# Kissy Validation 接口设计

## 规则绑定方式（bind）

三种形式将规则绑定到指定的表单字段：

1. 通过属性data-type方式绑定

这种是最简单的绑定方式，适合简单规则的验证。

示例：<input type=”text” id=”email” name=”email” data-type=”email” value=”” />

1. 自定义属性（attribute）方式绑定

自定义属性基本遵循web form2.0规范，常见的属性有requried；min；max等。适合需要往验证函数传入参数的复杂规则的验证。

示例：<input type=”text” id=”age” name=”age” data-type=”digits” min=”12” value=””/>

1. 静态（static）规则绑定

即在初始化验证类时通过配置参数rules传入的规则。

优先级顺序为：静态规则 ＞ 自定义属性规则 ＞ data-type规则。

## 内置验证规则（rules）

1. **required**

必填规则。

绑定方式：【data-type|attribute|static】

参 数：【Selector|Function|Boolean】（optional）

1. **equalTo**

值相等验证

绑定方式：【attribute|static】

参 数：【Selector|HTMLElement】

1. **email**

电子邮件格式验证

绑定方式：【data-type|static】

参 数：none

1. **url**

url格式验证

绑定方式：【data-type|static】

参 数：none

1. **number**

10进制数字格式验证（包涵浮点数）

绑定方式：【data-type|static】

参 数：none

1. **digits**

整数格式验证

绑定方式：【data-type|static】

参 数：none

1. **alpha**

字母格式验证

绑定方式：【data-type|static】

参 数：none

1. **mobile**

手机号码格式验证

绑定方式：【data-type|static】

参 数：none

1. **postcode**

邮编格式验证

绑定方式：【data-type|static】

参 数：none

1. **min**

最小数字验证

绑定方式：【atribute|static】

参 数：【Number】

1. **max**

最大数字验证

绑定方式：【atribute|static】

参 数：【Number】

1. **range**

最大数字验证，需配合min规则和max规则一起使用

绑定方式：【data-type |static】

参 数：none

1. **minlength**

最短字符长度验证

绑定方式：【atribute|static】

参 数：【Number】

1. **maxlength**

最长字符长度验证

绑定方式：【atribute|static】

参 数：【Number】

1. **rangelength**

字符长度范围验证，需配合minLength规则和maxLength规则一起使用

绑定方式：【data-type |static】

参 数：none

1. **pattern**

自定义验证正则

绑定方式：【atribute|static】

参数：【Regexp】

## 原型方法（publish）

* **validate (**element**)**

//手动验证，掉用该方法直接执行表单所有控件的验证。如果传入了参数element，则只验证elemen控件的数据。对应JQuery Validator的form和element方两个公开方法

//@param {String | HTMLElement}

//@return {Validator}

* **reset()**

//重置验证器，等价于JQuery Validator的resetForm公开方法

//@return {Validator}

* **getInvalid()**

//获取所有验证不通过的对象

//@return {Array} [{element:element，message:message}]

* **isValid（）**
* **getRules()**

## 工具方法（utils）/静态属性

* **S.validator.setDefault = function(){}**

//设置默认配置属性

* **S.validator.add = function(){}**

//动态添加一个验证器，等同于JQuery Validator的addMethod 方法。

* **S.validator.format = function(){}**

//简单的替换函数，用于替换字符串中的{0},{1}等。如果kissy core中加入这个功能的话，考虑移除。或者改原型内置，简化接口

* **S.validator. checkable = function(){}**

//判断一个元素是否是可勾选元素，简化接口，考虑取消

## 配置属性（config）

* **rules** //{Object} 绑定静态验证规则
* **message** //{Oject} 自定义提示信息
* **onSubmit** //{Boolean=true} 是否在提交表单是验证
* **autofocus** //{Booean=true}
* **focusCleanup** //{Booean=false 获得焦点后直接清除提示信息
* **isLazy** //{Boolean=false} 是否为懒验证模式，懒验证模式下，当值改变时不会立即触发验证
* **ignoreList** //{Array=[] | HTMLElement} 忽略列表
* **validClass** //{Stirng=”valid”} 验证通过后的样式标记
* **invalidClass** //{Sting=”invalid”} 验证失败后的样式标记
* **messageElement** //{String=”label”} 提示信息元素
* **messageWrapper** //{String=”” } messageElement元素外再包裹一层元素
* **messageContainer** //{String=”” | HTMLElement} 将提示信息在一个区域集中显示

## 自定义事件（event）

* **beforeValidate**

//验证开始前事件。反回false，跳过验证

//@param {Event} e

//@param {HTMLElement} form

* **invalid**

//每个表单控件验证失败后，显示错误信息前触发，若返回false，跳过演示错误信息步骤。

//@param {Event} e

//@param {Object} {element:element，message:message}

* **valid**

//每个表单控件验证通过后触发。仅绑定了验证规则的表单控件才会触发该事件。

//@param {Event} e

//@param { HTMLElement } element

* **validate**

//校验完成后触发，只要订阅了，不论验证通过与否都会触发。

//@param {Event} e

//@param {HTMLElement} form