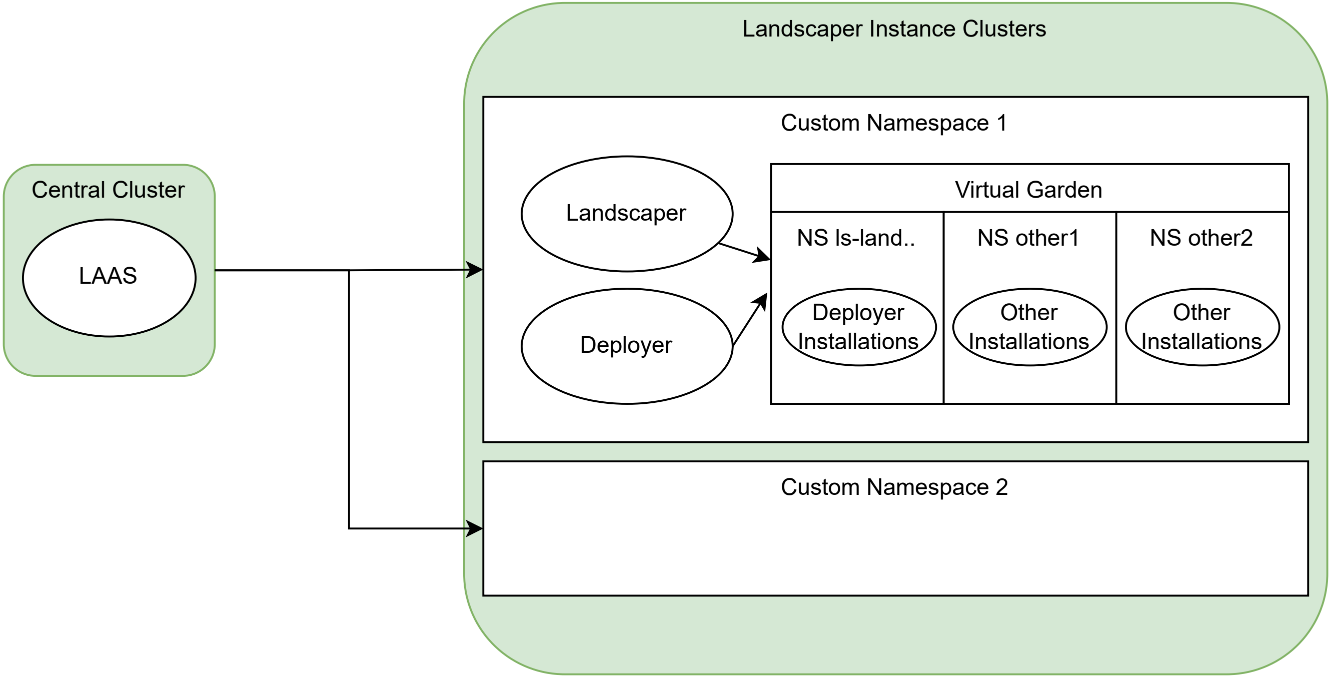
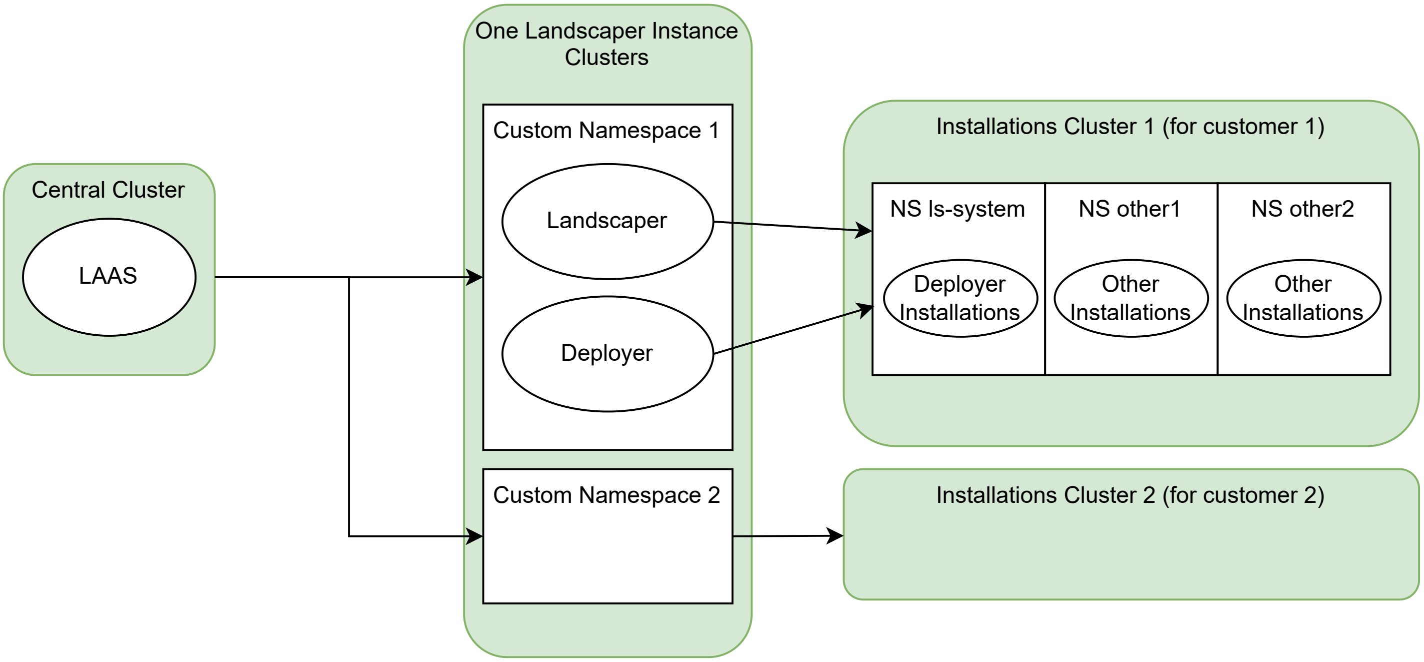
Current Realization:

- **Customer has access to Virtual Garden (including to the deployer installations)**



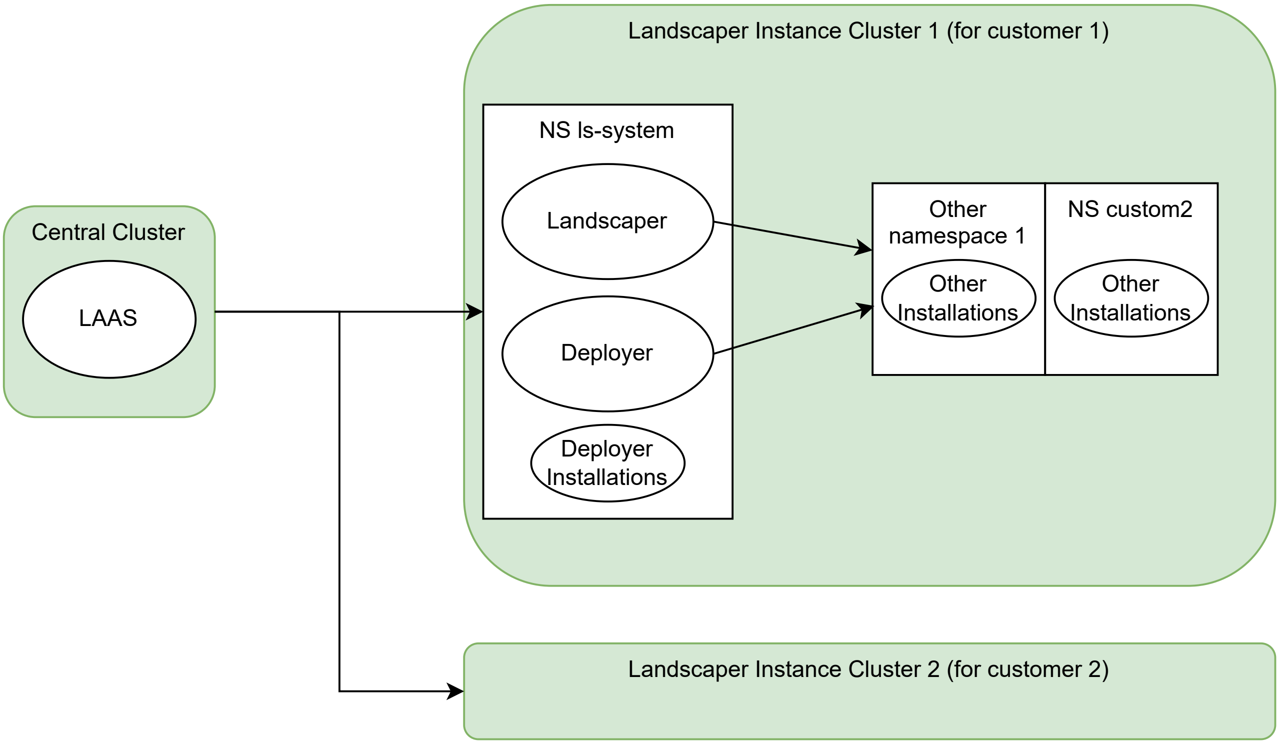
Alternative 1:

- **Installation Cluster is Garden Shoot Cluster**
- **Customer has access to Installation Cluster i (including to the deployer installations)**
  - **full Gardener Shoot Cluster**
  - **should be only restricted access for installations, secrets etc. but e.g. no deployments, pods ...**
- **In principal customer could also provide its own cluster but then he is responsible for app log configuration etc**



Alternative 2:

- **Installation Cluster is Garden Shoot Cluster**
- **Customer has access to Landscaper Instance Cluster**
  - **full Gardener shoot cluster**
  - **should be only restricted access for installations, secrets etc. but e.g. no deployments, pods ...**
  - **Problem: Customer has access to the secrets in ls-system except LAAS implements some controller**
    - **dynamically extending access to user created namespaces**
- **In principal customer could also provide its own cluster but then**
  - **he is responsible for app log configuration etc**
  - **he has access to the landscaper/deployer and could harm the system**



Assumption: It is sufficient to log modifications to installations

Comparison			
	Current Approach	Alternative 1	Alternative 2
Runtime costs	lowest costs because many instances on one shoot cluster	one additional cluster per customer	one additional cluster per customer
Maintenance costs	virtual garden with api server and etcd must be maintained including integration tests etc. support for runtime problems of api server and etcd		
Audit logging	must be implemented probably with side car approach in another project to decouple open source landscaper from SAP logging	Gardener out of the box approach	Gardener out of the box approach
sap vault integration	probably special implementation	probably standard approach	probably standard approach
separation of clients			best separation because landscaper are running on separate clusters which could be scaled independently
rotation of kubeconfig	required	required	required
security 1	customer has access to ls-system ns but secrets are external customer could modify deployer installations customer could modify deployitems - we should restrict this to read how to protect secret for service account used by landscaper restrict customer access to ns ls-system? requires further implementation	customer has access to ls-system ns but secrets are external customer could modify deployer installations customer could modify deployitems - we should restrict this to read how to protect secret for service account used by landscaper restrict customer access to ns ls-system? requires further implementation	customer has access to ls-system ns and there are also the secrets for the deployers customer could modify deployer installations customer could modify deployitems - we should restrict this to read how to protect secret for service account used by landscaper restrict customer access to ns ls-system? requires further implementation