



ECNU

# Julia 中文社区冬季会议-2022

## OSPP-22 · 二维码编解码器

报告人 王至宏

华东师范大学 计算机科学与技术学院

2022年12月09日

# 概要

CONTENTS

**01. 十分钟重新认识二维码**

**02. OSPP & Julia 项目经历**

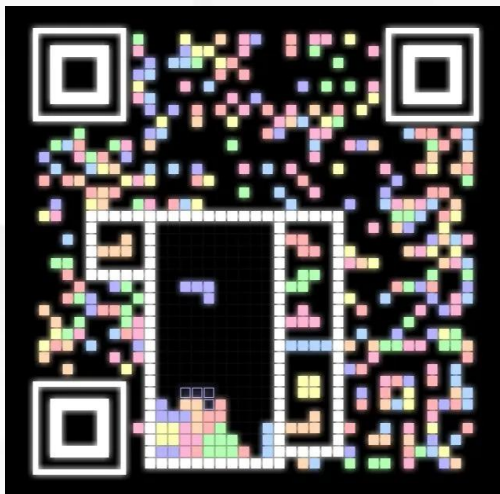
# 二维码脑洞

微信名片 | 健康码 | 单车码

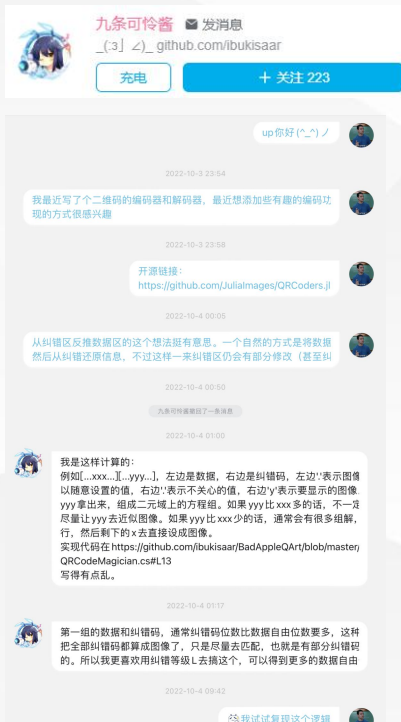
纠错原理 => 做更有创造性的事情



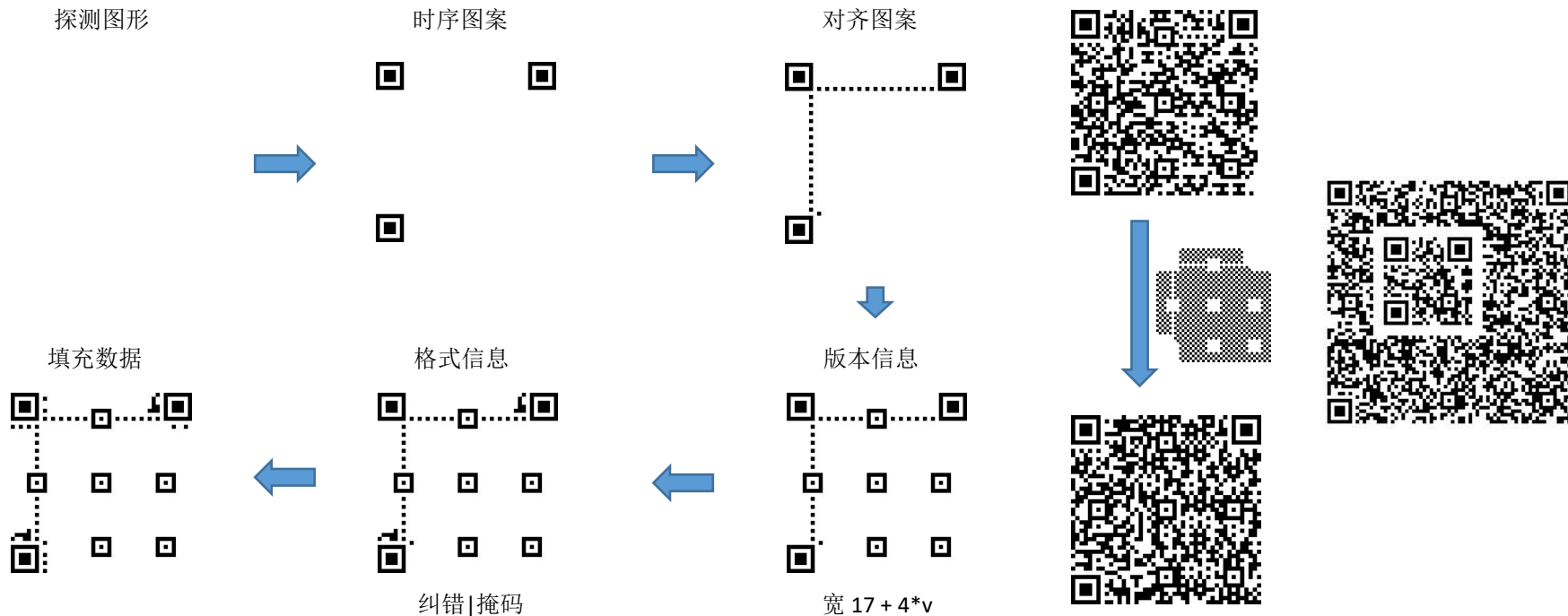
玩游戏-俄罗斯方块



看视频-Where there is a screen there can be *bad apple*

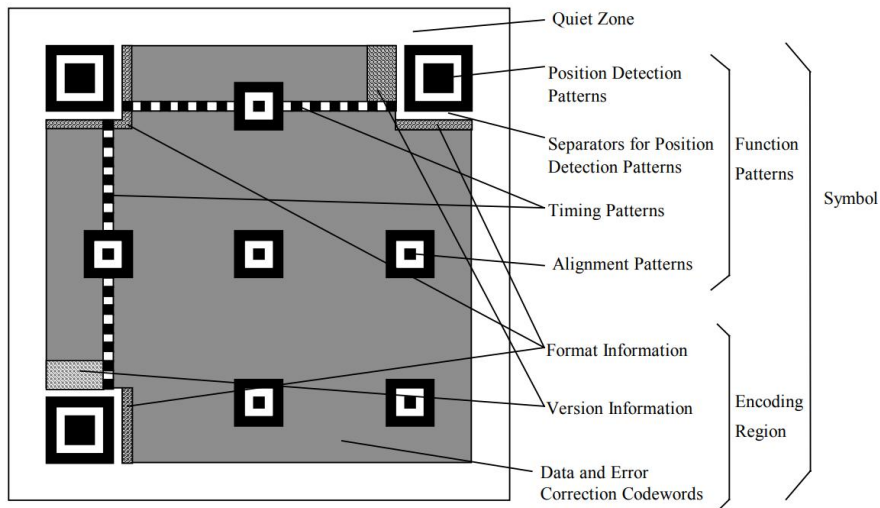


# 二维码是如何构建的？

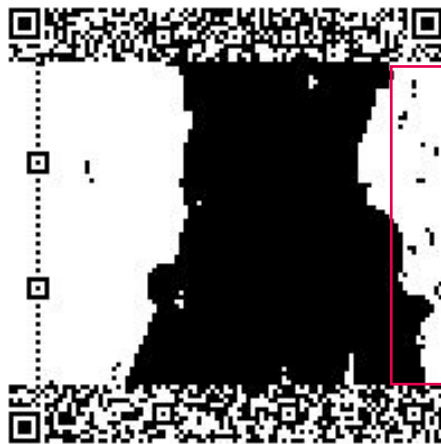


# 数据编码 & 交织分散

字符 -> 比特块 -> 纠错 -> **交错** -> 填充



追踪分散数据  
CartesianIndex + @view



薛定谔的二维码



镜像二维码?

# 二维码是怎么纠错的



Alice

$x \in \{0,1\}$



Bob

$x' = ? \in \{0,1\}$

$y \in \{000,111\}$

$y' = ? \in \{0,1\}^3$



1. 纠错：假设传输过程至多产生 1 比特错误

$$y = 000 \longrightarrow y' = 010 \longrightarrow \begin{aligned} d(y', 000) &= 1 \\ d(y', 111) &= 2 \end{aligned}$$

2. 检错：假设传输过程至多产生 2 比特错误

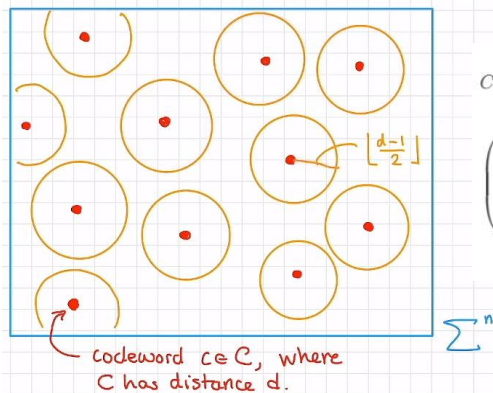
$$y = 000 \longrightarrow y' = 011 \longrightarrow d(y', \{000, 111\}) \neq 0$$

3. 补缺：假设传输过程至多丢失 2 比特信息

$$y = 000 \longrightarrow y' = *0* \longrightarrow \begin{aligned} d(y', 000) &= 0 \\ d(y', 111) &= 1 \end{aligned}$$

编码理论：设  $C$  的编码距离为  $d$

- 编码  $C$  能检测至多  $e \leq d - 1$  个错误
- 编码  $C$  能填补至多  $e \leq d - 1$  个缺失
- 编码  $C$  能纠正至多  $e \leq \lfloor \frac{d-1}{2} \rfloor$  个错误, 即  $2e \leq d - 1$



$$C: \Sigma^4 \rightarrow \Sigma^7$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \mapsto \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_2 + x_3 + x_4 \\ x_1 + x_3 + x_4 \\ x_1 + x_2 + x_4 \end{pmatrix}$$

线性码-MDS 码-BCH码

# 项目产出

## QRcoders.jl

- 性能优化的编码器
- 结构体 `QRCode` 及函数工具
- 二维码绘制图像

## 模块开发 tips:

1. Avoid type piracy
2. `pop,append => Vector{T}(undef,xx) + ind + @view`
3. 适配 1.6, 比如 `[1,;]`
4. 整体规划, 比如结构体设计 `QRInfo` 等

## QRDecoders.jl

- 实现 ReedSolomon Code 的两种解码算法
- 支持简单二维码图片的解码

## 后续计划:

- 基于数据编码的二维码样式
- 基于图像美化的 [二维码样式](#)
- 编解码封闭的系统 => 提升解码器能力



johnnynchen94 on Aug 5

Men

For fast-development purpose this is okay. But we'll need to move these to where it belongs: `QRCode.jl` -- technically, this is type piracy. <https://docs.julialang.org/en/v1/manual/style-guide/#Avoid-type-piracy>

I notice that you've used many `push!`, `pop!` operations. I'm not sure if this will be the performance bottleneck. solution is to pre-allocate the coefficient vector and use `StaticArrays` to further improve it.



# 概要

CONTENTS

01. 十分钟重新认识二维码

02. OSPP & Julia 项目经历



# OSPP & Julia 项目经历



## A More Complicated Example

Here are a few example templates that use the options and plugins explained above.

This one includes plugins suitable for a project hosted on GitHub, and some other customizations:

```
Template({
  user="my-username",
  dir=~"/code",
  authors="Acme Corp",
  julia=v"1.1",
  plugins=[
    License(; name="MPL"),
    Git(; manifest=true, ssh=true),
    GitHubActions(; x86=true),
    Codecov(),
    Documcenter(GitHubActions()),
    Devlop(),
  ],
})
```

Here's one that works well for projects hosted on GitLab:

RexWzh

1 5月7日

我在用 Julia 写包，啃文档前希望能得到些建议，提高效率  
和减少踩坑。

中文论坛上翻到的帖子比较旧，比如

- 19.10 [julia 开发包入门](#)
- 19.08 [Julia开发module的工作流程](#) 2
- 18.08 [如何轻松地编写一个Julia应用包?](#) 2

网上翻到视频资源不多，内容层次不齐（可能姿势不对）

我的学习模式是 **视频/文档 + 读书笔记**，当前在看这两个：

1. [PkgTemplates](#) 3
2. [Documcenter](#) 1

QQ | discourse | slack | zulip

理论学习 | 时间前移 | 毕业季

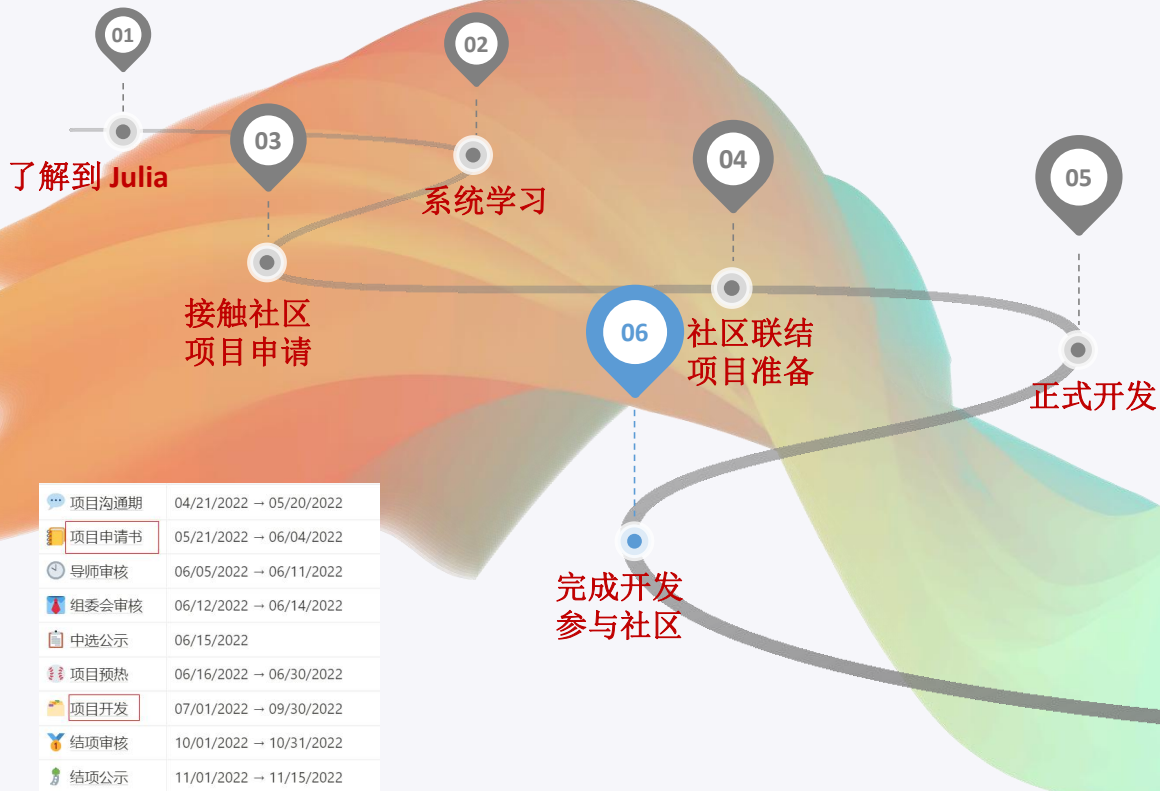
[起步](#) | PkgTemplates | Documcenter

短课程 | 论坛介绍 | Julia 冬季会议

打基础 | 混脸熟 | [LeetCode.jl](#)

导师 | OSPP | ImageMagick.jl | QRCode.jl

# OSPP & Julia 项目经历



add operation for Polynomials and GF integers ✓

#15 by RexWzh was merged on Aug 8

remove Documenter compat ✓

#13 by johnnychen94 was merged on May 1

八九月 | 编解码 | 开发维护 | 社区交流

The background features a low-angle shot of several tall, light-colored classical columns. A large, semi-transparent white diagonal shape cuts across the image from the top-left towards the bottom-right. A solid red triangle is positioned in the bottom-right corner, partially overlapping the white shape and the columns.

感谢您的聆听