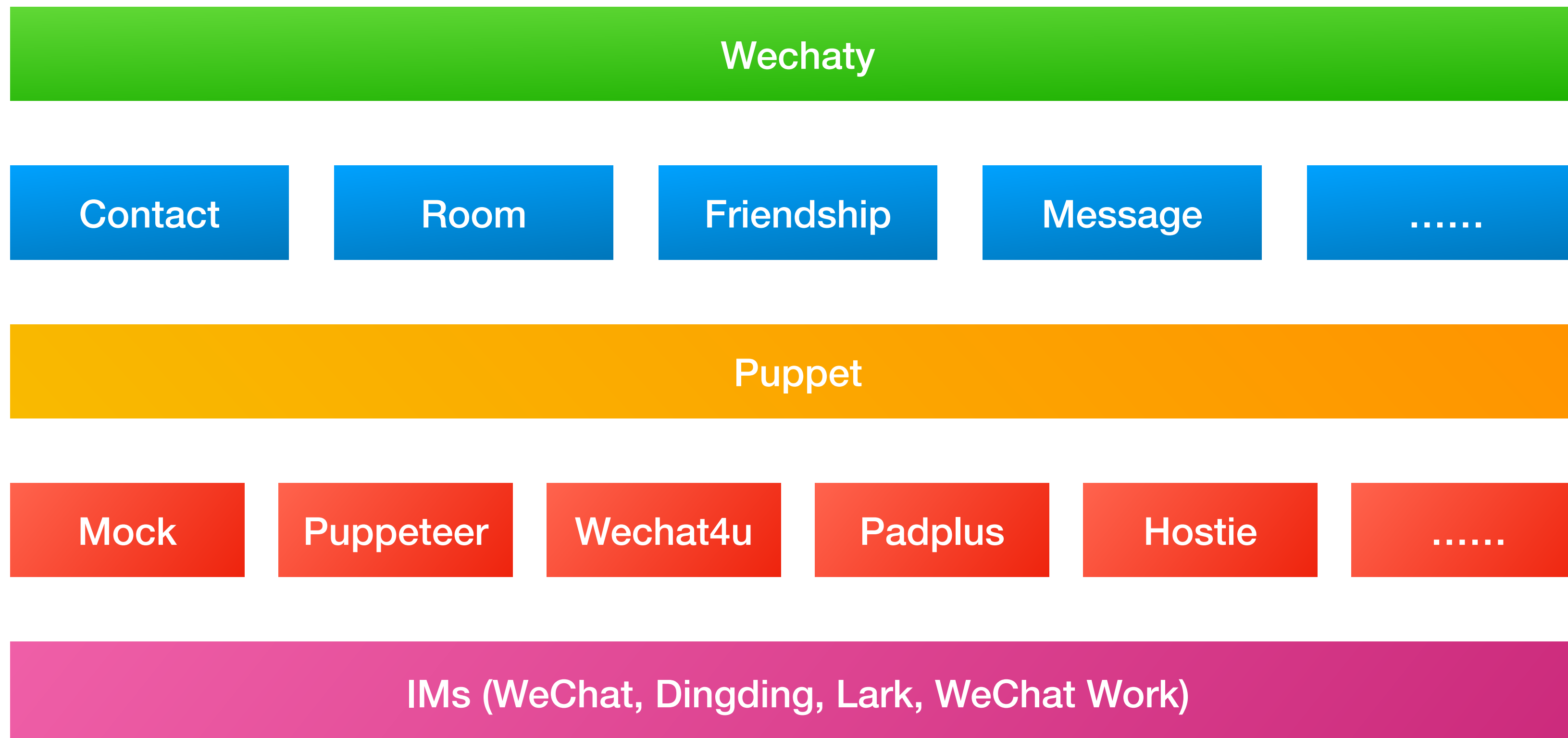


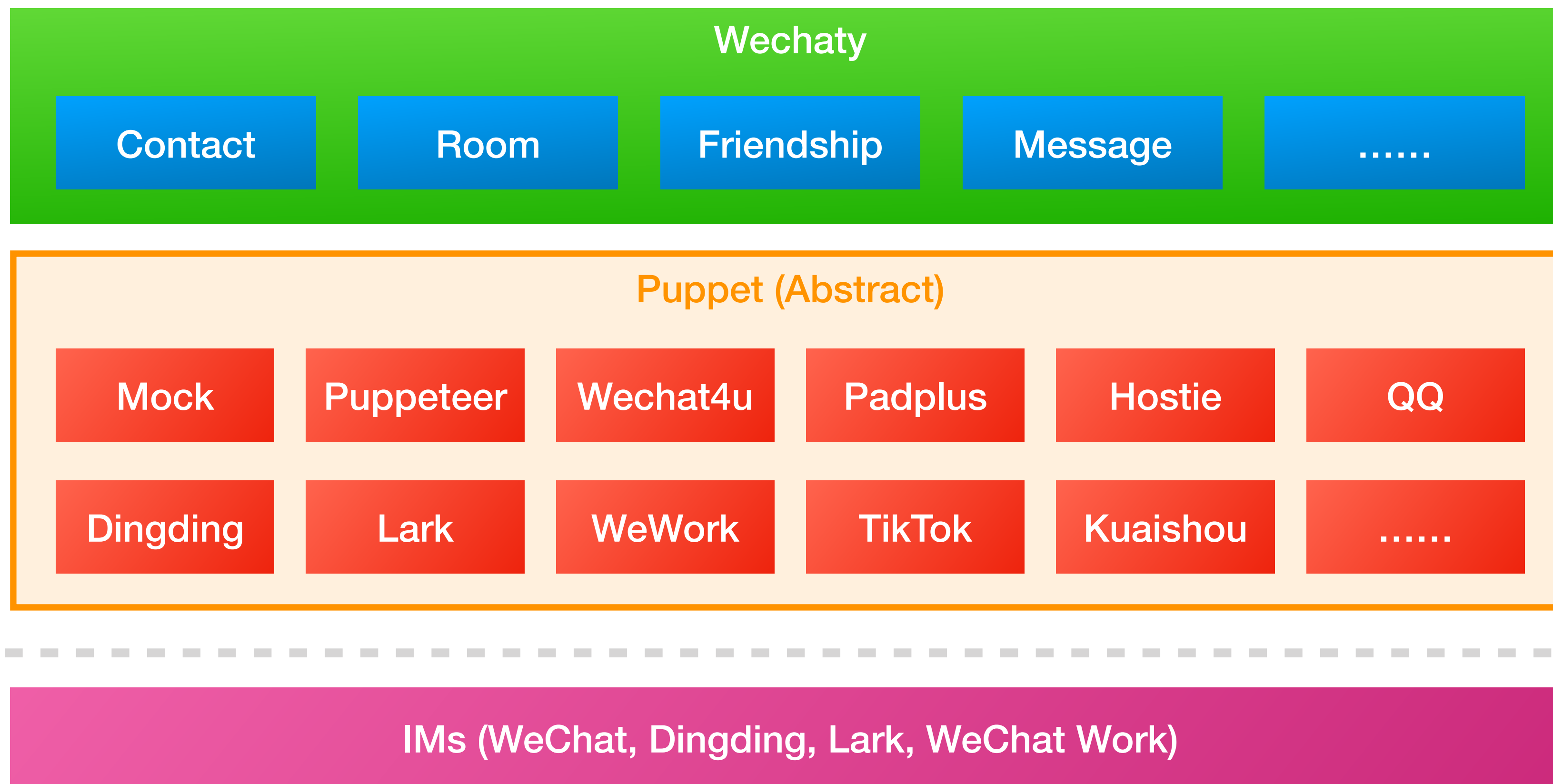
Wechaty

Architecture

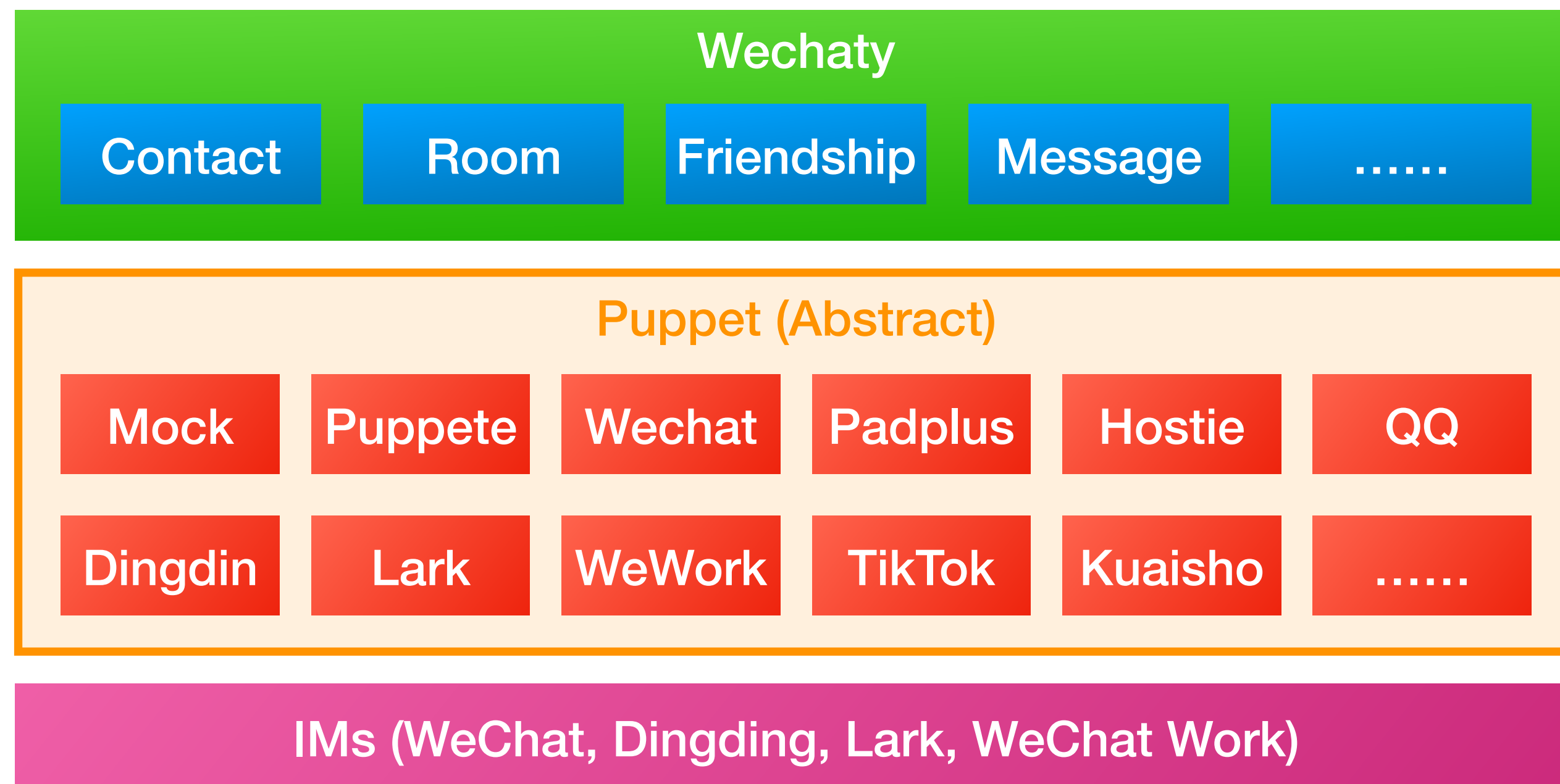
Wechaty Stack



Wechaty Architecture



Wechaty Architecture

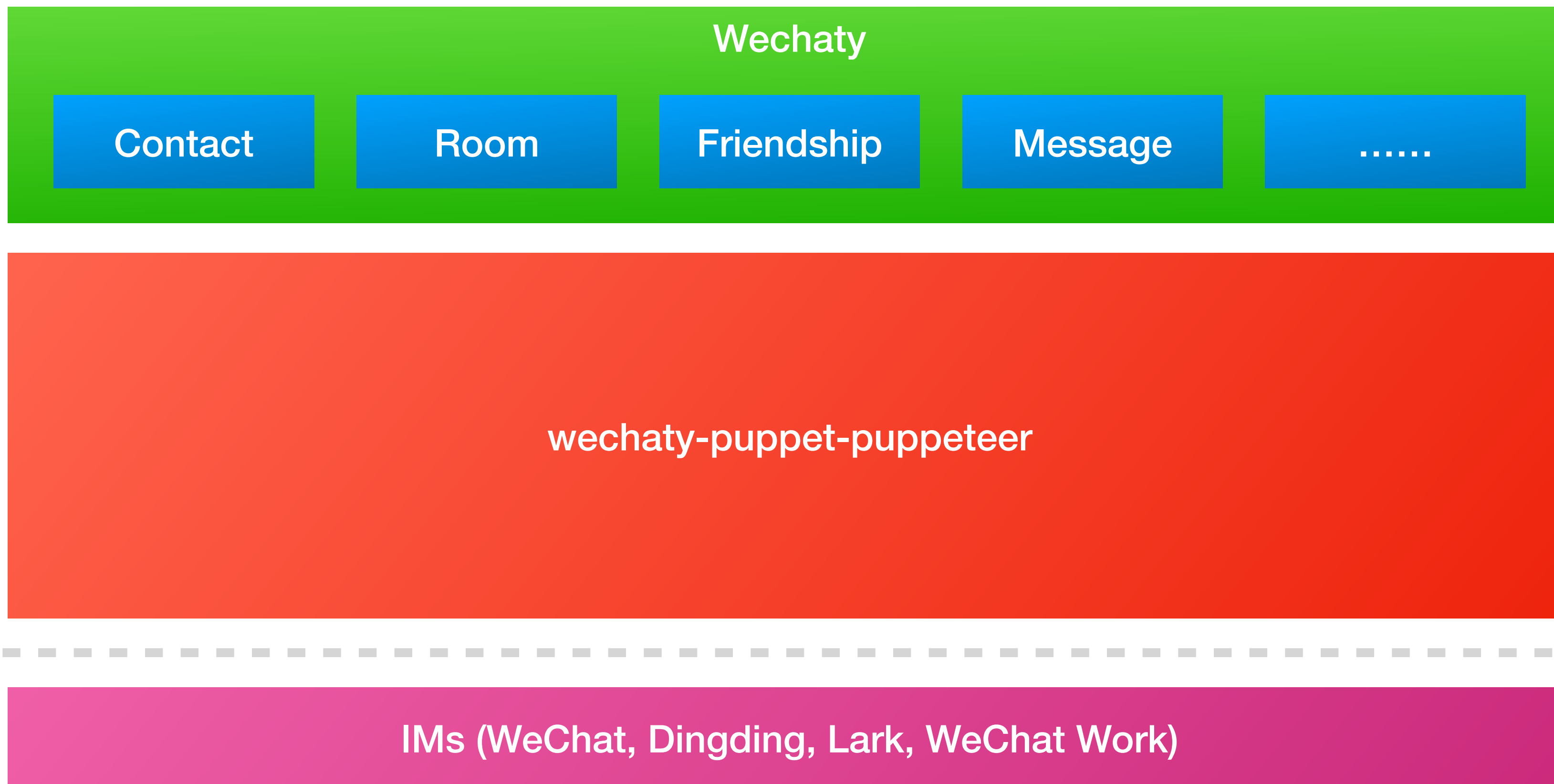


定义开发者使用的接口，连接开发者和底层Puppet实现

wechaty-puppet: 定义和Wechaty通讯的接口规范

Puppet实现: 实现 **wechaty-puppet** 定义的接口，并对接到真正的 **IM** 里面

When you using puppeteer



Contact.say()

- Wechaty:

```
class Contact {
  public async alias (newAlias?: null | string): Promise<null | string | void> {
    if (typeof newAlias === 'undefined') {
      return this.payload.alias || null
    }

    try {
      await this.puppet.contactAlias(this.id, newAlias)
      await this.puppet.contactPayloadDirty(this.id)
      this.payload = await this.puppet.contactPayload(this.id)
      if (newAlias && newAlias !== this.payload.alias) {
        log.warn('Contact', 'alias(%s) sync with server fail')
      }
    } catch (e) {
      log.error('Contact', 'alias(%s) rejected: %s', newAlias, e.message)
      Raven.captureException(e)
    }
  }
}
```

Contact.say()

- wechaty-puppet - contactAlias():

```
export abstract class Puppet extends EventEmitter {  
  public abstract async contactAlias (contactId: string) : Promise<string>  
  public abstract async contactAlias (contactId: string, alias: string | null) : Promise<void>  
}
```

Contact.say()

- wechaty-puppet-puppeteer - contactAlias():

```
export class PuppetPuppeteer extends Puppet {
  public contactAlias (contactId: string)                : Promise<string>
  public contactAlias (contactId: string, alias: string | null): Promise<void>

  public async contactAlias (
    contactId : string,
    alias?    : string | null,
  ): Promise<string | void> {
    if (typeof alias === 'undefined') {
      throw new Error('to be implement')
    }

    try {
      const ret = await this.bridge.contactAlias(contactId, alias)
      if (!ret) {
        throw new Error('bridge.contactAlias fail')
      }
    } catch (e) {
      log.warn('PuppetPuppeteer', 'contactRemark(%s, %s) rejected: %s', contactId, alias, e.message)
      throw e
    }
  }
}
```


How Hostie works

Wechaty

wechaty-puppet-hostie

wechaty-puppet-hostie Server

Wechaty

Some Puppet

IMs (WeChat, Dingding, Lark, WeChat Work)

Q&A