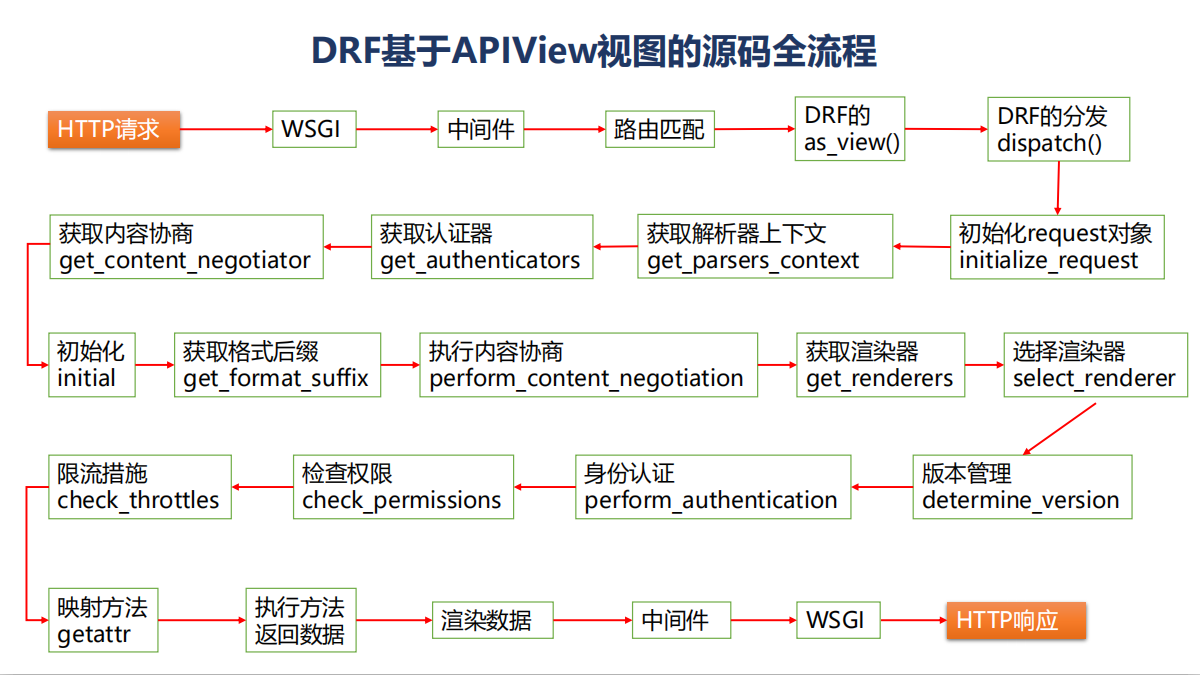


APIView



GenericAPIView

**get\_queryset(self)**

返回list视图中使用的查询集，该查询集还用作detail视图中的查找基础。

如：queryset = Post.objects.all()

**get\_object(self)**

返回用于detail视图的对象实例。默认使用 lookup\_field 参数过滤基本的查询集。 该方法可以被重写以提供更复杂的行为，例如基于多个 URL 参数的对象查找。

如：

filter\_kwargs = {self.lookup\_field: self.kwargs[lookup\_url\_kwarg]}

# 默认是{“pk”:”1”}

# 然后在根据过滤的来找，找不到就报错404

obj = get\_object\_or\_404(queryset, \*\*filter\_kwargs)

**filter\_queryset(self, queryset)**

给定一个queryset，使用任何过滤器后端进行过滤，返回一个新的queryset。

在过滤

**get\_serializer\_class(self)**

选择你想要使用的序列化类。默认返回 serializer\_class 属性的值。

可以被重写以提供动态的行为，例如对于读取和写入操作使用不同的序列化器或者为不同类 型的用户提供不同的序列化器。

get\_serializer\_class(self)本身没做啥，只是返回serializer\_class

这是一个钩子，通过这个可以指定返回自己需要的序列化类，如：

def get\_serializer\_class(self):

if self.request.user.is\_staff:

return FullAccountSerializer

return BasicAccountSerializer

**保存与删除操作中提供的钩子**:

以下方法由mixin类提供，并提供对象保存或删除行为的简单重写。

perform\_create(self, serializer) - 在保存新对象实例时 由 CreateModelMixin 调用。

perform\_update(self, serializer) - 在保存现有对象实例时 由 UpdateModelMixin 调用。

perform\_destroy(self, instance) - 在删除对象实例时由 DestroyModelMixin 调

用。