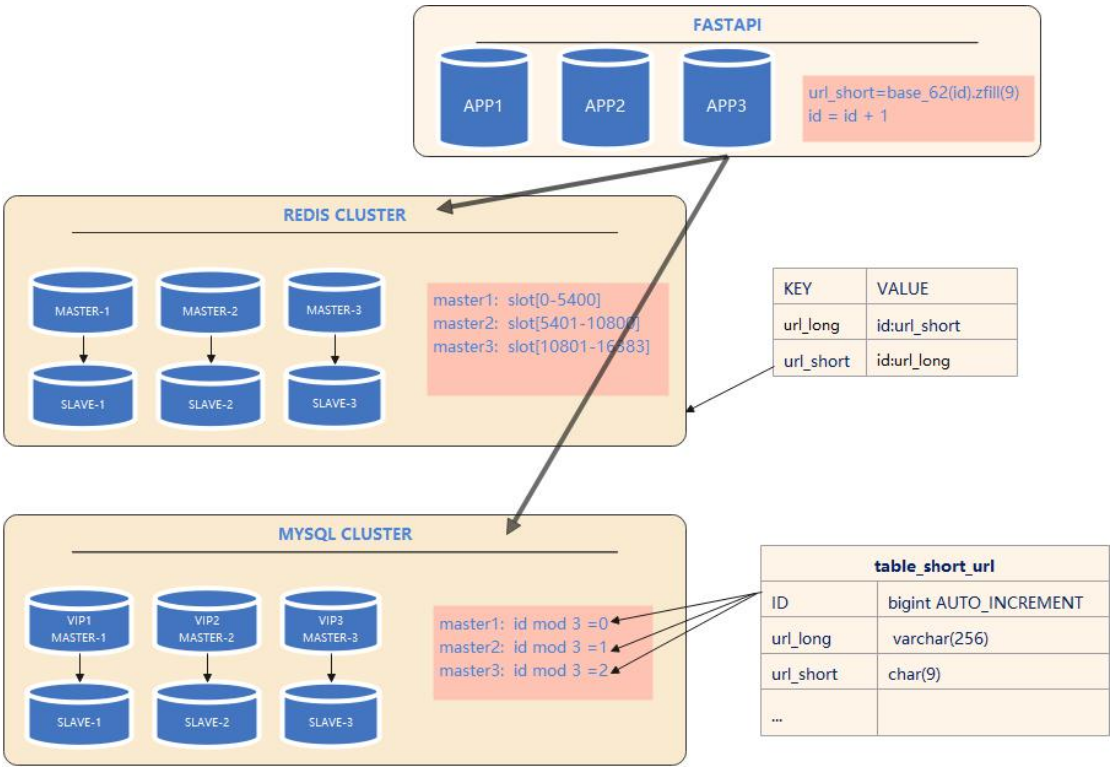
1. Architecture Diagram

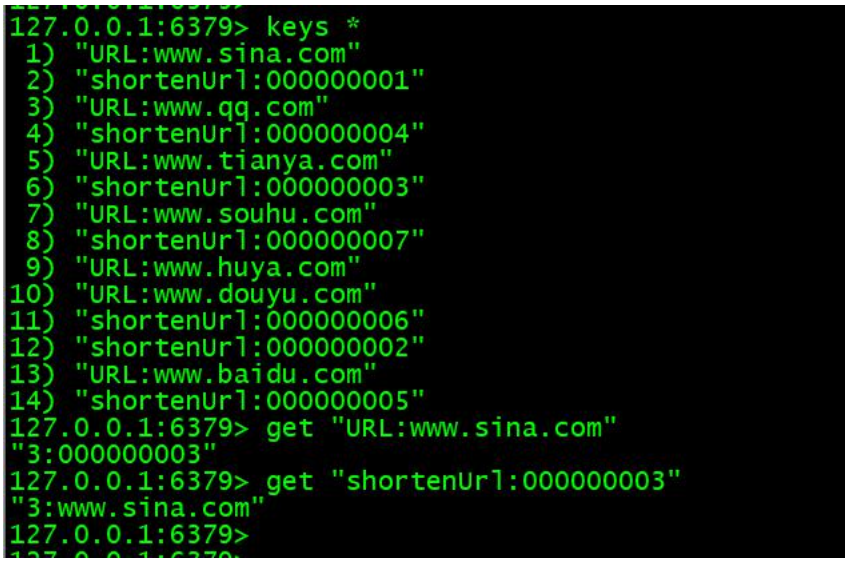


2. Detail database schema design

MYSQL:

```
CREATE TABLE `table_short_url` (
    `id` bigint NOT NULL AUTO_INCREMENT,
    `url_long` varchar(256) NOT NULL,
    `url_short` char(9) NOT NULL,
    `type` tinyint DEFAULT NULL,
    `general_count` int DEFAULT NULL,
    `click_count` int DEFAULT NULL,
    `share_count` int DEFAULT NULL,
    `create_time` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
    `update_time` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
    PRIMARY KEY (`id`),
    KEY `idx_url_long` (`url_long`)
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

REDIS:



3. Pseudo code

3.1. Web API endpoint for url submission

```
@app.post('/newurl')
async def newurl(request_data: Item):
    global lastrowid
    shortenUrl = rs.get("URL:" + request_data.url)
    if shortenUrl:
        res = {"url":request_data.url,"shortenUrl":shortenUrl}
    else:
        data = get_from_mysql_url_long(request_data.url)
        if data:
            [(id,url_short)] = data
            rs.set("URL:" + request_data.url,str(id) + ':' + url_short)
            res = {"url":request_data.url,"shortenUrl":url_short}
        else:
            id = lastrowid + 1
            shortenUrl = base_62_converter.int_to_string(id)
            store_url_mysql(id, request_data.url, shortenUrl)
            rs.set("URL:" + request_data.url, str(id) + ":" + shortenUrl)
            rs.set("shortenUrl:"+shortenUrl,str(id) + ":" + request_data.url)
            lastrowid = id
            res = {"url":request_data.url,"shortenUrl":shortenUrl}
    return res
```

3.2. Web API endpoint for redirectin

```
@app.get('/{shortenUrl}')
async def redirect(shortenUrl: str):
    print(shortenUrl)
    matchObj = re.match( r'(^([a-zA-Z0-9]{9}))$', shortenUrl, re.M|re.I)
    if not matchObj:
        res = {"error":shortenUrl + " does not match [a-zA-Z0-9]{9}."}
        return res
    value = rs.get("shortenUrl:" + shortenUrl)
    print(value)
    if value:
        print(value)
        value = str(value, encoding = "utf-8")
        [id,url] = value.split(':',1)
        print("test1",url)
        return RedirectResponse("http://"+url, status_code = 302)
    else:
        id = base_62_converter.string_to_int(shortenUrl)
        data = get_from_mysql_id(id)
        print(data)
        if data:
            [(url,)] = data
            print(url)
            rs.set("URL:" + url, str(id) + ":" + shortenUrl)
            rs.set("shortenUrl:"+shortenUrl,str(id) + ":" + url)
            return RedirectResponse(url, status_code=302)
        else:
            res = {"error":shortenUrl + " has not been generated, pls newurl it."}
            return res
```

4. Others Reference

https://blog.csdn.net/somenzz/article/details/115018991