3/14/2025

BILH Interface Engine

HealthShare Health Connect

Robert Hurst

BETH ISRAEL LAHEY HEALTH

Table of Contents

BILH Interface Engine

Health Connect

InterSystems HealthShare Health Connect is a healthcare integration engine that delivers high-volume transaction support, process management, and monitoring to support mission critical applications.

At the heart of Health Connect is a high performance, multi-model data engine called `IRIS` that seamlessly handles multiple forms of data at high speed. Health Connect easily scales from serving small clinics to handling the transaction volumes of the largest and most complex healthcare delivery systems in the world.

Tangible benefits

- Proven rapid scalability and fast implementation beyond competitor benchmarks
- Zero-downtime stability and reliability for even the largest IDNs with thousands of connections and millions of daily message transactions
- Faster interface development to save costs, with additional savings from eliminating separate vendor fees.

Capabilities

- Interoperability by design
- Mirroring with fast failover recovery
- Source control for HL7 schemas
- Intuitive drag-and-drop HL7 schema editing
- A flexible, adaptable security model and more

See also the **IRIS Technology Overview** section at the end of this document.

Overview

The purpose of this interface engine is to enable our Application Integration team to connect systems, so that they can transform and route messages between them. To connect systems, they develop, configure, deploy, and manage productions, which integrate multiple software systems.

An `interoperability production` is an integration `framework` for easily connecting systems and for developing applications for interoperability. A production provides built-in connections to a wide variety of message formats and communications protocols. Developers can easily add other formats and protocols – and define business logic and message transformations either by coding or using graphic interfaces.

Productions provide persistent storage of messages, which allow you to trace the path of a message and audit whether a message is successfully delivered. The elements in a production are known as `business hosts`. There are three kinds of business hosts, with different purposes as follows:

- `Business Service` connects with external systems and receive messages from them. Business services relay the messages to other business hosts in the production;

- **Business Process** allows the developer to define business logic, including routing and message transformation. Business processes receive messages from other business hosts in the production and either process the requests or forward them to other business hosts; and

- **Business Operation** connects with external systems and send the messages to them. Business operations receive messages from other business hosts in the production and typically send them to external systems.

Business hosts communicate with each other via messages. All messages are stored in an IRIS database and can be seen via the **Management Portal**.
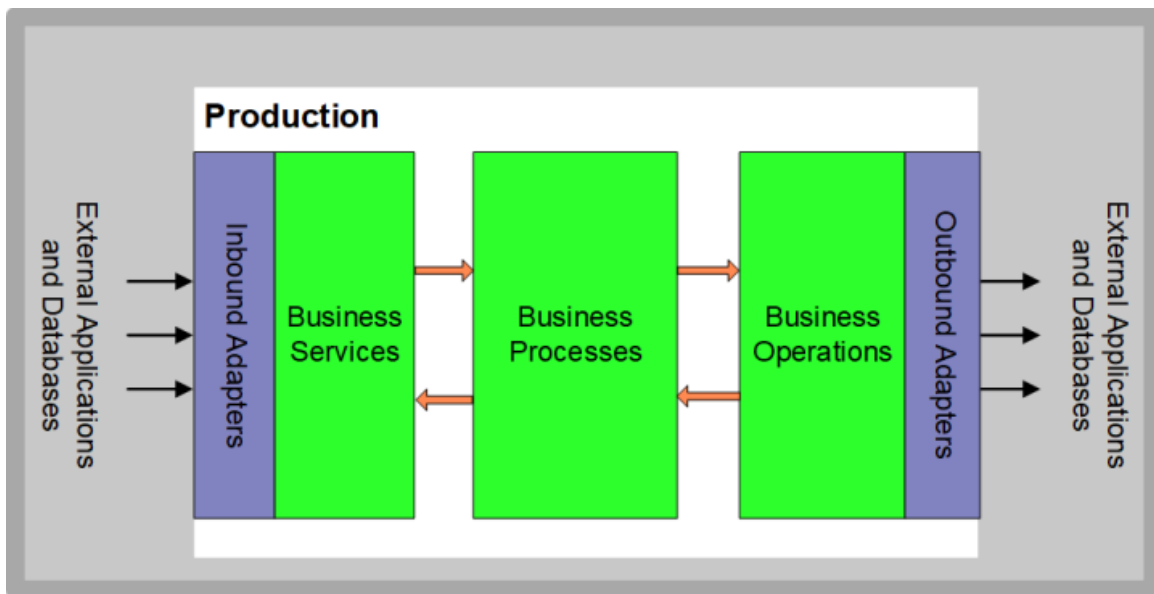
Technology & Innovation Service Line

Application Integration team

DevOps remote access into the BILH system Health Connect productions:

`https://hciedev.laheyhealth.org/devops`

From there are all the relevant links to each Management Portal, Dashboards, and Documentation.

Cockpit Navigator for Files

- Live – `https://hcieprd.laheyhealth.org/files/navigator`
- Test – `https://hcietst.laheyhealth.org/files/navigator`
- Development – `https://hciedev.laheyhealth.org/files/navigator`

InterSystems ObjectScript

Caché Studio IDE that uses direct access into the platform SuperServer port is deprecated. Their Atelier API web service for developers was ported from Eclipse as a plug-in extension pack for Microsoft VS Code:

- `https://intersystems-community.github.io/vscode-objectscript`

Interoperability

DevOps is performed chiefly within these menu items on the Management Portal, such as:

- Switching between IRIS Namespaces that run a production
- Maintaining a Business Partner directory to identify their interfaces
- Credential management for business host adapters that use them
- Data Lookup Tables for local dictionary use in data transformations
- Developing Business Processes and Rules for message routers
- Defining record maps, both flat and complex
- Manage messages
- View alerts and logs
- Production monitoring

Task Manager

Other scheduled tasks can be constructed here outside a production, including but not limited to:

- Standard instance journal purge
- Standard production message & activity purges
- Production code and data dictionaries exporting with files archive rotation and retention
- Daily update for the instance `Port Authority Report` stored & viewed out of `HSCUSTOM`

Files

Logical links `/files` and `/hs` point to their respective dev-test-live filesystem designated for external production files. Both are provided to maintain logical consistency for BIDMC and Lahey current state with interfaces.

Archive

As listed in each Task Manager, a nightly archive script `/mnt/autofs/media/bin/archive.sh` runs to:

1. export class source code and dictionary lookup tables out of each production
2. `zip` archive its `etc` subfolder containing the productions' critical files kept external to IRIS
3. rotate these designated `/files` subfolders into the NAS media share for archiving
    a. `in` – daily files & subfolders can accumulate here
    b. `scripts` – reserved for runtime production scripts with log files
    c. `tmp` – runtime files & subfolders are purged
    d. `Deploy` – move **all** dev,tst,prd Ensemble deployments to avoid accidental mistakes into the nightly HCIE**DEV** `/files/Archive`
4. prune aged files outside of retention days, but always keep the last `50` files in its Production subfolder

An example of Archive test listing; note that it also references the other productions dev & prd:

```
[...@...t02ss ~]$ ls /files/Archive/
… snipped …
20230924  20231018  20231111  20231205  20231229  20240122  20240215  20240310
20230925  20231019  20231112  20231206  20231230  20240123  20240216  20240311
20230926  20231020  20231113  20231207  20231231  20240124  20240217  BILHHOSPITALS
20230927  20231021  20231114  20231208  20240101  20240125  20240218  BILHSFTP
20230928  20231022  20231115  20231209  20240102  20240126  20240219  dev
20230929  20231023  20231116  20231210  20240103  20240127  20240220  LAHEYHEALTH
20230930  20231024  20231117  20231211  20240104  20240128  20240221  prd
20231001  20231025  20231118  20231212  20240105  20240129  20240222
… snipped …
```

Each daily folder contains the data cleared out of `/files` with each Production namespace class code exported:

```
[...@...t02ss ~]$ ll /files/Archive/20240311/
total 8725
-r-xr-xr-x 1 root irisdev  2931430 Mar 11 23:59 BILHHOSPITALS.xml
-r-xr-xr-x 1 root irisdev   317248 Mar 11 23:59 BILHSFTP.xml
-r-xr-xr-x 1 root irisdev      928 Mar 11 23:59 BILHWORKDAY.xml
-r--r--r-- 1 root irisdev   143313 Mar 11 23:59 etc.zip
-r-xr-xr-x 1 root irisdev   769798 Mar 11 23:59 HSCUSTOM.xml
dr-xr-sr-x 2 root irisdev        0 Mar 10 23:59 in
-r-xr-xr-x 1 root irisdev 28999923 Mar 11 23:59 LAHEYHEALTH.xml
dr-xr-sr-x 2 root irisdev      145 Mar 11 13:47 scripts
dr-xr-sr-x 2 root irisdev        0 Mar 10 23:59 tmp
```

DISCLAIMER: *when a developer works outside this schematic, their choices are not protected and invites unexpected, but avoidable, problems. Communicate and collaborate with Linux and ODBA teams if a desired **change** to this schematic is warranted.*

cifs

Has a bind mount to `/mnt/autofs` which is Linux `inotify` monitored to automatically mount a remote filesystem upon first access. Those remote filesystem entries are managed in `/etc/auto.misc` as exampled:

```
# external shares
media   -rw,hard,vers=4.0,sec=sys lhnas03.lahey.org:/ifs/bosinas03/nfs/media
EDI -fstype=cifs,credentials=/etc/ITS,dir_mode=0775,file_mode=0664,gid=irisdev,vers=3.0
://galaxy.bidmc.harvard.edu/clinical/EDI
```

Having this managed by the Linux service, `autofs`, allows for improved operational resilience as it is managed in user space mode; and it will also dismount and release the remote share when it sits idle for a few minutes:

```
$ date; cd /files/cifs/EDI ; df -t cifs ; cd - ; sleep 200 ; date; df -t cifs
Wed Oct 18 11:06:35 EDT 2023
Filesystem                              Type  Size  Used Avail Use% Mounted on
//galaxy.bidmc.harvard.edu/clinical/EDI cifs  2.0P  1.7P  310T   85% /hcdev-files/cifs/EDI
Wed Oct 18 11:09:55 EDT 2023
df: no file systems processed
```

Deploy

Like Archive, a logical link is provided out of each dev-test-live instance to deploy production interfaces between them:

```
$ ll /files/Deploy/
drwxrwsr-x 3 irisusr irisdev 180 Jan 31 08:22 dev
drwxrwsr-x 2 irisusr irisdev   0 Jan 18 13:44 prd
drwxrwsr-x 3 irisusr irisdev 115 Jan 28 06:52 tst
```

The Developer should deploy export **into** the respective `dev-tst-prd` folder, then deploy import **from** that source within the target Production.

DISCLAIMER: *when a developer works outside this schematic, their choices are not protected and invites unexpected, but avoidable, problems. Review above* `Archive` *section for details.*

Internet outbound

It may also be necessary to inform external sites to allow client connections in from these BILH public addresses out of the `Markley data center`:

- `208.64.112.0/21`
- `199.255.48.0/21`

The `Lahey PAT` pool is `208.65.115.1-10`

DevOps

Custom functions are kept mainly in the Ens.Rule.FunctionSet extended class named `Functions.Library`. Each method in it can be found within the `RuleSet Editor` under its function [*fx*] dropdown pick-list and can also be invoked from DTL using a formal `##class(Functions.Library)` method call.

This section details the custom `Common.Production.*` `business hosts` used throughout the `Productions` in order to better accommodate operational practices and business need:

Business Services

`FilePassthroughService`[+] `FTPPassthroughService`[+]
`RecordMapService`[+]
`RecordMapBatchFileService`[+] `RecordMapComplexBatchFileService`[+]
`RecordMapBatchFTPService`[+] `RecordMapComplexBatchFTPService`[+]
`RestClientService`
`XMLFileService`[+] `XMLFTPService`[+]

Business Processes

`Common.Util.Pipe` is a viable alternative for any business need that can use its `RunCommand` method to invoke a hosted tool within an EDI DTL or BPL code block.

Business Operations

`AWSOperation` not actively in-use, but available as a potential option if a business need for it surfaces to use its AWSRequest / AWSResponse message formats.

`FilePassthroughOperation`[-] `FTPPassthroughOperation`[-]
`RecordMapBatchFileOperation`[-] `RecordMapComplexBatchFileOperation`[-]

`SMTPClientOperation` uses `Common.Message.EmailRequest` to allow for a custom email message with optional file attachments out of a business process.

+ includes structured file `Archive` option to keep a copy in `/files/Archive/system/namespace/interface-name/mmdd-hhmmss_filename`

- inherits `Pre-Process` inbound adapter options as implemented in `Common.Util.Preprocess` class, which includes:

- `Upsize` allows for the process to allocate a larger runtime memory heap, in kilobytes, to avoid `<STORE>` error that can happen when processing large XML files and Complex Record batches.
- `Required` allows a value >0 to signal an alert if at least *n*-FILE(s) expected to be processed within its scheduled period (not interval) was not met.
- `DoAll` can pass ALL filename(s) discovered together to `Command`, rather than the usual one-at-a-time.
- `OnInit` allows `Command` to invoke only once at process startup, rather than invoke each time before passing the message along.
- `ObjectScript` is a code block, if any, for IRIS to execute first as part of `Command` need.
- `Env` is a free-text placeholder to pass along overriding `Script` values, if any, as part of `Command` need

- Script is a code block, if any, for Linux to execute last as part of Command need.
- NewFile is the file naming expression expected after Command executes, and it is this new file that is passed along in the message output.

Common.Util.EDI class provides a working set of methods used primarily with the **Production** classes listed above, but can also be used in any code block:

```
BILHHOSPITALS>write ##class(Common.Util.EDI).#FILES
/files
BILHHOSPITALS>write ##class(Common.Util.EDI).#INSTANCE
HCTEST
BILHHOSPITALS>write ##class(Common.Util.EDI).#SHARED
/files/Archive
BILHHOSPITALS>write ##class(Common.Util.EDI).#TEMP
/files/TRASH
```

Its other methods:

Archive(pDocument As %Stream.Object, pFolder As %String, pFile As %String) to **copy the IRIS stream passed into the structured archive folder schematic**; and

MakeTemp(pSource As %Stream.Object, ByRef pTarget As %Stream.Object, pFolder As %String, pFilename As %String) to **clone the IRIS stream passed into a working file on the host** that gets expunged nightly.

ODBA team

IRIS Installation

The installation binary is kept on the auto-mount `Media` share as noted in this example:

```
$ sudo -s
# cd ~irisusr
# tar xzvf /mnt/autofs/media/src/HealthConnect-2022.1.4.812.0.22957-lnxrh8x64.tar.gz
# cd HealthConnect-2022.1.4.812.0.22957-lnxrh8x64
# ./irisinstall
instance: HCDEV
SuperServer: 1972
```

Its `iris.key` is also kept in the media subfolder: `etc`

Configuration

Each instance is configured with a complement of database and routine cache that allows for the Health Connect productions to run efficiently and safely within the host resource capacity.

The advanced memory attribute, `LockSharedMemory`, is enabled for Linux to mark those allocations as non-swappable, which avoids unnecessary contention for consistent access:

```
$ sudo ipcs -m
------ Shared Memory Segments --------
key         shmid    owner     perms    bytes         nattch    status
0x5a5b768c  2        irisusr   660      15272214528   704       locked
```

Security

Start by enabling the instance `Authentication` options:

System > Security Management > Authentication/Web Session Options - (security settings)

Authentication/Web Session Options   [Save] [Cancel]

Edit Security Authentication/Web Session Options:

Allow Unauthenticated access	☑
Allow O/S authentication	☑
Allow O/S authentication with Delegated authorization	☑
Allow O/S authentication with LDAP authorization	☐
Allow Password authentication	☑
Allow Delegated authentication	☑
Always try Delegated authentication	☐
Allow Kerberos authentication	☐
Allow LDAP authentication	☐
Allow LDAP cache credentials authentication	☐
Allow creation of Login Cookies	☐ Login Cookie expire time (secs) [0] 0 implies non-persistent cookie
Allow Two-factor Time-based One-time Password authentication	☐
Allow Two-factor SMS text authentication	☐

Then enable and assign those authentication methods on the instance `Services`:

System > Security Management > Services

Services

Services are the primary means by which users and computers connect to InterSystems IRIS. The following services are

Page size: 0   Max rows: 1000   Results: 14   Page: |< << 1 >> >| of 1

	Name	Enabled	Public	Authentication Methods	Allowed Connections	Description
	%Service_Bindings	Yes	N/A	Password,Delegated	Unrestricted	Controls SQL or Objects
	%Service_CacheDirect	No	Yes	Password	Unrestricted	Controls Cache Direct
	%Service_CallIn	No	Yes	Operating System,Password	Unrestricted	Controls the Call-In Interface
	%Service_DataCheck	Yes	N/A		Unrestricted	Controls this system as a DataCheck source
	%Service_DocDB	No	No		Unrestricted	Controls Doc DB applications
	%Service_ECP	No	N/A		Unrestricted	Controls Enterprise Cache Protocol (ECP)
	%Service_Login	Yes	No	Delegated	Unrestricted	Controls SYSTEM.Security.Login
	%Service_Mirror	Yes	N/A		Unrestricted	Controls Mirroring
	%Service_Monitor	No	N/A		Unrestricted	Controls SNMP and remote Monitor commands
	%Service_Shadow	No	N/A		Unrestricted	Controls if this system can be the source of a shadow
	%Service_Sharding	No	N/A		Unrestricted	Controls this system as a Shard Server
	%Service_Terminal	Yes	Yes	Operating System,Password,Delegated	Unrestricted	Controls terminal session on Unix
	%Service_WebGateway	Yes	Yes	Password,Unauthenticated,Delegated	Unrestricted	Controls Web Gateway access
	%Service_Weblink	No	N/A	Unauthenticated	Unrestricted	Controls Weblink

`Delegated` access uses the IRIS ZAUTHENTICATE site implementation that allows:

1. single sign-on authentication with our AD LDAP service
2. authorization is delegated using Linux groups, in order:
   i. `sysadm` for SysOps
   ii. `irisadm` for DevOps administration
   iii. `irisdev` for DevOps

NOTE: privileged escalation with the Linux `root` or `sudo` account access is only allowed from a Terminal Session (`irisdb` or `irissession`) off the host. The `root` account is provisioned with IRIS `%All` access:

```
 $ sudo irisdb
%SYS> write $username,!,$roles
root
%All
%SYS> do $SYSTEM.OBJ.Load("/mnt/autofs/media/src/ZAUTH-dev.xml","ck")
```

Role-based Access Controls

`Microsoft Visual Studio Code` with the InterSystems `ObjectScript` pack extension installed is the preferred development suite for IRIS. To allow DevOps connectivity, the `%Developer` role requires additional SQL access – for each Namespace allowed – to the stored procedures and tables that are referenced by the `Atelier` API web services:

```
$ sudo irisdb
%SYS>do $SYSTEM.SQL.Shell()
[SQL]%SYS>>GRANT EXECUTE ON %Library.RoutineMgr_StudioOpenDialog TO %Developer
[SQL]%SYS>>GRANT SELECT, INSERT, UPDATE, DELETE ON %Studio.Project, %Studio.ProjectItem TO
%Developer
[SQL]%SYS>>GRANT EXECUTE ON %Studio.Project_ProjectItemsList TO %Developer
[SQL]%SYS>>GRANT SELECT ON SCHEMA %Dictionary TO %Developer
… repeat for each production namespace, and also …
[SQL]%SYS>>! zn "BILHHOSPITALS"
[SQL]BILHHOSPITALS>>GRANT SELECT, INSERT, UPDATE, DELETE ON Ens_Util.LookupTable TO
%EnsRole_Administrator
… additional SQL permissions needed …
[SQL]%SYS>>! zn "HSCUSTOM"
[SQL]HSCUSTOM>>GRANT SELECT ON Ens_Activity_Data.Days TO %EnsRole_Operator
[SQL]HSCUSTOM>>GRANT SELECT ON Ens_Activity_Data.Hours TO %EnsRole_Operator
[SQL]HSCUSTOM>>GRANT SELECT ON Ens_Activity_Data.Seconds TO %EnsRole_Operator
```

NOTE: IRIS 2022.1.1+ only %Development is required, not these SQL permissions.

Mirror Member Management

LIVE

- https://lvhshcadp01ss.laheyhealth.org/csp/sys/op/%25CSP.UI.Portal.Mirror.Monitor.zen
- https://lvhshcadp02ss.laheyhealth.org/csp/sys/op/%25CSP.UI.Portal.Mirror.Monitor.zen
- https://lvhshcadp01g1.laheyhealth.org/csp/sys/op/%25CSP.UI.Portal.Mirror.Monitor.zen
- https://lvhshcadp02g1.laheyhealth.org/csp/sys/op/%25CSP.UI.Portal.Mirror.Monitor.zen

Test

- https://lvhshcadt01ss.laheyhealth.org/csp/sys/op/%25CSP.UI.Portal.Mirror.Monitor.zen
- https://lvhshcadt02ss.laheyhealth.org/csp/sys/op/%25CSP.UI.Portal.Mirror.Monitor.zen
- https://lvhshcadt01g1.laheyhealth.org/csp/sys/op/%25CSP.UI.Portal.Mirror.Monitor.zen
- https://lvhshcadt02g1.laheyhealth.org/csp/sys/op/%25CSP.UI.Portal.Mirror.Monitor.zen

Development

- https://lvhshcadd01g1.laheyhealth.org/csp/sys/op/%25CSP.UI.Portal.Mirror.Monitor.zen
- https://lvhshcadd02g1.laheyhealth.org/csp/sys/op/%25CSP.UI.Portal.Mirror.Monitor.zen

Monitor

Configure severity level 2 (and higher) messages to be emailed via mailrelay.laheyhealth.org to a recipient list of email addresses:

```
$ sudo irisdb
%SYS> do ^MONMGR
```

Journals and WIJ

A separate filesystem for each dataset journaling and the instance write-image journal are mounted under the dataset filesystem as `jrn`:

```
Filesystem                    Type  Size  Used Avail Use% Mounted on
/dev/mapper/vg_data-lv_data xfs   1.2T  376G  853G  31% /hcprod-data
/dev/mapper/vg_jrn-lv_jrn   xfs   600G  177G  424G  30% /hcprod-data/jrn
```

NOTE: taking a storage snapshot off the volume group's physical volume with the datasets and journal activities provides for an improved crash-consistent copy for automated recovery on IRIS startup. Separating the physical volumes requires all of them to belong to a storage consistency group.

Journals are configured to switch when reaching 200mb to allow for faster viewing and profiling, and also for the potential for more granular point-in-time recovery. The journal compression feature is also enabled.

Journal purging is automated from `Task Manager` with an operational level of retention, accordingly.

Should unplanned utilization of journals exceed online capacity, IRIS alerts and switches to the alternate journal directory assigned within the media share subfolder: `jrn`

```
$ cd /mnt/autofs/media/jrn
$ ll
total 20
drwxrwsr-x 2 irisusr irisdev 26 Dec  6 10:25 d01g1
drwxrwsr-x 2 irisusr irisdev 26 Dec  6 10:26 d02g1
drwxrwsr-x 2 irisusr irisdev 26 Jan 10 16:41 p01g1
drwxrwsr-x 2 irisusr irisdev 26 Jan 10 16:24 p01ss
drwxrwsr-x 2 irisusr irisdev 26 Jan 10 15:54 p02g1
drwxrwsr-x 2 irisusr irisdev 26 Jan 10 16:39 p02ss
drwxrwsr-x 2 irisusr irisdev 26 Dec  6 10:27 t01g1
drwxrwsr-x 2 irisusr irisdev 26 Dec  6 11:00 t01ss
drwxrwsr-x 2 irisusr irisdev 26 Dec  6 10:30 t02g1
drwxrwsr-x 2 irisusr irisdev 26 Dec  6 11:01 t02ss
```

Backup

A daily Linux `cron` script executes off the IRIS `Backup` Mirror member only to perform a concurrent, non-disruptive IRIS dataset full backup onto its respective `instance` backup destination → an NFS folder share off a storage NAS that is policy-managed for replication and archiving.

/etc/cron.daily/backup

```
#!/bin/bash
/mnt/autofs/media/bin/backup.sh
EXITVALUE=$?
if [ $EXITVALUE != 0 ]; then
    /usr/bin/logger -t backup.sh "ALERT exited abnormally with [$EXITVALUE]"
fi
exit $EXITVALUE
```

/mnt/autofs/media/bin/backup.sh

```
… snipped …
sudo irisdb <<EOD
set media = "/mnt/autofs/media/bck/${INSTANCE}"
set TARGET = ##class(%Library.File).NormalizeFilename("AutoBackupFull_${DATE}.bck", media)
set LOGFILE = \$p(TARGET,".",1)_".log"
do BACKUP^DBACK("", "F", "${INSTANCE} auto backup", TARGET, "N", LOGFILE, "NOISY")
halt
EOD
```

Business Continuity

IRIS uses Mirroring to provide for classic Primary-Backup failover events. Without ECP, this is not an HA solution, but rather an automated sequence for:

1. Shutting down all interfaces on running productions
2. Shutting down the IRIS Primary instance and relinquishing its VIP
3. The IRIS Backup instance takes over as Primary with read-write datasets and activating the VIP
4. productions auto-start and enable interfaces

This usually takes a few minutes of downtime to orchestrate until completion.

If failover between data centers becomes necessary:

1. if possible, promote a DR instance to become the Primary Backup to follow business continuity sequencing
2. if not possible, select the DR instance to fail into, by using Make Primary
3. Once it is Primary, assign the DR VIP to the mirrorset, because they operate in separate VLANs
4. NetScaler GSLB with auto-detect that the new VIP with the IRIS SuperServer port that is in effect and will DNS resolve calls into hcie*.laheyhealth.org using the new VIP

Namespaces

Each Namespace is configured to have separated datasets as CODE for routines and DATA for globals, i.e.,

```
$ cd /hcprod-data/
$ ls BILHWORKDAY/*
BILHWORKDAY/code:
IRIS.DAT  iris.lck  stream
BILHWORKDAY/data:
IRIS.DAT  iris.lck  stream
```

It is important to follow this convention that was started by the original LAHEYHEALTH production, because the archive script separately traverses each Namespace CODE dataset to export the class source code each night.

The CODE and DATA datasets should also have their Stream directory redirected to use a subfolder (named exactly as the Namespace) in /files/STREAMS to allow for interfaces running in those Namespaces to link their file-based Ens.StreamContainer messages to the content stored in there. Like the IRIS databases, this is necessary to keep them logically consistent for any Primary-Backup failover event.

The %EnsRole_Administrator role is added to each database resource used by the Namespace with Interoperability registered, so DevOps accounts can switch and have the appropriate access rights to them.

NOTE: You can manually enable a Namespace as a Health Connect production by executing its class method:

```
%SYS> zwrite ##class(%EnsembleMgr).EnableNamespace("HSCUSTOM",1)
```

Namespace Mappings

Our Integration Team requires any custom HL7 definitions they define; and also, Ensemble Lookup Table dictionaries that start with the keyword Share are to be used between all Production Namespaces.

This is accomplished by identifying those data elements in the %All Namespace global mappings as captured here.

There is shared class code in the %All Namespace package mappings to manage one set of Common and Functions between Productions.

The shared, custom code and data is stored appropriately in the HSCUSTOM namespace.

System > Configuration > Namespaces > Global Mappings

Global Mappings   [New]   [Save Changes]

The global mappings for namespace %ALL are displayed below:

Filter: [   ]   Page size: 0   Max rows: 1000   Results: 19   Page: |<

Global	Subscript	Database	
Ens.LookupTable		%DEFAULTDB	Edit
Ens.LookupTable	("EpicFacID_to_OrgID")	HSCUSTOM	Edit
Ens.LookupTable	("Share."):("Share/")	HSCUSTOM	Edit
EnsHL7.Annotation		HSCUSTOM	Edit
EnsHL7.Annotation	("2."):("2/")	ENSLIB	Edit
EnsHL7.Annotation	("HealthShare_2.5")	HSLIB	Edit
EnsHL7.Annotation	("ITK")	ENSLIB	Edit
EnsHL7.Annotation	(2):(3)	ENSLIB	Edit
EnsHL7.Description		HSCUSTOM	Edit
EnsHL7.Description	("2."):("2/")	ENSLIB	Edit
EnsHL7.Description	("HealthShare_2.5")	HSLIB	Edit
EnsHL7.Description	("ITK")	ENSLIB	Edit
EnsHL7.Description	(2):(3)	ENSLIB	Edit
EnsHL7.Schema		HSCUSTOM	Edit
EnsHL7.Schema	("2."):("2/")	ENSLIB	Edit
EnsHL7.Schema	("HealthShare_2.5")	HSLIB	Edit
EnsHL7.Schema	("ITK")	ENSLIB	Edit
EnsHL7.Schema	(2):(3)	ENSLIB	Edit
WebTerminal		HSCUSTOM	Edit

SQL Gateway Connections

These data source names are provisioned for EDI DevOps to use, typically for a linked table or stored procedure need. You can manually inspect the IRIS table data that contain any DSN entries:

```
select * from  %Library.sys_SQLConnection
```

You must manually replicate any changes over to backup mirrors. That can be expedited out of its portal using `Explorer → SQL wizard: Data Export`

On the backup mirror, reverse the operation using `SQL wizard: Data Import` as exampled here:

1. Drop any connection entries first

```
delete from  %Library.sys_SQLConnection
```

2. Import the connection entries that were exported from the primary mirror

Infrastructure Service Line

Data Center Servers team

Topology

Boston Data Center	VIP address / NAT	Hostname	IP address
hcieprd	172.21.129.130	lvhshcadp01ss	172.21.129.131
	208.64.113.143	lvhshcadp02ss	172.21.129.132
hcietst	172.21.26.240	lvhshcadt01ss	172.21.26.241
	208.64.113.141	lvhshcadt02ss	172.21.26.242

Gordon Data Center	VIP address / NAT	Hostname	IP address
hcieprd	172.27.129.130	lvhshcadp01g1	172.27.129.131
	208.64.113.142	lvhshcadp02g1	172.27.129.132
hcietst	172.26.5.249	lvhshcadt01g1	172.26.5.254
	208.64.113.140	lvhshcadt02g1	172.26.5.251
hciedev	172.26.5.112	lvhshcadd01g1	172.26.5.253
		lvhshcadd02g1	172.26.5.252

The topology above is representative of a typical IRIS deployment below using a Failover Pair with geographically separated, fully redundant DR environment:
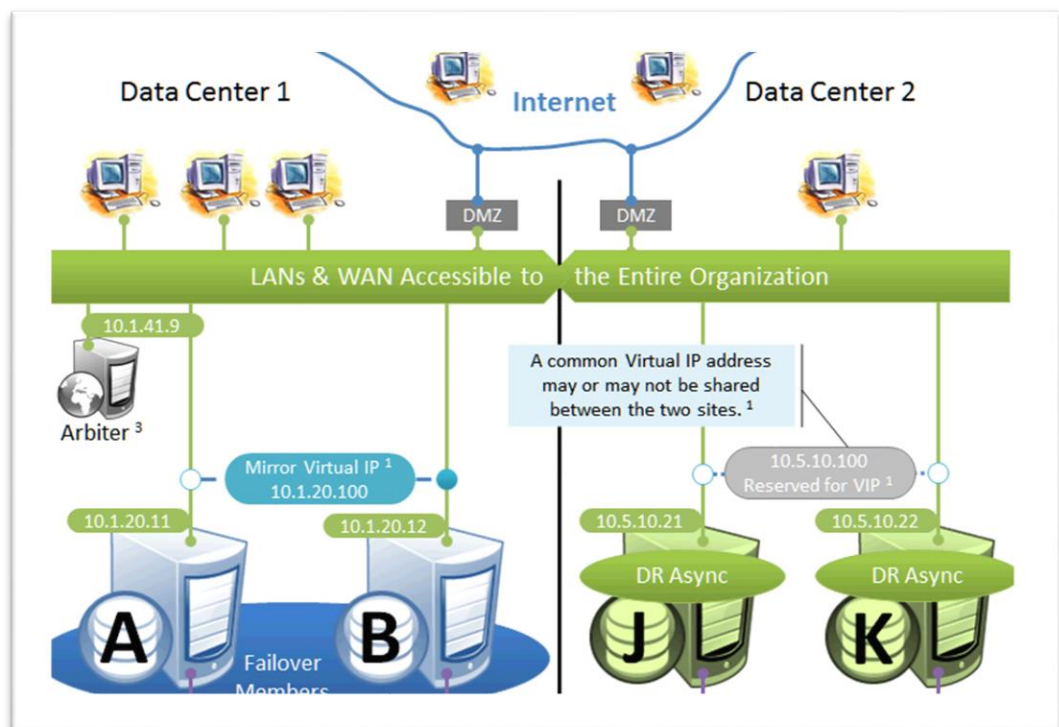
OS

Red Hat Enterprise Linux 8 running in VMware hypervisor farms.

/etc/sysctl.conf

```
# Allows for shorter, gratuitous client checks for an abnormal socket close
net.ipv4.tcp_fin_timeout = 20
net.ipv4.tcp_keepalive_time = 60
net.ipv4.conf.all.rp_filter=2
```

Single Sign-On

/etc/sssd/sssd.conf

```
[sssd]
domains = ad.laheyhealth.org
config_file_version = 2
services = nss, pam

[domain/ad.laheyhealth.org]
ad_domain = ad.laheyhealth.org
krb5_realm = AD.LAHEYHEALTH.ORG
realmd_tags = manages-system joined-with-adcli
cache_credentials = False
id_provider = ad
krb5_store_password_if_offline = True
default_shell = /bin/bash
override_shell = /bin/bash
shell_fallback = /bin/bash
ldap_id_mapping = True
use_fully_qualified_names = False
fallback_homedir = /home/%u
ignore_group_members = True
```

/etc/group

```
wheel:x:10:… list of AD user accounts for SysOps …
lahey:x:443800513:
sysadm:x:443817227:
irisusr:x:1007:apache
irisdev:x:1008:irisusr,… list of AD user accounts for DevOps …
irisadm:x:1009:… list of AD user accounts for DevOps administration …
```

The lahey and sysadm group entries here are simply mapped on its AD group id, which allows for a friendlier naming convention for Linux file / folder lists, ACLs, and scripting.

Access Controls

PAM

```
$ sudo authconfig --enablepamaccess --update
```

/etc/security/access.conf

```
- : ALL EXCEPT sysadm wheel irisusr irisdev: ALL
```

/etc/sudoers

```
# Defaults specification
Defaults               requiretty
Defaults:%irisusr      !requiretty
Defaults:nagios        !requiretty
Defaults:root          !requiretty
Defaults:%sysadm       !requiretty
Defaults:%wheel        !requiretty

## Same thing without a password
%wheel  ALL=(ALL)       NOPASSWD: ALL
%sysadm ALL=(ALL)       NOPASSWD: ALL
nagios  ALL=(ALL)       NOPASSWD: /usr/local/nagios/libexec/
%irisusr ALL=(ALL)      NOPASSWD: /bin/,/usr/bin/,/sbin/,/usr/sbin/,/mnt/autofs/media/bin/
%irisadm ALL=(ALL)      NOPASSWD:
/usr/bin/killall,/usr/bin/su,/usr/sbin/groupmems,/usr/sbin/usermod
```

Services

Apache 2.4

IRIS requires the following endpoints to use with their CSP gateway:

- `/api/` – instance public web services
- `/csp/` – management portal and production applications
- `/isc/` – vendor private web services

The service endpoint `/server-status` opens a snapshot of runtime metrics useful for trouble-shooting.

Cockpit – Files Navigator plug-in

```
$ sudo cp /mnt/autofs/media/etc/45drives.repo /etc/yum.repos.d/
$ sudo cp /mnt/autofs/media/etc/cockpit.conf /etc/cockpit/
$ sudo dnf install cockpit-navigator
```

ISCAgent and ISC@instance

These `systemd` services are enabled for automatic startup / shutdown.

`ISCAgent` service is a part of the IRIS instance install by the ODBA team, which is required for IRIS Mirroring.

`ISC@instance` is our site service for automatic startup / shutdown of the IRIS named instance, but also for SysOps to execute as needed:

```
$ sudo systemctl enable ISC@HCDEV
$ sudo systemctl start ISC@HCDEV
$ sudo systemctl stop ISC@HCDEV
$ sudo systemctl status ISC@HCDEV
```

/etc/passwd

Both of these services require a local non-root service account to startup:

```
iscagent:x:1002:1002::/home/iscagent:/bin/bash
irisusr:x:1007:1007::/home/irisusr:/bin/bash
```

Monitoring

Xymon is installed standard for constant host health checks on capacity and services.

Node.js

```
$ sudo dnf install @nodejs:18
$ sudo npm -g install @types/fs-extra carbone csvjson fs-extra typescript xml2js
$ sudo dnf install google-noto-fonts-common \
   google-noto-emoji-color-fonts google-noto-emoji-fonts \
   google-noto-mono-fonts google-noto-sans-fonts google-noto-serif-fonts
```

Storage

In addition to the complement of logical volume storage space allocated to run the host OS with services, the Health Connect instance requires these IRIS filesystems to accommodate:

- `/hc*-sys` – the instance install directory: bin, mgr, etc.
- `/hc*-data` – the datasets for each Namespace with code and data subfolders
- `/hc*-data/jrn` – the instance primary write image and dataset journals

NFS

There are two types of NFS v4 shares in use:

1. each instance has their own `files` for the Health Connect productions `DevOps` use
2. every Health Connect host shares `media` for `SysOps` use

These NFS v4 shares are provisioned by a storage NAS that has changes replicated between Primary and DR data centers once daily at 5:00am.

Service Credentials

These are the EDI delegated AD service accounts used to authorize access with infrastructure dependent on its participating domain:

- `CN=svc_hcdev,OU=Service Accounts,DC=ad,DC=laheyhealth,DC=org`
- `CN=svc_hcuat,OU=Service Accounts,DC=ad,DC=laheyhealth,DC=org`
- `CN=svc_hcprod,OU=Service Accounts,DC=ad,DC=laheyhealth,DC=org`

… or …

- `CN=svc-hcdev,OU=Service Accounts,OU=Service Accounts,DC=bilh,DC=itsystems,DC=org`
- `CN=svc-hcuat,OU=Service Accounts,OU=Service Accounts,DC=bilh,DC=itsystems,DC=org`
- `CN=svc-hcprod,OU=Service Accounts,OU=Service Accounts,DC=bilh,DC=itsystems,DC=org`

Backup

Networker manages archiving for each host on the following directories:

- `/etc` – the host configurations
- `/hc*-data/jrn` – the instance primary write image and dataset journals
- `/hc*-sys` -- the instance install directory: bin, mgr, etc.
- `/usr/local` – other locally-installed configs, libs, and scripts

SysOps

Preparation

The `Backup` and any `Disaster Recovery` members can and should be lifecycle managed prior to its scheduled `Primary` member failover event to its `Backup` member:

```
$ iris list
Configuration 'HCTEST'   (default)
        directory:    /hctest-sys
        versionid:    2022.1.4.812.0.22957
        datadir:      /hctest-sys
        conf file:    iris.cpf  (SuperServer port = 1972, WebServer = 52773)
        status:       running, since Wed Nov  6 09:17:04 2024
        mirroring: Member Type = Disaster Recovery; Status = Connected
        state:        ok
        product:      InterSystems HealthConnect
$ sudo systemctl stop ISC@HCTEST
$ sudo dnf update
$ sudo reboot
… do some post system startup sanity checks …
$ df -t nfs4
Filesystem                                  Type  Size  Used Avail Use% Mounted on
lhnas06:/ifs/burinas04/mix/EpicDMSTransfers     nfs4  1.4P  998T  376T  73% /LaheyNFS/EpicDMSTransfers
lhnas03.lahey.org:/ifs/bosinas03/nfs/hctest-files nfs4  1.8P  1.5P  255T  86% /hctest-files
lhnas03.lahey.org:/ifs/bosinas03/nfs/media      nfs4  1.8P  1.5P  255T  86% /mnt/autofs/media
$ df /files/cifs/{BILH-EDI,EDI,EpicEDI,Workday}
Filesystem                          Type  Size  Used Avail Use% Mounted on
//galaxy.bidmc.harvard.edu/clinical/BILH-EDI cifs  2.6P  1.8P  783T  70% /hctest-files/cifs/BILH-EDI
//galaxy.bidmc.harvard.edu/clinical/EDI      cifs  2.6P  1.8P  783T  70% /hctest-files/cifs/EDI
//epicnas-np/NPINT/EDI                       cifs  1.8P  1.5P  255T  86% /hctest-files/cifs/EpicEDI
//lhnas06/workday                            cifs  1.4P  1.1P  376T  74% /hctest-files/cifs/Workday
$ sudo systemctl status ISC@HCTEST
● ISC@HCTEST.service - InterSystems Corporation - HCTEST instance
   Loaded: loaded (/etc/systemd/system/ISC@.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2024-10-22 17:16:42 EDT; 3 weeks 1 days ago
  Process: 1431 ExecStart=/usr/local/etc/irissys/iris start HCTEST (code=exited,
status=0/SUCCESS)
 Main PID: 1431 (code=exited, status=0/SUCCESS)
    Tasks: 45 (limit: 100440)
   Memory: 5.9G
   CGroup: /system.slice/system-ISC.slice/ISC@HCTEST.service
… allow a little time for IRIS to resume its member type and mirror status …
$ iris list
```

Failover

The `Primary` member failover event to its *waiting to takeover* `Backup` member can take up to 5-minutes to complete. Log into both members, concurrently, with access to its VMware management console should it be needed.

NOTE: *you should always remote* `ssh` *into the* `hostname` *and not its* `VIP`.

Terminal session #1:

```
… check the current Backup status if it is ready to takeover …
$ iris list
Configuration 'HCTEST'   (default)
        directory:    /hctest-sys
        versionid:    2022.1.4.812.0.22957
        datadir:      /hctest-sys
        conf file:    iris.cpf  (SuperServer port = 1972, WebServer = 52773)
        status:       running, since Tue Oct 22 17:16:37 2024
        mirroring: Member Type = Failover; Status = Backup
        state:        ok
        product:      InterSystems HealthConnect
… you can watch IRIS event log throughout its takeover …
$ tail -f /hctest-sys/mgr/messages.log

…
11/12/24-21:01:56:303 (11201) 0 [Generic.Event] MirrorClient: Set status for TSTMIRROR to Transition
11/12/24-21:01:56:306 (10967) 0 [Utility.Event] Backup taking over due to clean shutdown of primary
…
11/12/24-21:02:15:895 (10967) 0 [Utility.Event] Initializing Interoperability during mirror initialization
…
11/12/24-21:02:15:895 (10967) 2 [Utility.Event] Becoming primary mirror server
11/12/24-21:02:22:124 (10981) 0 [Utility.Event] [SYSTEM MONITOR] Mirror status changed. Member type =
Failover, Status = Primary
```

Terminal session #2:

```
… check the current Primary status before stopping its instance …
$ iris list
Configuration 'HCTEST'   (default)
        directory:    /hctest-sys
        versionid:    2022.1.4.812.0.22957
        datadir:      /hctest-sys
        conf file:    iris.cpf  (SuperServer port = 1972, WebServer = 52773)
        status:       running, since Tue Oct 22 16:39:55 2024
        mirroring: Member Type = Failover; Status = Primary
        state:        ok
        product:      InterSystems HealthConnect
$ sudo systemctl stop ISC@HCTEST
… wait for an 'all clear' by the ODBA & EDI teams …
```

All Clear event

By allowing for the validation time spent between ODBA & EDI teams to be satisfied with their operations post failover, the former Primary host remains available to re-takeover – *prior to its lifecycle management* – should something unplanned occur.

Planned: move forward to complete on Terminal session #2:

```
$ sudo dnf update
$ sudo reboot
… repeat post system startup sanity checks as done before on backup …
$ sudo systemctl status ISC@HCTEST
$ tail -f /hctest-sys/mgr/messages.log
…
11/12/24-21:00:52:613 (10967) 0 [Utility.Event] (TSTMIRROR) Becoming a backup mirror member
…
11/12/24-21:00:54:427 (11201) 0 [Generic.Event] MirrorClient: The backup node has become active
```

Unplanned: rollback to return services, start on Terminal session #2:

```
$ sudo systemctl start ISC@HCTEST
$ iris list
$ tail -f /hctest-sys/mgr/messages.log
```

… and takedown the failed Primary member on Terminal session #1:

```
$ sudo systemctl stop ISC@HCTEST
… only if that fails to comply to release itself as Primary with its VIP …
$ sudo iris force HCTEST
$ reboot
```

NOTE: ODBA team may need to get involved in order to force the Backup member to takeover as Primary. That is the last resort, because all of the other mirrors in the set will likely be broken beyond repair and will then need to have their datasets restored from the last backup.

HCIE Working Examples

Business Service

Common

As detailed in DevOps, the custom Common`.Production.*` `business hosts` are used throughout the Ensemble Productions in order to better accommodate operational practices and business need.

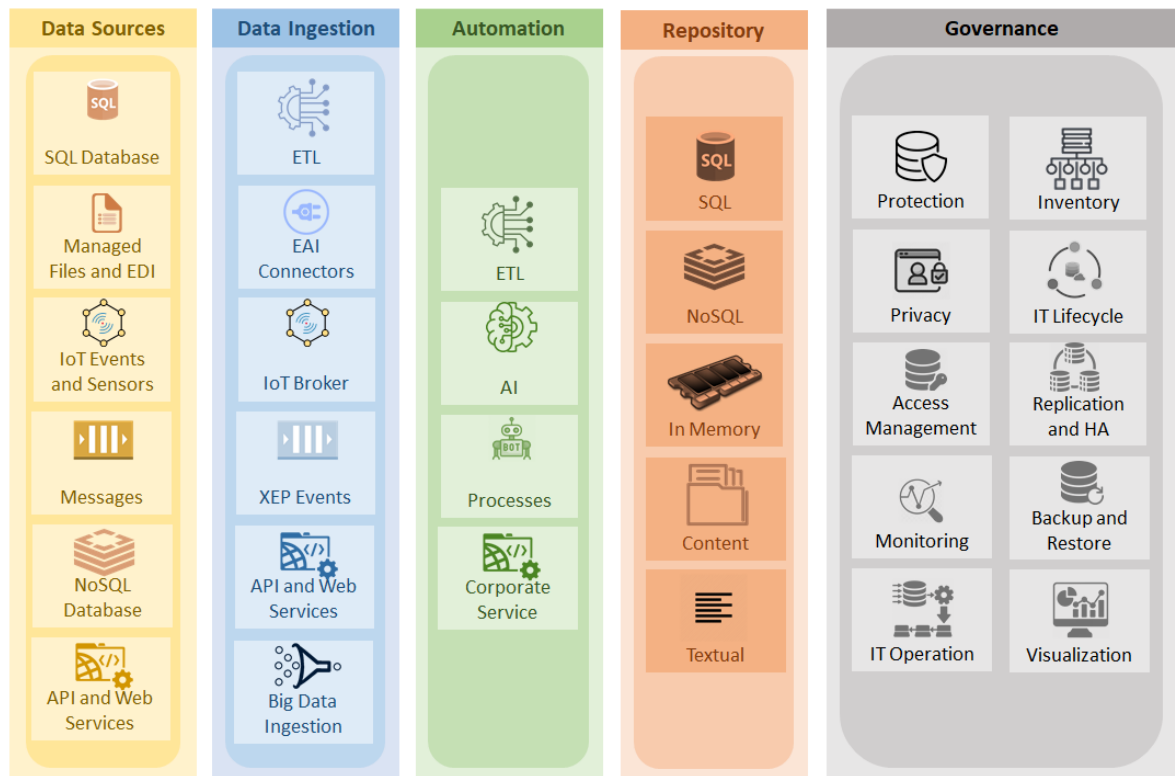- Required

Adapter-less

Business Process

Rules

DTL

BPL

Business Operation

IRIS Technology Overview

Corporate Data Architecture with Intersystems IRIS

Data Sources	Data Ingestion	Automation	Repository	Governance	
SQL Database	ETL		SQL	Protection	Inventory
Managed Files and EDI	EAI Connectors	ETL	NoSQL	Privacy	IT Lifecycle
IoT Events and Sensors	IoT Broker	AI	In Memory	Access Management	Replication and HA
Messages	XEP Events	Processes	Content	Monitoring	Backup and Restore
NoSQL Database	API and Web Services	Corporate Service	Textual	IT Operation	Visualization
API and Web Services	Big Data Ingestion				

Architecture mapping

Data Sources

- **SQL Database**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=GSQL
- **Managed Files**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=AFL_mft
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=SETEDIGuides
- **IoT Broker, Events and Sensors**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=EMQTT
- **Messages**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=EMQS
- **NoSQL**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=GDOCDB
- **API and Web Services**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=GREST
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=GSOAP
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=AFL_iam
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=PAGE_interoperability

Data Ingestion

- **ETL**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=SETAdapters
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=EDTL
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=EBPL
- **EAI Connectors**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=SETAdapters
- **XEP Events**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=BJAVXEP
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=BNETXEP
- **Big Data Ingestion**
  - https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=BSPK

Automation

- **AI**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=PAGE_text_analytics
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=APMML
  - https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=PAGE_python_native
  - https://www.intersystems.com/br/resources/detail/machine-learning-made-easy-intersystems-integratedml/
- **Processes**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=EBPL
- **Corporate Service**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=EESB
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=AFL_iam

Repository

- **In memory**
  - https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=GSCALE_ecp
- **Content**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=GDOCDB
- **Textual**
  - https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=AFL_textanalytics

Governance

- **Protection**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=SETSecurity
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=TSQS_Applications
  - https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=GCDI
  - https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=GCAS
- **Inventory**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=GSA_using_portal
  - https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=GOBJ_xdata
- **Privacy**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=GCAS_encrypt

- **IT Lifecycle, Backup and Restore**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=GSA_using_portal
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=GCDI_backup
- **Access Management**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=TSQS_Authentication
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=TSQS_Authorization
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=TSQS_Applications
- **Replication and HA**
  - https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=PAGE_high_availability
- **Monitoring**
  - https://docs.intersystems.com/sam/csp/docbook/DocBook.UI.Page.cls?KEY=ASAM
  - https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=PAGE_monitoring
- **IT Operation**
  - https://docs.intersystems.com/irislatest/csp/docbook/Doc.View.cls?KEY=PAGE_platform_mgmt
- **Visualization**
  - https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=PAGE_bi

HealthShare Platform – Architectural Diagram