

Si ce service est aussi important et puisqu'il est confiné dans l'espace de nommage default, pourquoi un service équivalent n'est-il pas créé dans chaque nouvel espace de nommage ?

Le service **kubernetes** est créé uniquement dans l'espace de nommage `default` car il sert de **point d'accès unique et stable à l'API Server** pour tout le cluster, et il n'est pas nécessaire de le dupliquer dans chaque namespace.

Raisons principales

- **Résolution DNS globale :**

En Kubernetes, tous les pods de tous les namespaces peuvent accéder à l'API Server via le nom DNS :

```
kubernetes.default.svc
```

ou, en version complète :

```
kubernetes.default.svc.cluster.local
```

Ce nom DNS est accessible depuis n'importe quel namespace, car la résolution DNS de Kubernetes cherche d'abord dans le namespace courant, puis dans `default`, puis dans le cluster globalement.

Il n'est donc pas nécessaire d'avoir un service identique dans chaque namespace pour garantir l'accessibilité à l'API Server^{[1] [2] [3]}.

- **Unicité et stabilité :**

Le service `kubernetes` dans `default` est une **ressource centrale et unique**. Il pointe vers les endpoints de l'API Server (qui tournent dans `kube-system`), et son IP reste la même, même si les pods API Server changent d'IP ou sont redémarrés^[3].

- **Simplicité de gestion :**

Créer ce service dans chaque namespace serait redondant, inutile et source de confusion. Kubernetes garantit que ce service existe toujours dans `default` et le recrée automatiquement si besoin^[3].

- **Comportement standard :**

Lorsqu'un nouveau namespace est créé, il n'a pas besoin de ce service : tous les pods peuvent déjà joindre l'API Server via le service existant dans `default`^[1].

En résumé

- **Un seul service** `kubernetes` **dans** `default` **suffit** pour tout le cluster.
- **Tous les pods, dans tous les namespaces, peuvent accéder à l'API Server via ce service** grâce à la résolution DNS interne de Kubernetes.
- **Le dupliquer dans chaque namespace n'apporterait rien** et ne correspond pas au modèle d'architecture de Kubernetes^{[1] [2] [3]}.

~

1. <https://stackoverflow.com/questions/45123225/how-to-expose-kubernetes-api-internally-in-new-name-space>
2. <https://stackoverflow.com/questions/47523136/whats-the-purpose-of-the-default-kubernetes-service>
3. <https://www.linkedin.com/pulse/role-k8s-core-service-which-runs-default-namespace-deepak-kumar>