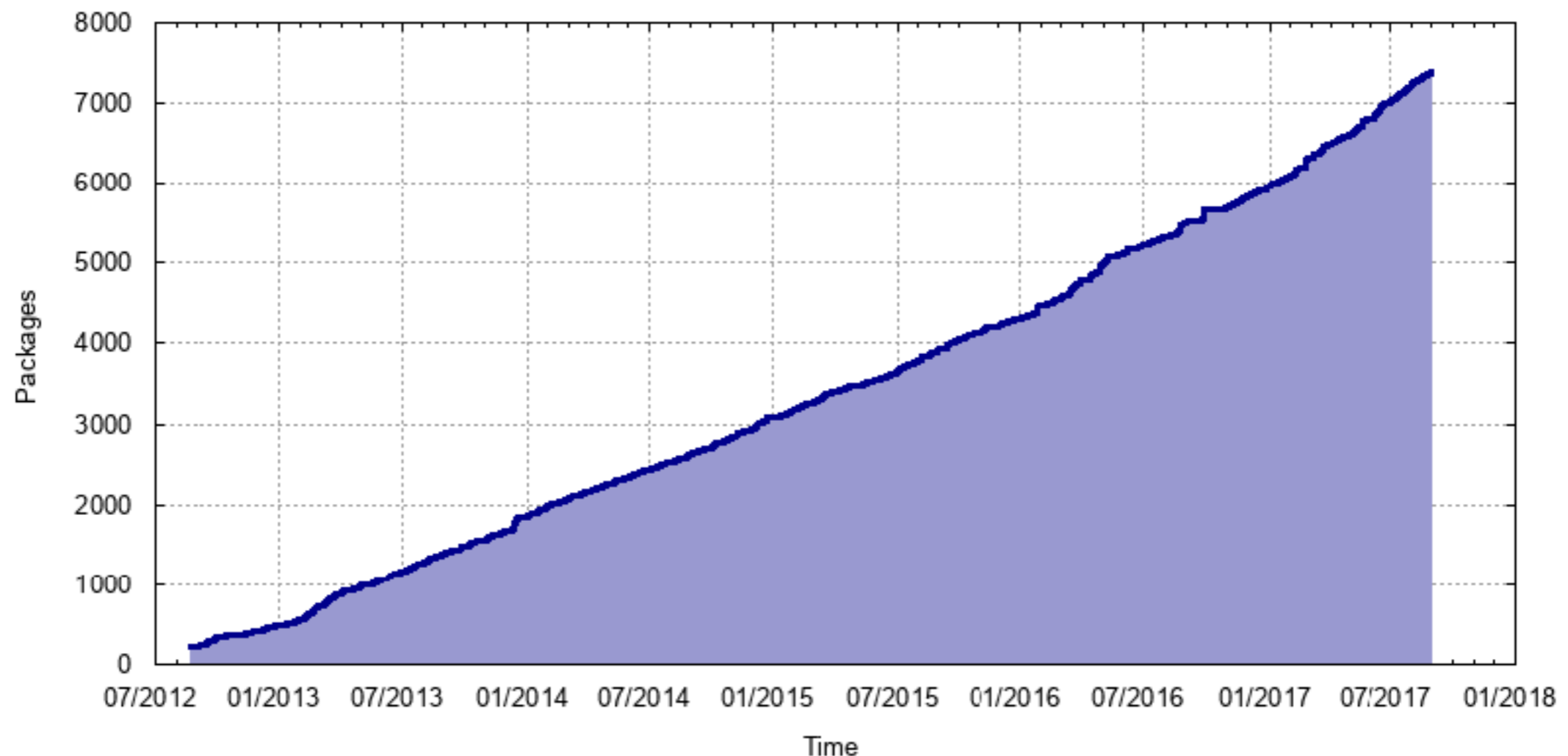
OPAM 2.0beta

- OPAM is the source-based package manager for publishing OCaml code
- Focus this year has been on stabilising the upcoming 2.0
- **Crossed PR #10000 in opam-repository and PR #3000 in opam!**



OPAM 2.0beta

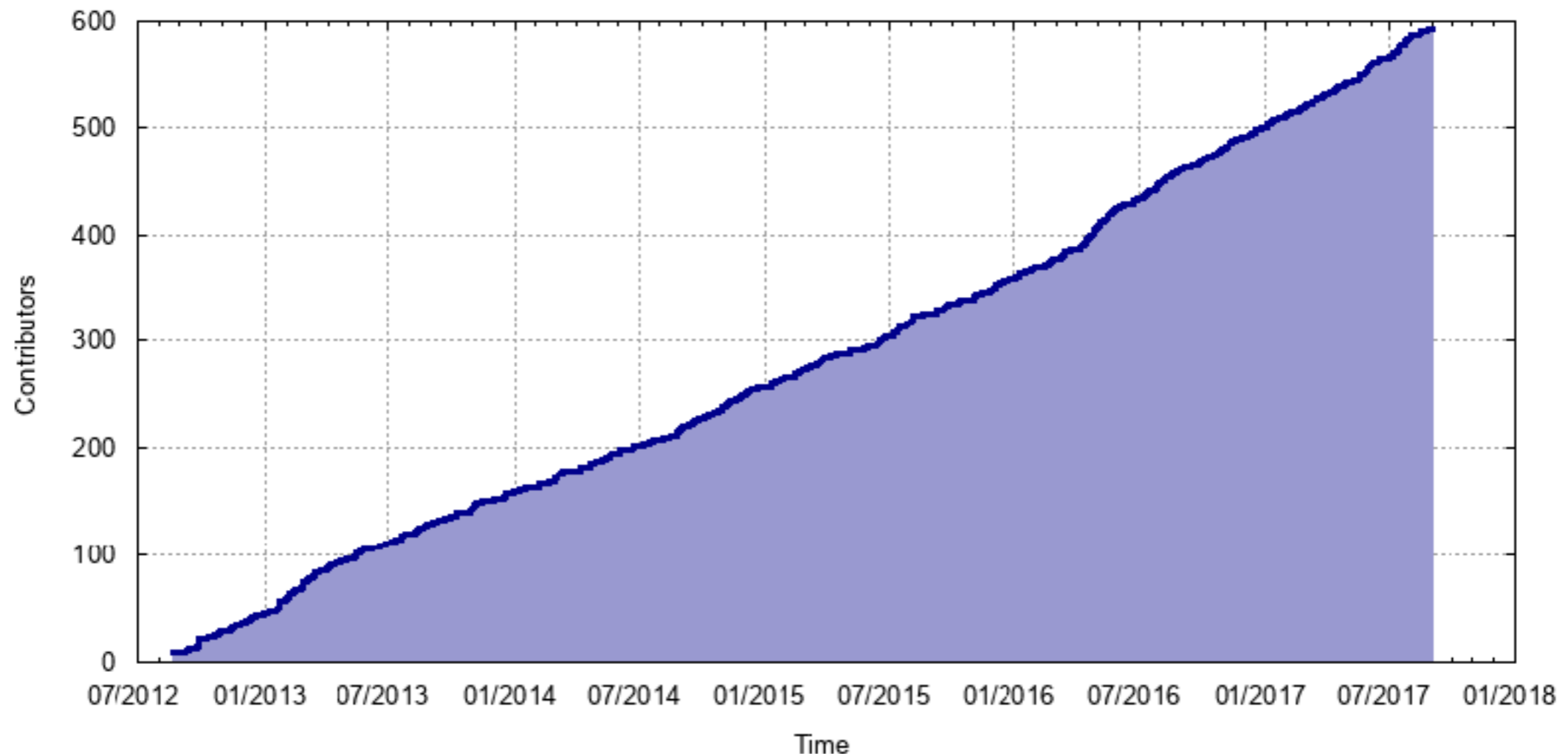
- OPAM is the source-based package manager for publishing OCaml code
- Focus this year has been on stabilising the upcoming 2.0
- **Over 7000 packages now managed**





OPAM 2.0beta

- OPAM is the source-based package manager for publishing OCaml code
- Focus this year has been on stabilising the upcoming 2.0
- **Over 600 individual contributors to the repository**





OPAM 2.0beta

- **New features in recent betas:**
 - more expressive package dependencies

```
depends: [ "foo" {>= "3.0" & < "4.0~" & os = "linux"} ]
```

```
depends: [ "ocaml" >= "4.05.0" ]
```



OPAM 2.0beta

- **New features in recent betas:**
 - more expressive package dependencies

```
depends: [ "foo" {>= "3.0" & < "4.0~" & os = "linux"} ]
```

```
depends: [ "ocaml" >= "4.05.0" ]
```

```
depends: [  
  ...  
  "datakit-server" {>= "0.9.0"}  
  "datakit-client" {with-test & >= "0.9.0"}  
  "datakit-github" {with-test & >= "0.9.0"}  
  "alcotest" {with-test & >= "0.7.0"}  
]
```



OPAM 2.0beta

- **New features in recent betas:**
 - more expressive package dependencies
 - computed versions to make multiple packages easier

```
depends: [  
  ...  
  "datakit-server" {>= version}  
  "datakit-client" {with-test & >= version}  
  "datakit-github" {with-test & >= version}  
  "alcotest" {with-test & >= version}  
]
```

Simplifies the package repository by making it easier to have many small libraries



OPAM 2.0beta

- **New features in recent betas:**
 - more expressive package dependencies
 - computed versions to make multiple packages easier
 - local switches for use per-project

```
cd ~/src/project
opam switch create ./

...

opam list
```



OPAM 2.0beta

- **New features in recent betas:**
 - more expressive package dependencies
 - computed versions to make multiple packages easier
 - local switches for use per-project
- **Solver integrated as a library, no external solver needed!**



OPAM 2.0beta

- **New features in recent betas:**
 - more expressive package dependencies
 - computed versions to make multiple packages easier
 - local switches for use per-project
- **Solver integrated as a library, no external solver needed!**
- **Windows support is being upstreamed!**



OPAM 2.0beta

- **New features in recent betas:**
 - more expressive package dependencies
 - computed versions to make multiple packages easier
 - local switches for use per-project
- **Solver integrated as a library, no external solver needed!**
- **Windows support is being upstreamed!**

But how do we get a usable package repository?
Need a simpler build system

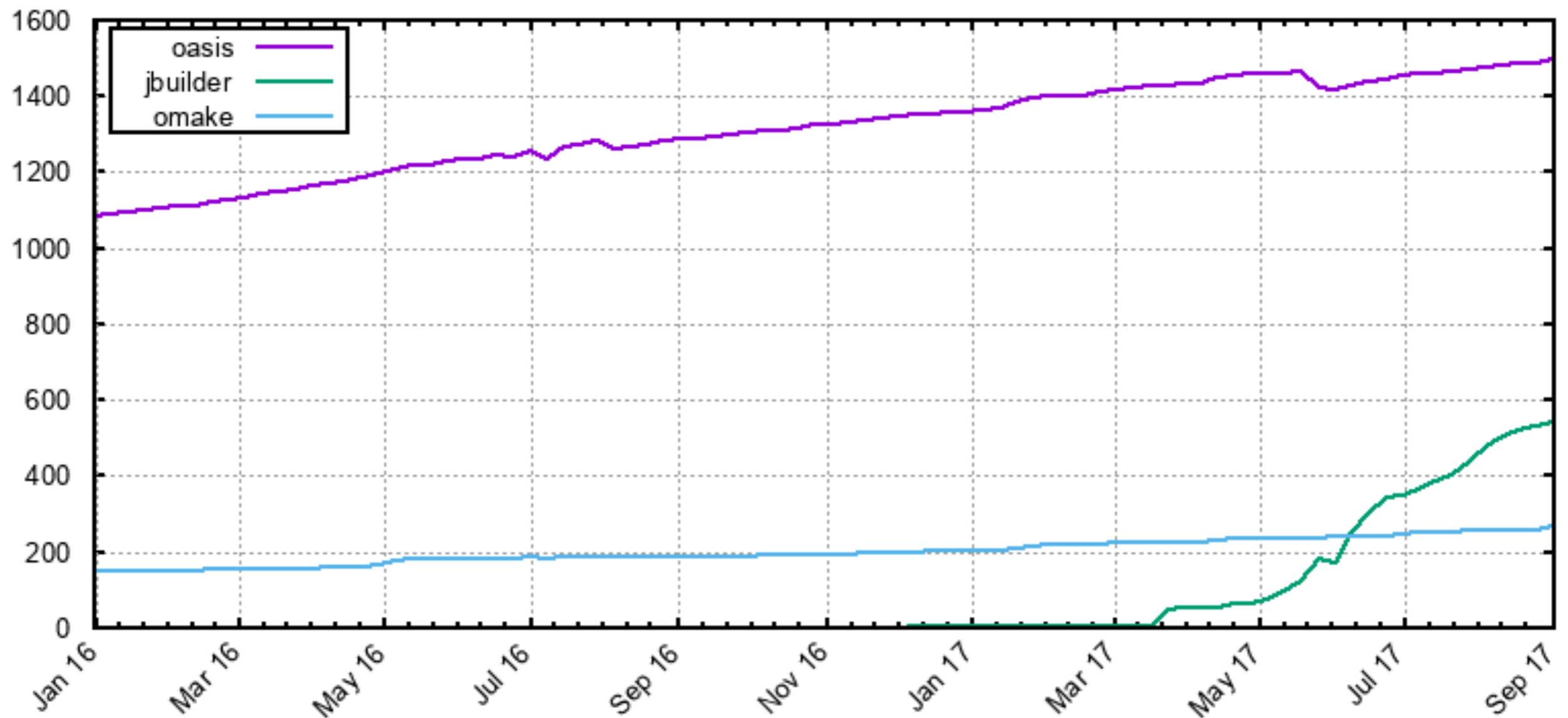
Luckily 2017 has been kind to build...

BUILD

Jbuilder

A build system specialised to real world OCaml code. Provide a description of your project, and it will be built!

Number of packages by build system (all versions)





BUILD

Jbuilder

A build system specialised to real world OCaml code. Provide a description of your project, and it will be built!

- **Compose multiple checkouts** in subdirs and it can be built in one pass
- **Integrates with opam** to locate external packages
- **Multiple workspaces** to support different OCaml versions or build options (e.g. afl or flambda)
- **Declarative model** encourages portable build rules, so it often "just works" on Windows.
- **Fast. Really fast.**

OPAM 2.0 release

Why isn't it released yet?

I want these cool features!

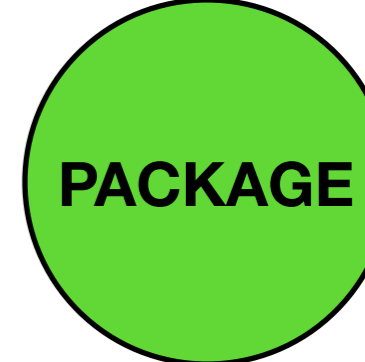
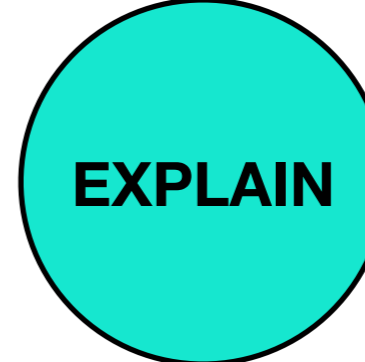
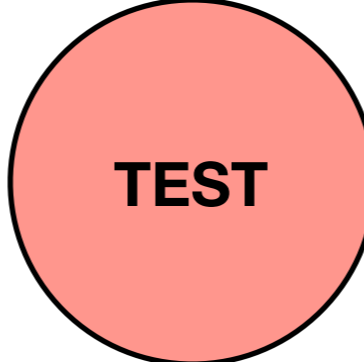
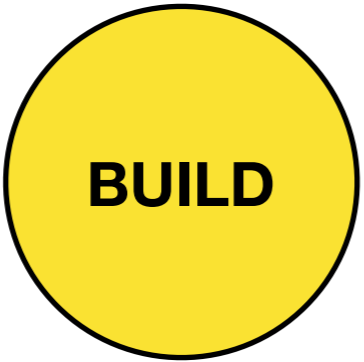
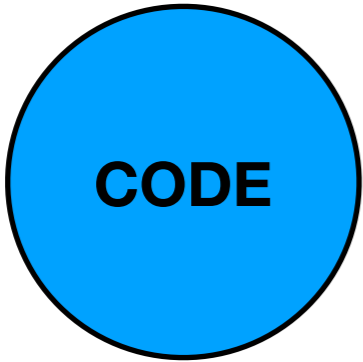


TEST

Continuous Integration

- 2017 has been a good year: big releases to opam, significantly improved CI to submitted packages
- Automated infrastructure is very efficient vs engineering time, so scripting everything we can!
- Currently expanding to meet new needs: **benchmarking**, **fuzzing**, and **portability**
- **Also need to automate the opam1 -> opam2 transition**

Please join the opam maintenance team!



	CODE	BUILD	TEST	EXPLAIN	PACKAGE
<2008	ocamlfind	omake		ocamldoc	godit
2009			ounit		
2010		oasis			
2011	mirage				
2012					
2013	merlin		ocamlot	RWO codoc	opam 1.0
2014	"One in one out"	ocaml	Sustainable ecosystem	ocaml	Consistent interfaces
2015		AFF			
2016		ocamlabs.io			
2017	ppx	jbuilder	crowbar	odoc odig	opam 2.0

travis

**visual
studio**

github

odoc

topkg

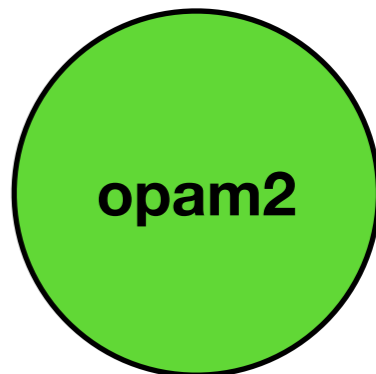
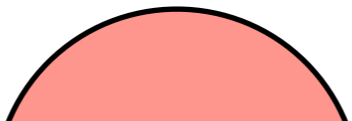
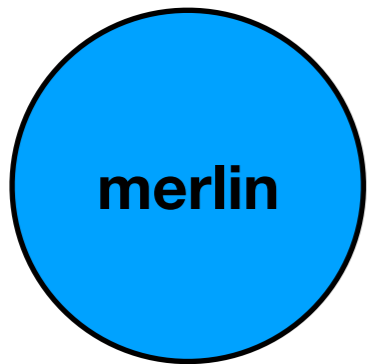
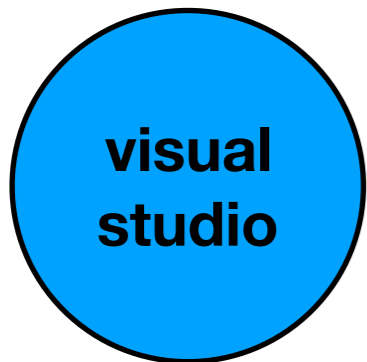
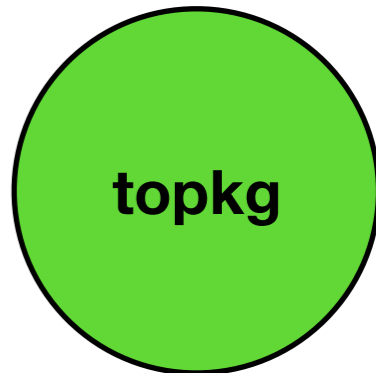
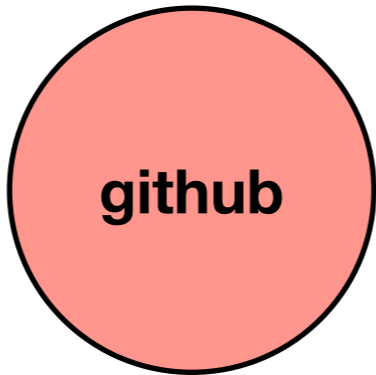
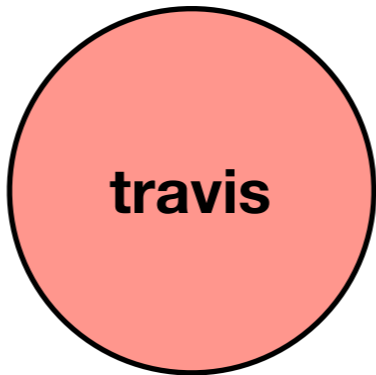
merlin

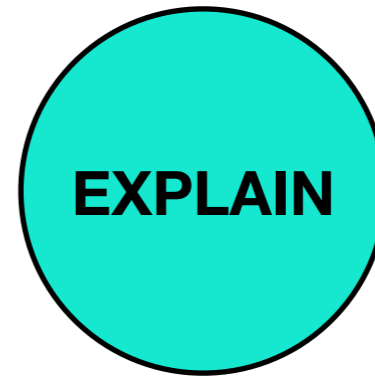
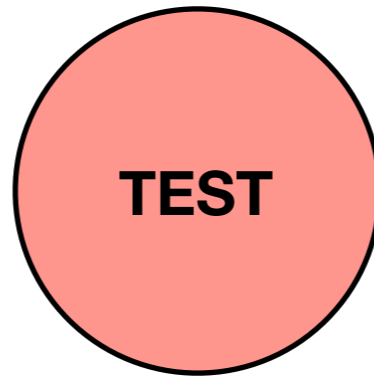
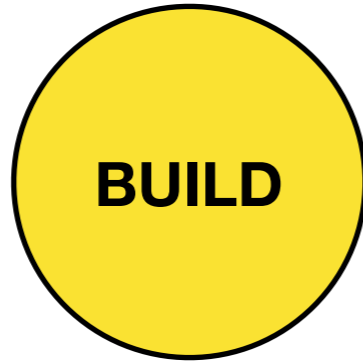
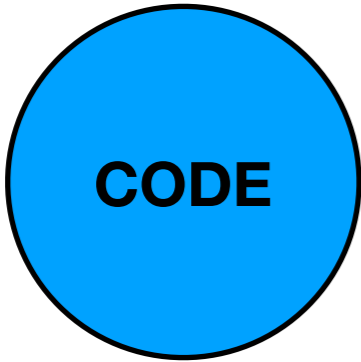
jbuilder

crowbar

odig

opam2





2013

merlin

assemblage

ocamlot

RWO codoc

opam 1.0

2014

irmin
ctypes

ocaml.org

2015

Work in Progress

2016

jbuild

hllabs.io

topkg

2017

git

crowbar

odoc
odig

opam 2.0

2018

codename: bob

2019

2020

2021

2022





an aside: cargo/rust

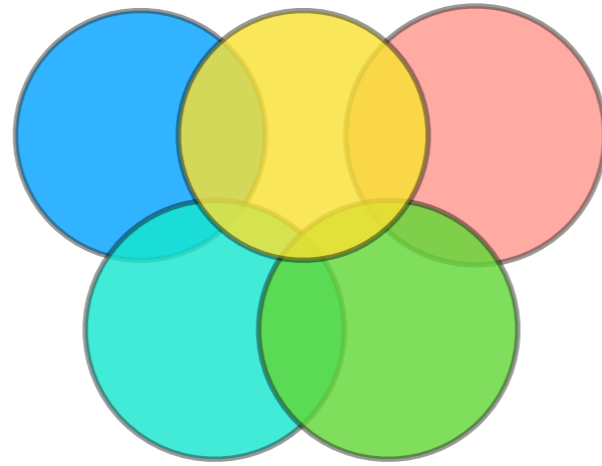
```
Some common cargo commands are (see all commands with --list):
build      Compile the current project
check      Analyze the current project and report errors
clean      Remove the target directory
doc        Build this project's and its dependencies' docs
new        Create a new cargo project
init       Create a new cargo project in an existing directory
run        Build and execute src/main.rs
test       Run the tests
bench      Run the benchmarks
update     Update dependencies listed in Cargo.lock
search     Search registry for crates
publish    Package and upload this project to the registry
install    Install a Rust binary
```

Single binary, gives access to full Rust ecosystem
Can we adopt same approach for Platform?



codename: bob

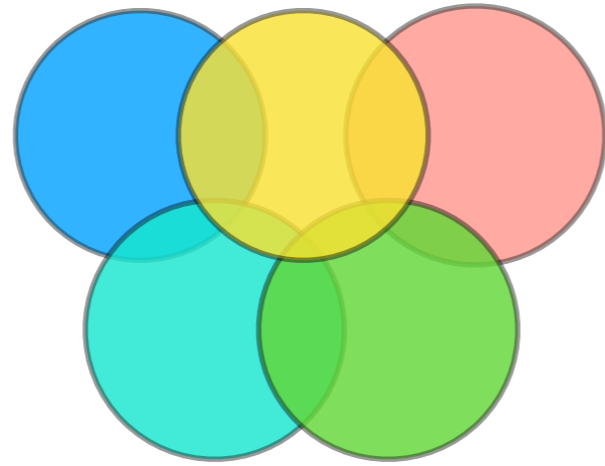
```
Some common bob commands are (see all commands with --list):
build      Compile the current project
check      Analyze the current project and report errors
clean      Remove the target directory
doc        Build this project's and its dependencies' docs
new        Create a new OCaml project
init       Create a new OCaml project in an existing directory
run        Build and execute src/main.ml
test       Run the tests
fuzz       Run the fuzzer
compat     Check compatibility across different operating systems
bench      Run the benchmarks
update     Update dependencies listed in opam.lock
search     Search registry for opam packages
publish    Package and upload this project to the opam registry
install    Install an OCaml binary
```



bob

- **Single CLI tool for all OCaml platform activities.**
 - Includes `jbuilder`, `odoc`, `merlin`, `opam`, `alcotest`, `crowbar`, `topkg` with a single CLI
 - Sensible defaults, with escape hatches for advanced users
- **Access to offline/online infrastructure:**
 - dedicated **continuous integration** access
 - **consistent OS support** for Linux, Windows, macOS
 - **search online** community (`odoc`, `opam`, `github`)

Will be released in 2018 as the first OCaml Platform release, with associated tools bundled



bob

- Bob is **not yet complete** or even fully designed
- We are starting an open design process this year, as we release opam 2.0
- opam maintenance window is around ~3-5 years, so want to get this revision right

Online Community

`ocaml.org`

`opam.ocaml.org`

`realworldocaml.org`

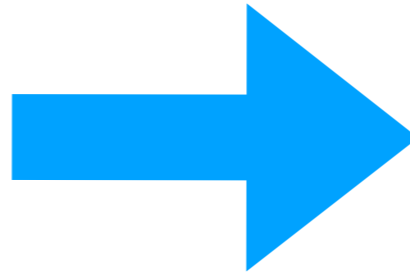
`lists.ocaml.org`

`discuss.ocaml.org`

`ci.ocaml.io`

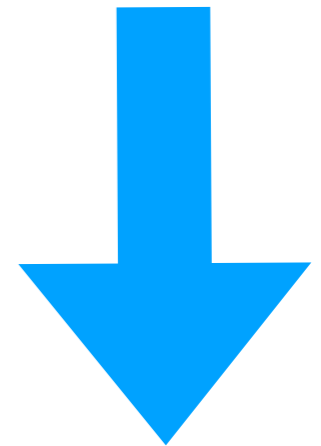
`docs.ocaml.org`

`github.com/ocaml`

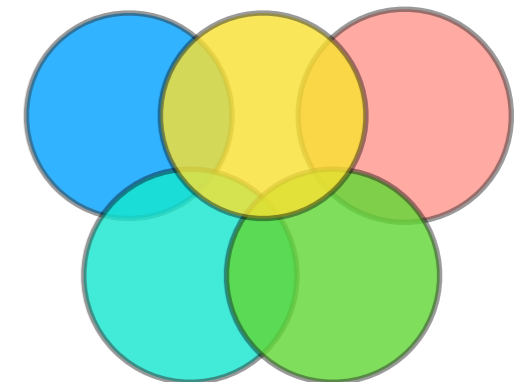


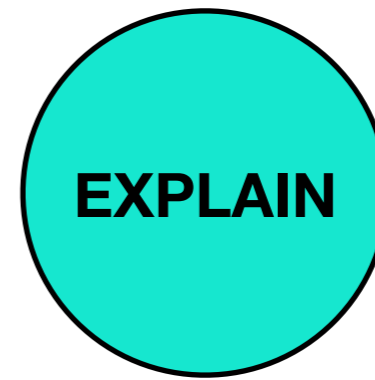
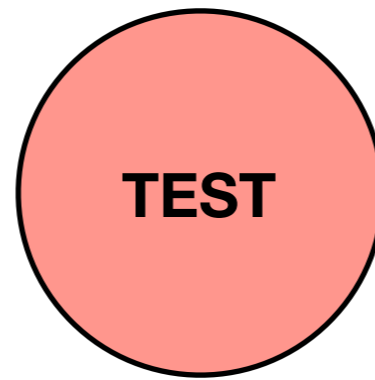
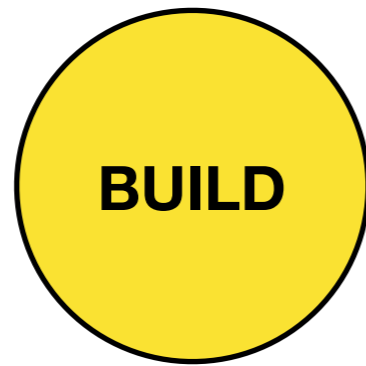
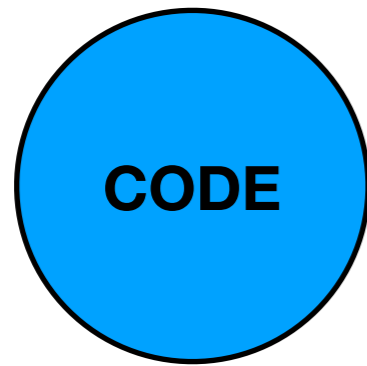
Unify the design of these sites, with ocaml.org as the main public site

Create searchable API for all online OCaml resources - used by bob search



Category	Items	Flags	News	Activity
Community	30	10	140	
Tools	14	110	15	
Industry	4	140	15	
Tools	8	110	20	
Community	10	100	20	
Tools	9	80	10	
Search	5	21	10	





What is the Platform?

The OCaml Platform combines the OCaml compiler toolchain with a coherent set of tools for build, documentation, testing and IDE integration.

The project is a collaborative effort across the OCaml community, tied together by the OCaml Labs group in Cambridge, and OCamlPro in Paris, **and the whole OCaml community.**

See you on discuss.ocaml.org :)