# IBM FileNet Content Manager 5.7 Production Deployment on Red Hat OpenShift

An end-to-end technical approach from Operator-based deployment to persistent storage



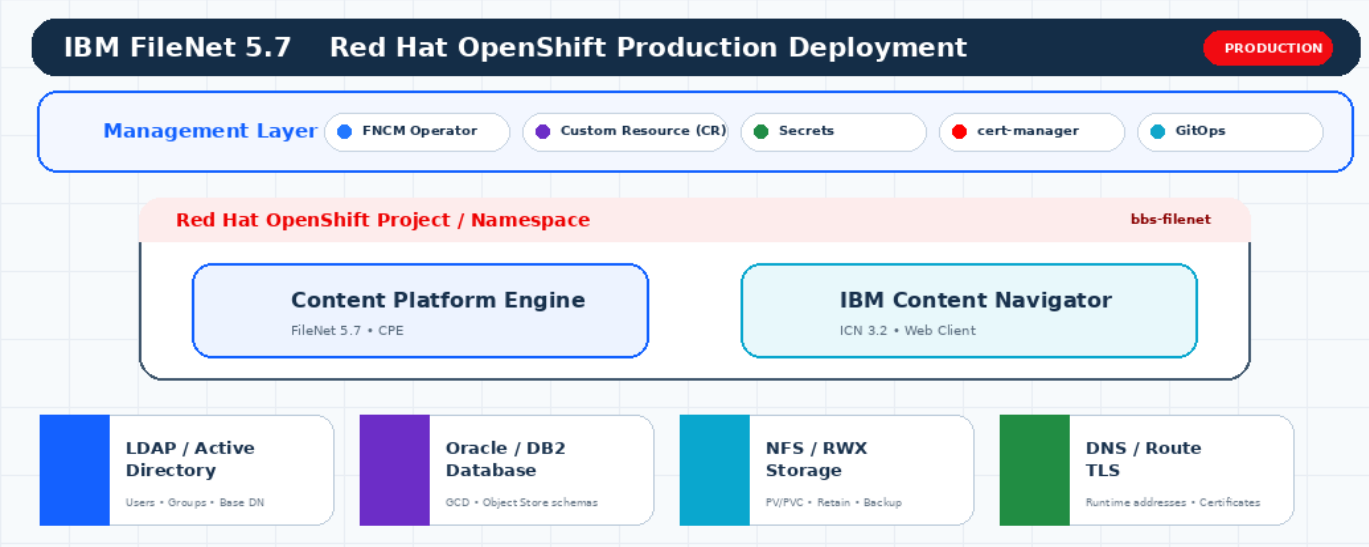
*Erkan Sezer | Deputy Manager, IBM Business Applications, Bilgi Birikim Sistemleri (BBS)*

## Executive Summary

Migrating IBM FileNet Content Manager to a container architecture is not merely about running pods; it requires identity, database, persistent storage, certificates, routes, and lifecycle management to be integrated into a single production design. This article provides a step-by-step approach to deploying IBM FileNet Content Manager 5.7 and IBM Content Navigator 3.2 on Red Hat OpenShift using the Operator, Custom Resource (CR), LDAP, Oracle/DB2, IBM Entitlement Key, and static NFS PV/PVC components.

**Core principle:** The CR file defines the desired state, and the FNCM Operator continuously reconciles the environment toward that state. Therefore, version-controlled, environment-specific management of CR, secret, and storage definitions is central to production reliability.

## Target Architecture




## Before You Start

| | | |
|---|---|---|
| **OpenShift**<br>Supported OCP version, worker capacity, DNS/route, and registry access | **Identity**<br>LDAP/AD bind account, user/group Base DNs, and administrator groups | **Database**<br>At minimum GCD and Object Store schemas; JDBC access and backup policy |
| **Storage**<br>RWX-capable storage class or static PV/PVC; Retain policy and capacity plan | **Security**<br>IBM Entitlement Key, secrets, TLS certificates, and least privilege | **Operations**<br>Monitoring, log collection, RPO/RTO, rollback, and maintenance window |

## 1. OpenShift Project, Operator, and Custom Resource

| | |
|---|---|
| 1 | **Project creation** — Create an isolated namespace from a Linux bastion using “oc new-project bbs-filenet”. ResourceQuota, LimitRange, NetworkPolicy, and RBAC decisions should be finalized before application deployment. |
| 2 | **Operator installation** — Install the FileNet Content Manager Operator from the IBM FNCM catalog in the OpenShift Console. The cert-manager Operator should also show a `Succeeded` status for certificate lifecycle management. Operator logs are the first place to investigate deployment and reconciliation errors. |

### Core validation commands

```
oc project bbs-filenet
oc get csv -n bbs-filenet
oc logs -f deployment/ibm-fncm-operator -c operator
```

| | |
|---|---|
| 3 | **Define the CR scope** — If only CPE will be deployed initially, set “cpe: true”; keep unused components such as CMIS, CSS, External Share, Task Manager, and GraphQL set to “false”. IBM Content Navigator can later be added through the same CR and the same Operator by setting “ban: true”. |

### Component selection - simplified example

```
content_optional_components:
  cmis: false
  css: false
  es: false
  tm: false
  cpe: true
  graphql: false
  ban: false
```

## 2. LDAP, Database, and Secret Design

The FileNet domain and Object Stores are tightly coupled with the identity directory and relational database. Before applying for the CR, connectivity tests, DNS resolution, port access, certificate chains, and service-account permissions should be validated from both the bastion host and the pod network.

| LDAP / AD | Oracle / DB2 | Secrets |
|---|---|---|
| Bind account; user/group Base DN; separate groups for domain administrators and Object Store administrators. | One scheme for GCD and a separate scheme for each Object Store; JDBC URL, service name, and pool settings. | Database, application, and keystore credentials in “ibm-fncm-secret”; passwords are not stored in the CR. |

IBM documentation recommends using separate groups for FileNet P8 domain administrators and Object Store administrators. This enables team membership to be managed through LDAP and reduces dependency on a single user account. If FIPS is enabled, password-length requirements for the key store and certain database passwords should also be verified.

### Critical references in the CR

```
ecm_configuration:
  fncm_secret_name: ibm-fncm-secret
  cpe:
    hostname: filenet.example.com
    replica_count: <capacity-and-ha-plan>

shared_configuration:
  image_pull_secrets:
    - ibm-entitlement-key
```

**Production Note:** The source field example used “replica_count: 1”. In production, the replica count should be determined together with licensing, the support matrix, capacity testing, pod anti-affinity, and maintenance/failure scenarios. The example value should not be copied directly.

## 3. Initial Deployment and Observation

The “initialize_configuration” section is critical when the FileNet domain and Object Store objects are created for the first time. Initial deployment must be distinguished from expansion/update scenarios so that the same infrastructure is not accidentally initialized again. After the CR is applied, Operator status, pod events, deployments, services, and routes should be monitored together.

### Deployment and runtime status

```
oc apply -f ibm_fncm_article.yaml
oc get pods,deployments,svc,routes -n bbs-filenet
oc get events -n bbs-filenet --sort-by=.lastTimestamp
```

## 4. Adding IBM Content Navigator 3.2 to the Same Architecture

IBM Content Navigator (ICN) is the web client through which FileNet users access documents, folder, search, and workflow capabilities. In the source implementation, ICN was enabled in the “bbs-filenet” project by extending the existing FNCM Operator and the existing CR; a separate Navigator Operator was not installed.

### Adding ICN - simplified CR delta

```
content_optional_components:
  cpe: true
  ban: true

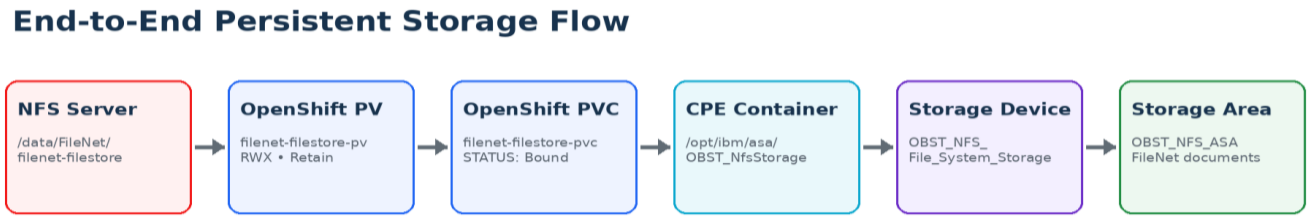
navigator_configuration:
  ban_secret_name: ibm-ban-secret
  hostname: navigator.example.com
  replica_count: <capacity-and-ha-plan>

shared_configuration:
  sc_content_initialization: false
```

**Why “sc_content_initialization: false”?** CPE has already been deployed, and the Content Services infrastructure has been initialized. When ICN is added, the same infrastructure should not be initialized again. The value used for the initial deployment must be distinguished from the value used in subsequent expansion steps.

## 5. Static NFS PV/PVC and Advanced Storage Area

For production content storage, persistence should be decoupled from the application pod. IBM documentation allows static or dynamic provisioning. In the static model, PV and PVC resources are created in advance, and the existing claim name is referenced in the CR. For FileNet volumes that contain data, `ReadWriteMany` access mode and the `Retain` reclaim policy are two critical controls.




**PV**

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: filenet-filestore-pv
spec:
  capacity: {storage: 100Gi}
  accessModes: [ReadWriteMany]
  persistentVolumeReclaimPolicy: Retain
  storageClassName: managed-nfs-storage
  nfs:
    server: <nfs-server>
    path: /data/FileNet/filenet-filestore
```

**PVC**

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: filenet-filestore-pvc
  namespace: bbs-filenet
spec:
  accessModes: [ReadWriteMany]
  resources:
    requests: {storage: 100Gi}
  volumeName: filenet-filestore-pv
  storageClassName: managed-nfs-storage
  volumeMode: Filesystem
```

The CR should not be applied until the PVC “STATUS” is “Bound”. The claim for the CPE file store is defined under “datavolume.existing_pvc_for_cpe_filestore” in the CR. According to IBM's volume table, the container mount path for the FileNet file-system-based Advanced Storage Device is “/opt/ibm/asa”.

### Binding the Existing PVC to CPE

```
ecm_configuration:
  cpe:
    datavolume:
      existing_pvc_for_cpe_filestore:
        name: filenet-filestore-pvc
        size: 100Gi
```

### Final Mapping in ACCE

- Administration Console for Content Platform Engine (ACCE) → Advanced Storage Areas → New Advanced Storage Area.
- Example storage area name: “OBST_NFS_ASA”; device type: File System Device; example device name: “OBST_NFS_File_System_Storage”.
- The container equivalent of the NFS folder “/data/FileNet/filenet-filestore/OBST_NfsStorage” is entered as “/opt/ibm/asa/OBST_NfsStorage”.
- Add a document to the test folder and validate both the ACCE counters and creation of the physical file in the NFS directory.

## 6. Critical Production Controls

| Area | Control | Acceptance Criterion |
|---|---|---|
| **Capacity / HA** | Replicas, CPU/RAM requests-limits, anti-affinity, PDB, and worker distribution | Service objective is maintained during maintenance or a single-node loss |
| **Database** | GCD/OS schemas, connection pool, HA, maintenance, and consistent backup | Connectivity tests and rollback rehearsal completed |
| **Storage** | RWX, Retain, NFS/CSI performance, quota, inode, and growth | PVC Bound; read/write and restore tests successful |
| **Identity** | LDAP access, Base DN, admin groups, service-account rotation | Administrator and end-user sessions validated |
| **Security** | TLS, route, secret management, RBAC, NetworkPolicy, registry access | No plaintext passwords; certificate chain valid |
| **Operations** | Logs/metrics/alerts, runbook, RPO/RTO, DR, and upgrade plan | Monitoring and rollback scenario exercised |

## 7. Go-Live and Validation Flow

| | |
|---|---|
| **1** | Step — Review the CR and secret manifests through Git; separate environment values from the template and do not commit secrets to the repository. |
| **2** | Step — Independently validate the Operator/CSV, cert-manager, StorageClass, PV/PVC, DNS, LDAP, and database connections. |
| **3** | Step — Apply the CR; monitor Operator logs, events, pod readiness, services, and routes. In the event of an error, identify the first root event rather than treating only the symptom. |
| **4** | Step — Run ACCE ping/access checks, validate the FileNet domain and Object Store, and then test the ICN desktop, repository connection, search, document upload/download, and permissions. |
| **5** | Step — Write a test document to the Advanced Storage Area; verify end-to-end consistency across the ACCE counter, container mount, and physical NFS path. |
| **6** | Step — Establish a performance baseline, execute backup/restore and node-maintenance scenarios, and open production traffic only after both technical and business acceptance criteria have been met. |

## Conclusion

Successfully running IBM FileNet Content Manager 5.7 on OpenShift depends less on the Operator installation itself than on disciplined design of its dependencies. When properly separated LDAP groups, independent GCD/Object Store schemas, secure secret management, replicas sized for production requirements, and persistent storage aligned with the RWX-Retain principles are addressed together, the platform becomes a scalable, observable, and manageable enterprise content service. The CR approach and static NFS chain demonstrated in the source implementation provide a practical field model for this transformation.