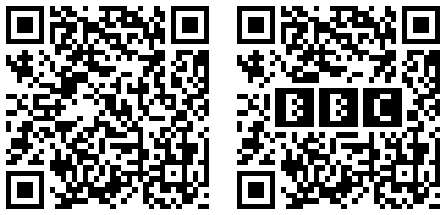
**QArt Codes 技术报告**

二维码是根据一定规则设计的一种二维的点阵画面，用于编码任意的文本字符串。通常用于编码URL链接，实现扫码登陆的功能。

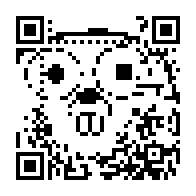
二维码基于Reed-Solomon error-correcting codes原理编码，拥有一定得容错能力，对于特定的编码字符串，并非所以点阵都是严格固定的。因此，二维码的编码原理允许发挥个性，进行改变。

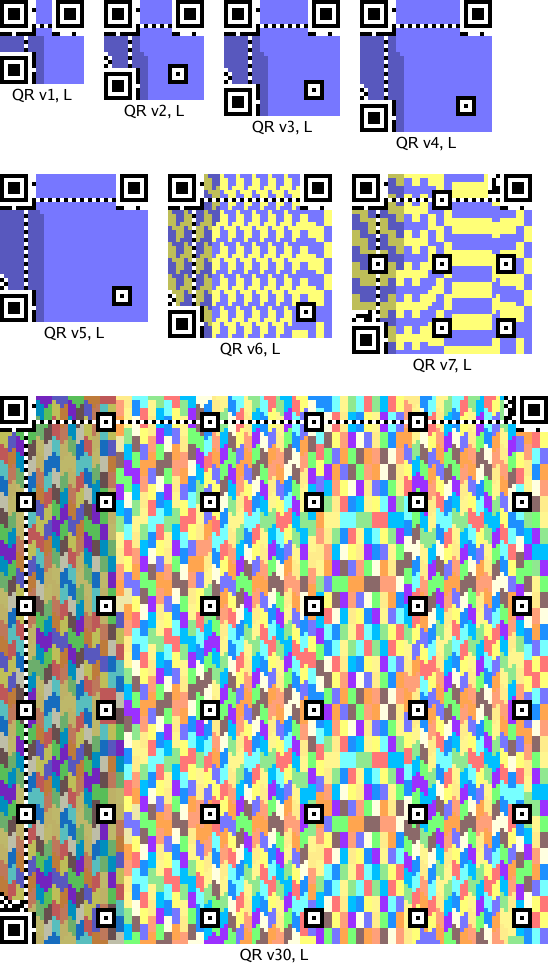
如，左图（“http://bbc.co.uk/programmes”）可以变更成右图，而不会影响编/解码的正确性。



QArt Codes实现个性化定制的的原理就基于此。

Demo如下所示。



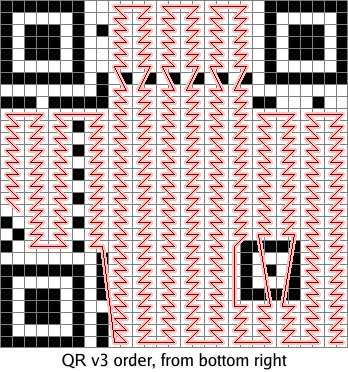


如图所示，彩色像素就是Reed-Solomon-encoded像素位。每个二维码都有若干个如图所示的像素位块，具体取决于错误修正的层级。

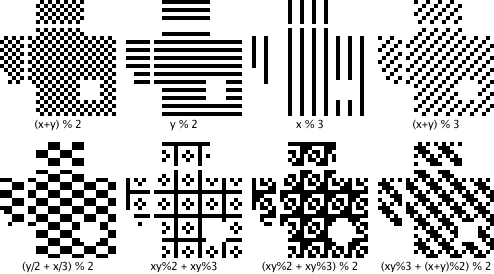
例如：L级别表示最低的纠错冗余度，约20%的纠错冗余码；M约38%，Q约55%，H约65%。



原始信息的数据位被直接用于Reed-Solomon编码，因此被编码数据的每一个位，实质上都对应着二维码中的一个像素。如上图所示，亮色区域就是编码数据位对应的像素区域。而暗色（阴影）部分，就是错误校验码对应的像素区域。

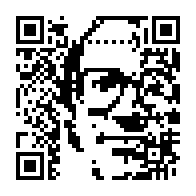


具体的编码顺序如图所示。图示为第三代标准的二维码编码顺序，低位数据从右下角开始编码。



QArt Codes的实质其实是利用二维码具有较多冗余校验码，解码器可以自动实现错误识别和校正修复，最后还原正确的原始信息的原理。QArt Codes一定程度上破坏原始信息数据对应的编码位，根据自定义函数或映射，设置个性化的图案蒙版，根据需要破坏不符合需要的像素位，以此实现自定义、个性化的二维码图案。

QArt Codes其实是与二维码结果设计的初衷相违背的，它破坏了二维码原本的原始数据+冗余校验及错误修正的严格设计规范，而是利用其较高程度的错误冗余设计来实现“破坏式”的编码。



此案例中，灰色阴影区域表示被占用的像素位区域。具体而言，左边的阴影部分是错误校验码的编码区域，右侧的阴影区域则是原始信息占用的编码像素区域。虽然根据自定义的蒙版有所“破坏”，但仍然可以在解码时通过左侧部分的错误校验码实现恢复。因此，这样的二维码仍然是可以正常扫描并被正确识别出来的。