

Syracuse University

Office of the Chief Information Officer

USPTO Systems Development Life Cycle

System Design Document for Supplemental Complex Repository for Examiners (SCORE)

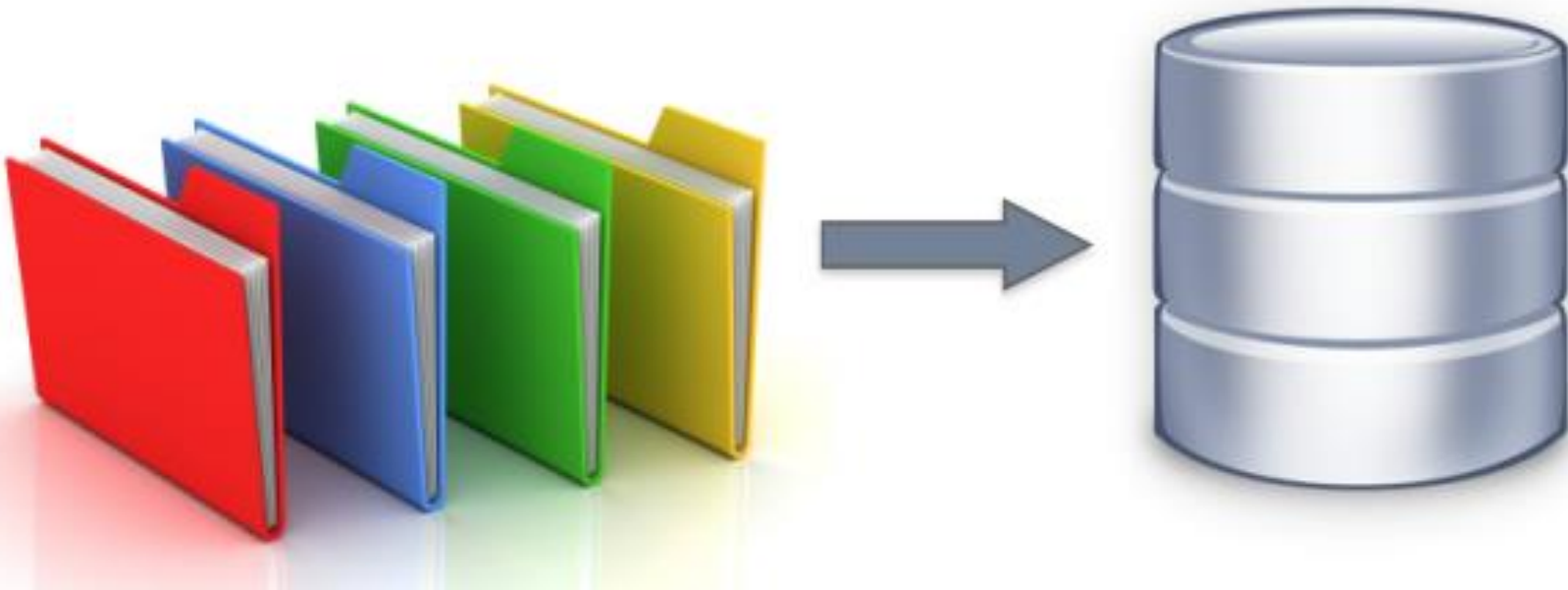


Table of Contents:

- **Business Description (PBO)**
- **Problem Description (PD)**
- **Solution Overview**
- **DATA-Modeling**
- **Business Rules**
- **ER-DIAGRAM**
- **Explanation of ER with cardinality**
- **Physical DataModel**
- **Data Questions**
- **Project Summary**
- **Appendix**
- **Glossary**

1. Business Description (PBO)

SCORE is a Web-based system developed under the direction of the United States Patent and Trademark Office (USPTO) as specified by the Chief Information Officer (CIO) for Automated Information Systems (AIS)

The purpose of the Supplemental Complex Repository for Examiners (SCORE) 3.2 is to provide users with access to unpublished non-image application data and files that cannot be scanned into the Image File Wrapper (IFW) repository in the tagged image file format (TIFF) because of their file size or type. These files contain sequence listings with millions of pages, tables, or biotechnology information that requires specific file types and has specific viewing requirements. USPTO patent examiners, other USPTO staff, and applicants can then use SCORE to access these application files. USPTO benefits from SCORE include consolidating data management and increasing efficiency in data flow and quality control.

This document provides details for the design of the Supplemental Complex Repository for Examiners (SCORE) 3.2 based on the system requirements approved by the United States Patent and Trademark Office (USPTO).

2. Problem Description (PD)

The intention behind kicking of this project is to overcome the current challenges users and business are facing with the system. Slag Performance and limited functionalities which are identified after the application is developed and went on live. Challenges or problem areas are described as below -

This application is using Cassandra database at back end to store metadata and file-store to store files. Cassandra is a non-relational database but it gives maximum up time because of its server architecture. It's through put is max and downtime is almost tends to zero. Even though it has good features, drawback is revealed a year later when new requirements started coming in.

The main challenge with Cassandra is to decide first what kind of queries can be asked against database. Once the database is designed and start using it, there no much changes can be done and no one can make any different kind of query against the database as it was not captured before datamodel was designed. Cassandra needs fields to be part of culture key to make query. Cassandra can't do any kind of joining or in query.

So ultimately business landed up in a bottleneck situation as they can't enhance functionalities. USPTO generates several reports and does heavy analysis on business data on Patents regular basis and they have different needs on time. Based on international treaty and users' need application functionalities gets changed frequently which need to be accommodate at earliest.

3. Solution Overview:

Chief Information Officer (CIO) for Automated Information Systems (AIS) decided to hire data analyst to find optimal database solution for the business needs with limited cost as they have already spent huge money on Cassandra.

USPTO was need a data solution would take minimum license cost, Data independence, efficient data access, Data integrity and security, Easy Data administration, Concurrent access and crash recovery and reduced application development time.

So as first step it was required to know how business performs and to understand their need daily basis. CIO and other stake holders like Scientific and Technology Information Center (STIC), Examiner (bio-sequence), end users were involved to explained their needs and describes minute attributes of the business process.

After lots of analysis Microsoft SQL server was proposed for optimal database solution. The best part of the proposed database is **cost (pretty pricey)**. It provides multiple level of security (which was lack in Cassandra). It provides excellent data recovery at Enterprise-grade management software. It supports all relational functionalities, no limitation on queries and any kind of

analysis can be done on data. MS SQL data can be pulled into MS Access for report generation and MS Access form wizard can be for designing user form and query designer can be used to make any complex query on relations with limited time and with limited knowledge on database query.

Requirement of the business process was provided by the stake holders at minute level. Some top few business problem are captured as below for which CIO is looking for new solution –

- Business want a dashboard which will give entire data in one glance which will display the relation and information on image or bio-sequence application for major version, minor version, their international WIPO standard, file format, decision of the application and publication status.
- How many applications are certified or rejected over time.
- How many Specific standard of bio-sequence have been submitted over time.
- How many bio-sequence patent has been approved over time.
- How many bio-sequence was edited over time.
- Projection of how many bio-sequence has been submitted after certain period.

Above analysis will help CIO to decide how much STIC or examiners are required time to time to support patent application and it will help them on resource optimization at the same time it helps PTO to respond to users query on their filing status for a patent.

To support above business needs, it was interpreted in database query term to project how it will solve the needs. The interpretation of business need in terms of database query is converted as below

- Complete overview report of Bio-Sequence from submission to export/publication
- How many documents are certified or rejected in last one month By application number
- How many ST26 was loaded into system in last one year by projected year
- How many certified sequencelisting have been exported to ABSS after march 1st 2019.
- How many sequencelisting header have been edited in last one yea
- Provide the list of the sequencelisting or total count of sequencelisting are in the system have not been reviewed yet where sequencelisting is loaded after Jan 1, 2019

4. DATA-Modeling:

All the data CIO expects to store and display in dashboard are metadata of the application. But it was the responsibility of data analyst to understand the data, its use and its patterns. So gathered all the data and started removing multivalued dependencies separate tuples. Then all functional dependencies are identified and group of functions are separated out in different tuples. All transitional dependencies (where non key fields are depends on another non key attributes) and separate out in different tuple without dropping it out to avoid data loss. So finally business needs are captured in conceptual data model → logical data model.

5. Business Rules:

In this Bio-sequence business process when application is filed USPTO does not change/modify anything in the original file. System stores this as 1.0 version. System created another copy of it and split the file into .header, .body and .index files and it created version 1.1. Every submitted bio-sequence file should follow some WIPO standard ST23/25/26. Every format is having specific category of file like ST23/25 are text files ST26 are XML files. USPTO assigns SEQ.TXT.SUPP and SEQ.XML.SUPP to identify ST23/25 and ST26 respectively. After application is submitted for patent PTO assigns an application number for applicant. The application can use multiple source to submit the file LIKE EFSWeb (Electronic filing system), OIPE (Office of Initial Patent Examination). IB (International bureau) e.t.c. Each upstream system assigns system generated submission number to the application so that filling process can be tracked down through the process. Before a patent is issued by USPTO to an application files are reviewed by examiners/STIC. They may approve or reject the application. If they approve/certify it is called CRFE and if they reject it is called CRFD. If there is any issue in the application header, examiner is allowed to just change the application header but not the body part of the application because the body explains the details for which application is filed. Once the header is edited by examiner new document is stored in the system as next minor version 1.2. If there is application is rejected USPTO send mail to application on with show cause. If the application is certified by STIC/examiner then patent is approved and it is published (in technical term application is exported to publication site)

Based on the above business process/rules below are the identified tuples/tables following 1-2-3 normal form to avoid data redundancy and better storage optimization so that database query could be faster.

stnd_bio_sequence_format (Where the format of the ST23/25/26 is defined)

stnd_document_code (Category of file is defined with doc code like SEQ.TXT.SUPP and SEQ.XML.SUPP to identify ST23/25 and ST26 respectively)

stnd_source_system (Which system is used to submit application like EFSWeb, IB, OIPE)

stnd_document_decision (Standard Decision code is on filed application)

stnd_export_system (Publication site where application is published after patent is approved)

document_version

document_subversion

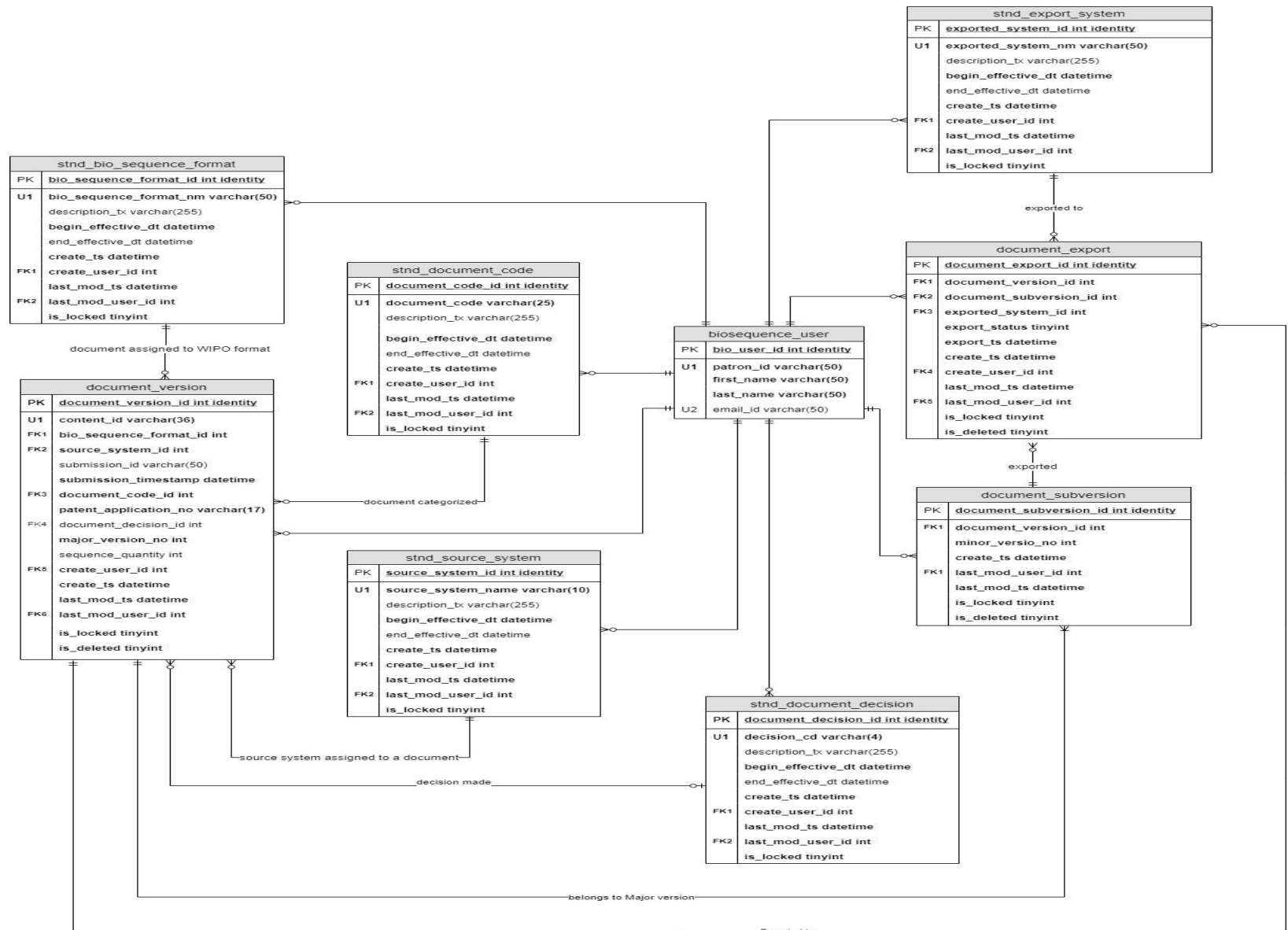
publication captured

document_export

biosequence_user (USPTO stake holders who are involved throughout the process)

} These are the three main table where application information, minor version of the application

6. ER-DIAGRAM:



7. Explanation of ER with cardinality:

stnd_bio_sequence_format – It captures standard format of the bio-sequence by USPTO users. It has effective period. One user can create multiple format in the system. One standard format can be assign to multiple document_version. But one document can have only one format.

stnd_document_code - It captures standard document code of the bio-sequence by USPTO users. It has effective period. One user can create multiple doc code in the system. One standard doc code can be assign to multiple document_version. But one document can have only one doc code.

stnd_source_system - It captures standard source system name of the bio-sequence by USPTO users. It has effective period. One user can create multiple source in the system. Application can be filed only through one system. But multiple application can be filed from same source system.

stnd_document_decision - It has standard decision code of the bio-sequence by USPTO users. It has effective period. One user can create multiple decision code in the system. One decision code can be assign to multiple document_version. But one document can have only one decision code.

stnd_export_system - It has standard publication site information of the bio-sequence by USPTO users. It has effective period. One user can create multiple publication site in the system.

document_version – Main tuple. Captures content reference information to locate the applicant file. It captures bio-sequence format, sourcesystem, submission id, submission time stamp, document code, application number, document decision, total sequence in the application, major version. One user can create multiple document_version but one document_version is created by one user.

document_subversion - Mainly it captures the minor version of the application for the same major version. One user can create multiple document_subversion but one document_subversion is created by one user.

document_export – It captures which major, minor/sub-version, when and where (publication site) and by whom it is done, when the export package is created. One user can do multiple exports but one specific is done by one user. **One publication site can be assign to multiple application/exports. But one application-export can be made to a specific publication site but same application can be published to multiple site.**

Note: It was a challenge for USPTO to capture this relationship. Which has been resolved through this solution. It will be available in dashboard later part of the document.

biosequence_user – This tuple is having Bio-Sequence stakeholder information (patron id, first name, last name email id) who are involved throughout the process.

Keys -

PK – Primary Key (auto generated by SQL server increment by 1)

FK[i] – Foreign Key, [i] is the index of the foreign key.

U[i] – Unique Key, [i] is the index of the Unique key.

Cardinalities -

➤ 0 or Many cardinality

‡ One cardinality

8. Physical DataModel: (described major attributes with data type and constraints)

biosequence_user:

```
bio_user_id    int identity, //Auto incremental primary key maintained by DB
patron_id      varchar(50) not null, //Patron ID unique for each person
first_name     varchar(50) not null, //First name of the user
last_name      varchar(50) not null, //Last name of the user
email_id       varchar(50),      //email ID
```

-- Constraints on the User Table

```
CONSTRAINT pk_biosequence_user PRIMARY KEY (bio_user_id),
CONSTRAINT u1_biosequence_user UNIQUE(patron_id),
CONSTRAINT u2_biosequence_user UNIQUE(email_id)
```

stnd_document_code:

```
document_code_id    int identity, //Auto incremental primary key maintained by DB
document_code       varchar(25) NOT NULL, //doc code SEQ.TXT.SUPP etc.
description_tx       varchar(255), //Description of the doc code
begin_effective_dt   datetime NOT NULL, //doc code effective date
end_effective_dt     datetime, //doc code expire date
create_ts           datetime DEFAULT CURRENT_TIMESTAMP NOT NULL,
create_user_id       int NOT NULL,
last_mod_ts         datetime DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_user_id     int NOT NULL,
is_locked           tinyint DEFAULT 0 NOT NULL,
```

-- Constraints on the document code Table

```
CONSTRAINT pk_stnd_document_code PRIMARY KEY (document_code_id),
CONSTRAINT u1_stnd_document_code UNIQUE (document_code),
CONSTRAINT fk1_stnd_document_code FOREIGN KEY (create_user_id) References
```

biosequence_user(bio_user_id),

```
CONSTRAINT fk2_stnd_document_code FOREIGN KEY (last_mod_user_id) References
```

biosequence_user(bio_user_id)

stnd_bio_sequence_format:

```
bio_sequence_format_id    int identity, // Auto incremental primary key maintained by DB
bio_sequence_format_nm     varchar(50) NOT NULL, // format name like ST23
description_tx             varchar(255), // format description
begin_effective_dt         datetime NOT NULL, // format effective date
end_effective_dt           datetime, // format expire date
create_ts                 datetime DEFAULT CURRENT_TIMESTAMP NOT NULL,
create_user_id             int NOT NULL,
last_mod_ts               datetime DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_user_id          int NOT NULL,
is_locked                 tinyint DEFAULT 0 NOT NULL,
```



```

-- Constraints on the sequence format Table
    CONSTRAINT pk_stdn_bio_sequence_format PRIMARY KEY (bio_sequence_format_id),
    CONSTRAINT u1_stdn_bio_sequence_format UNIQUE (bio_sequence_format_nm),
    CONSTRAINT fk1_stdn_bio_sequence_format FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),
    CONSTRAINT fk2_stdn_bio_sequence_format FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)

```

stdn_document_decision:

```

document_decision_id int identity,          // Auto incremental primary key maintained by DB
decision_cd          varchar(4)            NOT NULL, // decision code like CRFE
description_tx        varchar(255),         // Decision description
begin_effective_dt    datetime             NOT NULL, // decision code effective date
end_effective_dt      datetime             // decision code expire date
create_ts            datetime              DEFAULT CURRENT_TIMESTAMP NOT NULL,
create_user_id        int                  NOT NULL,
last_mod_ts          datetime              DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_user_id      int                  NOT NULL,
is_locked            tinyint               DEFAULT 0 NOT NULL,

```

-- Constraints on the document decision Table

```

    CONSTRAINT pk_stdn_document_decision PRIMARY KEY (document_decision_id),
    CONSTRAINT u1_stdn_document_decision UNIQUE (decision_cd),
    CONSTRAINT fk1_stdn_document_decision FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),
    CONSTRAINT fk2_stdn_document_decision FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)

```

stdn_source_system:

```

source_system_id      int identity,          // Auto incremental primary key maintained by DB
source_system_name     varchar(10)          NOT NULL, // Source system name
description_tx         varchar(255),         // Source system description
begin_effective_dt     datetime             NOT NULL, // Source system effective date
end_effective_dt       datetime             // Source system effective date
create_ts             datetime              DEFAULT CURRENT_TIMESTAMP NOT NULL,
create_user_id         int                  NOT NULL,
last_mod_ts           datetime              DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_user_id       int                  NOT NULL,
is_locked             tinyint               DEFAULT 0 NOT NULL,

```

-- Constraints on the Source system

```

    CONSTRAINT pk_stdn_source_system PRIMARY KEY (source_system_id),
    CONSTRAINT u1_stdn_source_system UNIQUE (source_system_name),
    CONSTRAINT fk1_stdn_source_system FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),
    CONSTRAINT fk2_stdn_source_system FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)

```

stdn_export_system:

```

exported_system_id     int identity,          // Auto incremental primary key maintained by DB

```

```

exported_system_nm    varchar(50)      NOT NULL,           //Exported system name/publication site
description_tx        varchar(255),    // description of the publication site
begin_effective_dt    datetime        NOT NULL,           // publication site effective date
end_effective_dt      datetime,        // publication site expiry date
create_ts            datetime        DEFAULT CURRENT_TIMESTAMP NOT NULL,
create_user_id        int             NOT NULL,
last_mod_ts          datetime        DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_user_id      int             NOT NULL,
is_locked            tinyint         DEFAULT 0 NOT NULL,

-- Constraints on the export system
CONSTRAINT pk_stnd_export_system PRIMARY KEY (exported_system_id),
CONSTRAINT u1_stnd_export_system UNIQUE (exported_system_nm),
CONSTRAINT fk1_stnd_export_system FOREIGN KEY (create_user_id)      References
biosequence_user(bio_user_id),
CONSTRAINT fk2_stnd_export_system FOREIGN KEY (last_mod_user_id)    References
biosequence_user(bio_user_id)

document_version:
document_version_id    int identity,    // Auto incremental primary key maintained by DB
content_id             varchar(36)     NOT NULL, // Unique to all content file
bio_sequence_format_id int            NOT NULL, //refer to format id
source_system_id       int            NOT NULL, //refer to system id
submission_id          varchar(50),    // submission id assigned by upstream system.
submission_timestamp   datetime       NOT NULL, // submission time stamp/mailroom date
document_code_id       int            NOT NULL, // refer to standard document code.
patent_application_no  varchar(17)    NOT NULL, // application number assigned to file.
document_decision_id   int,           // refer to decision code like CRFE/CRFD
major_version_no       int            DEFAULT -1 NOT NULL, //Major version for each file like 1 or 2
sequence_quantity      int,           //Number of sequence for the bio-sequence
create_user_id         int            NOT NULL,
create_ts              datetime       DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_ts            datetime       DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_user_id       int            NOT NULL,
is_locked              tinyint        DEFAULT 0 NOT NULL,
is_deleted              tinyint        DEFAULT 0 NOT NULL,

-- Constraints on the document version
CONSTRAINT pk_document_version PRIMARY KEY (document_version_id),
CONSTRAINT u1_document_version UNIQUE (content_id),
CONSTRAINT fk1_document_version FOREIGN KEY (bio_sequence_format_id) References
stnd_bio_sequence_format(bio_sequence_format_id),
CONSTRAINT fk2_document_version FOREIGN KEY (source_system_id)      References
stnd_source_system(source_system_id),
CONSTRAINT fk3_document_version FOREIGN KEY (document_code_id)      References
stnd_document_code(document_code_id),
CONSTRAINT fk4_document_version FOREIGN KEY (document_decision_id)  References
stnd_document_decision(document_decision_id),
CONSTRAINT fk5_document_version FOREIGN KEY (create_user_id)        References
biosequence_user(bio_user_id),

```

```

        CONSTRAINT fk6_document_version FOREIGN KEY (last_mod_user_id)
biosequence_user(bio_user_id)
References

```

document_subversion:

```

document_subversion_id    int identity, // Auto incremental primary key maintained by DB
document_version_id       int           NOT NULL, // Refer each minor version under major
minor_version_no          int           NOT NULL, // minor version number
create_ts                 DATETIME      DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_ts               DATETIME      DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_user_id          int           NOT NULL,
is_locked                 tinyint       DEFAULT 0 NOT NULL,
is_deleted                tinyint       DEFAULT 0 NOT NULL,

```

-- Constraints on the document subversion Table

```

        CONSTRAINT pk_document_subversion PRIMARY KEY (document_subversion_id),
        CONSTRAINT fk1_document_subversion FOREIGN KEY (document_version_id) References
document_version(document_version_id),
        CONSTRAINT fk2_document_subversion FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)

```

document_export:

```

document_export_id        int identity, // Auto incremental primary key maintained by DB
document_version_id       int           NOT NULL, // refer to document version
document_subversion_id    int           NOT NULL, // refer to document subversion
exported_system_id        int           NOT NULL, // refer where export is done
export_status             tinyint       NOT NULL, // export status success(1)/failure(0)
export_ts                 datetime      NOT NULL,
create_ts                 datetime      DEFAULT CURRENT_TIMESTAMP NOT NULL,
create_user_id            int           NOT NULL,
last_mod_ts               datetime      DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_user_id          int           NOT NULL,
is_locked                 tinyint       DEFAULT 0 NOT NULL,
is_deleted                tinyint       DEFAULT 0 NOT NULL,

```

-- Constraints on the document export

```

        CONSTRAINT pk_document_export PRIMARY KEY (document_export_id),
        CONSTRAINT fk1_document_export FOREIGN KEY (document_version_id) References
document_version(document_version_id),
        CONSTRAINT fk2_document_export FOREIGN KEY (document_subversion_id) References
document_subversion(document_subversion_id),
        CONSTRAINT fk3_document_export FOREIGN KEY (exported_system_id) References
std_export_system(exported_system_id),
        CONSTRAINT fk4_document_export FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),
        CONSTRAINT fk5_document_export FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id),

```

9. Data Questions (codes and additional screen is in appendix)

Form is designed to enter data in the standard look up table (stnd_export_system).

The screenshot displays the Microsoft Access application window. The title bar includes 'File', 'Home', 'Create', 'External Data', 'Database Tools', and a search bar. The ribbon contains various tabs like 'Clipboard', 'Filter', 'Records', 'Find', 'Text Formatting', and 'Query1'. The 'All Access Objects' pane on the left lists tables, queries, and forms. The 'dbo_stnd_export_system' form is open in 'Form View'. The form fields are as follows:

Field Name	Value
exported_system_id	
exported_system_nm	ABBSS
description_tx	AUTOMATED BIOTECHNOLOGY SEQUENCE SEARCH SYSTEM
begin_effective_dt	1/1/1970
end_effective_dt	12/31/2030
create_ts	3/28/2019 4:38:19 PM
create_user_id	Debasis Chatterjee
last_mod_ts	3/28/2019 4:38:19 PM
last_mod_user_id	Debasis Chatterjee
is_locked	0

The status bar at the bottom shows 'Form View', 'Records: 14 of 5', and a search bar. The Windows taskbar at the very bottom shows the system clock as 11:36 PM on 3/28/2019.

For the above business data questions data entry form is generated in MS Access. The same can be used for data modification. This form is designed to enter data in the document_version table. Where format, system, doc code, decision is coming from standard look up tables in a drop down.

Query1Query2dbo_document_versiondbo_stnd_export_systemdbo_document_version

dbo_document_version

content_id

cd123456

bio_sequence_format_id

ST23

source_system_id

EFSWEB

submission_id

s0091827123

document_code_id

SEQ.TXT.SUPP

document_decision_id

CRFE

major_version_no

1

sequence_qtquantity

102

Record is entered where source system 1 means EFSWEB.

document_v	content_id	bio_sequence	source_system	submission_id
1	cd123456	1	1	s0091827123
2	cd12345a	2	2	s0091827124
3	cd12345b	3	2	s0091827125
4	cd12345c	3	3	s0091827127
(New)				

Source system is updated to OIPE (2) from the form and it updated the back end database

document_v	content_id	bio_sequence	source_system	submission_id
1	cd123456	1	2	s0091827123
2	cd12345a	2	2	s0091827124
3	cd12345b	3	2	s0091827125
4	cd12345c	3	3	s0091827127
(New)				

Complete form with Sub-form is designed in MS access where data can be entered document_version and document_subversion table together in a single form.

dbo_document_version

content_id

cd123456

bio_sequence_format_id

ST23

source_system_id

SLIC

submission_id

s0091827123

document_code_id

SEQ.TXT.SUPP

document_decision_id

CRFE

major_version_no

1

sequence_quantity

102

dbo_document_subversion subform

minor_version_no

last_mod_user_id

2

patent_application_no

14289012

Record: 1 of 3

No Filter

Search

- Complete overview report of Bio-Sequence from submission to export/publication (This DASHBOARD is generated in MS Access)

Cut

Copy

Format Painter

Filter

Remove Sort

Toggle Filter

Refresh All

Delete

More

Ascending

Descending

Advanced

Save

Spelling

More

Find

Go To

Select

Calibri

11

</

This gives the complete information about the application when, what application from which system, when patent is approved and who published. This is a token dashboard to show that Kind of relational business data can be projected through dash board. Basically at back end it joins all the relational data and display the common set which is finally published or got patent.

Complete report on a particular Application designed in MS Access to show how report can be generate on this data model.

dbo_document_subversion

content_id	cd123456
submission_id	s0091827123
submission_timestamp	12/28/2018
patent_application_no	14289012
major_version_no	1
sequence_qtquantity	102
minor_version_no	2
export_status	1
export_ts	2/2/2019
bio_sequence_format_nm	ST23
document_code	SEQ.TXT.SUPP
decision_cd	CRFE
exported_system_nm	ABSS
source_system_name	SLIC

content_id	cd123456
submission_id	s0091827123
submission_timestamp	12/28/2018
patent_application_no	14289012
major_version_no	1
sequence_qtquantity	102
minor_version_no	2

export_status	1
export_ts	3/2/2019
bio_sequence_format_nm	ST23
document_code	SEQ.TXT.SUPP
decision_cd	CRFE
exported_system_nm	IDC
source_system_name	SLIC

- How many documents are certified or rejected in last one month By application number

The screenshot shows a SQL Server Enterprise Manager interface. The left pane displays the 'Object Explorer' with a tree view of the database structure, including tables like 'dbo.biobase_user', 'dbo.document_export', 'dbo.document_subversion', 'dbo.document_version', 'dbo.lab_Log', 'dbo.lab_Test', 'dbo.MyNewTable', 'dbo.stnd_bio_sequence', 'dbo.stnd_document_code', 'dbo.stnd_document_decisi', 'dbo.stnd_export_system', 'dbo.stnd_source_system', 'dbo.vc_FollowerList', 'dbo.vc_Status', 'dbo.vc_Tag', 'dbo.vc_User', 'dbo.vc_UserLogin', 'dbo.vc_UserTagList', 'dbo.vc_VidCast', and 'dbo.vc_VidCastTagList'. The main pane shows a query window with the following T-SQL code:

```

38 SELECT * FROM biobase_exports_projection
39
40 --How many documents are certified or rejected in last one month
41 SELECT dbo.document_version.major_version_no, dbo.stnd_document_decision.decision_cd, MAX(dbo.document_subversion.minor_version_no)
42 FROM (dbo.stnd_document_decision
43 INNER JOIN dbo.document_version ON dbo.stnd_document_decision.document_decision_id = dbo.document_version.document_decision_id)
44 INNER JOIN dbo.document_subversion ON dbo.document_version.document_version_id = dbo.document_subversion.document_version_id
45 GROUP BY dbo.document_version.major_version_no, dbo.stnd_document_decision.decision_cd;
46
47 --How many documents are certified or rejected in last one month By application number
48 CREATE VIEW count_process_seq_by_month AS
49 SELECT dbo.document_version.patent_application_no, dbo.document_version.major_version_no, dbo.stnd_document_decision.decision_cd, MAX(dbo.document_subversion.minor_version_no)
50 FROM (dbo.stnd_document_decision
51 INNER JOIN dbo.document_version ON dbo.stnd_document_decision.document_decision_id = dbo.document_version.document_decision_id)
52 INNER JOIN dbo.document_subversion ON dbo.document_version.document_version_id = dbo.document_subversion.document_version_id
53 WHERE dbo.document_subversion.last_mod_ts >= DATEADD(mm, DATEDIFF(mm, 0, GETDATE()), -1, 0) AND dbo.document_subversion.last_mod_ts <= DATEADD(mm, DATEDIFF(mm, 0, GETDATE()), 1, 0)
54 GROUP BY dbo.document_version.patent_application_no, dbo.document_version.major_version_no, dbo.stnd_document_decision.decision_cd;
55
56
57 CREATE FUNCTION dbo.count_certify_Reject_by_Last_month()
58 RETURNS int AS
59 BEGIN
60 DECLARE @returnValue int
61 SELECT @returnValue = count_process_seq_by_month.processed_sequencelisting_no FROM count_process_seq_by_month
62 RETURN @returnValue
63 END
64 GO
65
66 SELECT
67     dbo.count_certify_Reject_by_Last_month() as total_processed_sequence
68 FROM count_process_seq_by_month
69
70
71 -- How many ST26 was loaded into system in last one year by projected year
72 CREATE PROCEDURE loadCountST26 (@seqFormat VARCHAR(10))

```

The results pane shows a single row with the value 1 for the 'total_processed_sequence' column.

total_processed_sequence
1

Query executed successfully. ist-s-students.syr.edu (12... AD\dchatter (81) IST659_M403_dchatter 00:00:00 1 rows

- How many ST26 was loaded into system in last one year by projected year

The screenshot shows a SQL Server Enterprise Manager interface. The left pane displays the 'Object Explorer' with a tree view of the database structure, including tables like 'dbo.biobase_user', 'dbo.document_export', 'dbo.document_subversion', 'dbo.document_version', 'dbo.lab_Log', 'dbo.lab_Test', 'dbo.MyNewTable', 'dbo.stnd_bio_sequence', 'dbo.stnd_document_code', 'dbo.stnd_document_decisi', 'dbo.stnd_export_system', 'dbo.stnd_source_system', 'dbo.vc_FollowerList', 'dbo.vc_Status', 'dbo.vc_Tag', 'dbo.vc_User', 'dbo.vc_UserLogin', 'dbo.vc_UserTagList', 'dbo.vc_VidCast', and 'dbo.vc_VidCastTagList'. The main pane shows a query window with the following T-SQL code:

```

62 RETURN @returnValue
63 END
64 GO
65
66 SELECT
67     dbo.count_certify_Reject_by_Last_month() as total_processed_sequence
68 FROM count_process_seq_by_month
69 GO
70
71 -- How many ST26 was loaded into system in last one year by projected year
72 CREATE PROCEDURE loadCountST26 (@seqFormat VARCHAR(10))
73 AS
74 BEGIN
75 SELECT COUNT(dbo.document_version.document_version_id) as total_count_ST26,
76     dbo.stnd_bio_sequence_format.bio_sequence_format_nm, YEAR(dbo.document_version.create_ts) as projected_year
77 FROM dbo.stnd_bio_sequence_format
78 INNER JOIN dbo.document_version ON dbo.stnd_bio_sequence_format.bio_sequence_format_id = dbo.document_version.bio_sequence_format_id
79 WHERE dbo.stnd_bio_sequence_format.bio_sequence_format_nm=@seqFormat
80 AND dbo.document_version.create_ts > DATEADD(YEAR, -1, GETDATE())
81 GROUP BY dbo.stnd_bio_sequence_format.bio_sequence_format_nm, YEAR(dbo.document_version.create_ts);
82 END
83 GO
84 DECLARE @seqFormat VARCHAR(10)
85 SET @seqFormat = 'ST26'
86 EXEC loadCountST26 @seqFormat
87
88 -- How many certified sequencelisting have been exported to AB55 after march 1st 2019.
89 CREATE FUNCTION dbo.exported_seq_from_date (@fromDate DATETIME, @exportedSystem VARCHAR(10))
90 RETURNS int AS
91 BEGIN
92 DECLARE @returnValue int
93 SELECT @returnValue = COUNT(*) FROM biobase_exports_projection
94 WHERE dbo.biobase_exports_projection.exported_system_nm = @exportedSystem AND dbo.biobase_exports_projection.export_ts >= @fromDate
95 RETURN @returnValue
96

```

The results pane shows a single row with the values 2, ST26, and 2019 for the columns 'total_count_ST26', 'bio_sequence_format_nm', and 'projected_year' respectively.

total_count_ST26	bio_sequence_format_nm	projected_year
2	ST26	2019

Query executed successfully. ist-s-students.syr.edu (12... AD\dchatter (81) IST659_M403_dchatter 00:00:00 1 rows

- How many certified sequencelisting have been exported to ABSS after march 1st 2019.

The screenshot shows the SQL Server Enterprise Manager interface. The Object Explorer on the left displays the database structure, including tables like `dbo.stnd_bio_sequence_format` and `dbo.document_version`. The central pane shows a SQL query being executed. The query defines a function `dbo.exported_seq_from_date` that counts sequences exported after a specified date. The results pane at the bottom shows a single row with the value `1` for `total_abss_exported_sequence`.

```

76  dbo.stnd_bio_sequence_format.bio_sequence_format_nm, YEAR(dbo.document_version.create_ts) as projected_year
77  FROM dbo.stnd_bio_sequence_format
78  INNER JOIN dbo.document_version ON dbo.stnd_bio_sequence_format.bio_sequence_format_id = dbo.document_version.bio_sequence_format_id
79  WHERE dbo.stnd_bio_sequence_format.bio_sequence_format_nm = @seqFormat
80  and dbo.document_version.create_ts > DATEADD(YEAR, 1, GETDATE())
81  GROUP BY dbo.stnd_bio_sequence_format.bio_sequence_format_nm, YEAR(dbo.document_version.create_ts);
82  END
83  GO
84  DECLARE @seqFormat VARCHAR(10)
85  SET @seqFormat = 'ST26'
86  EXEC loadCountST26 @seqFormat
87
88  -- How many certified sequencelisting have been exported to ABSS after march 1st 2019.
89  CREATE FUNCTION dbo.exported_seq_from_date(@FromDate DATETIME, @exportedSystem VARCHAR(10))
90  RETURNS int AS
91  BEGIN
92      DECLARE @returnValue int
93      SELECT @returnValue = COUNT(*) FROM biosequence_exports_projection
94      WHERE biosequence_exports_projection.exported_system_nm = @exportedSystem and dbo.biosequence_exports_projection.export_ts > @FromDate
95      RETURN @returnValue
96  END
97  GO
98  DECLARE @FromDate DATETIME
99  SET @FromDate = '2019-03-01'
100 DECLARE @exportedSystem VARCHAR(10)
101 SET @exportedSystem = 'ABSS'
102 SELECT
103     dbo.exported_seq_from_date(@FromDate, @exportedSystem) as total_abss_exported_sequence
104 FROM count_process_seq_by_month
105 GO
106
107 --How many sequencelisting header have been edited in last one year
108
109 CREATE PROCEDURE editHeaderSequencelistingCount(@minorVer int)
110 AS
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
```

- Provide the list of the sequencelisting or total count of sequencelisting are in the system have not been reviewed yet where sequencelisting is loaded after Jan 1, 2019

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The left pane shows the 'Object Explorer' with a tree view of the database structure, including tables like 'dbo.bioprocess_user', 'dbo.document_export', and 'dbo.document_subversion'. The central pane shows a SQL query window with the following code:

```

114 WHERE dbo.document_subversion.minor_version_no = @minorVer
115 and dbo.document_subversion.create_ts > DATEADD(WEEK, -8, GETDATE())
116 END
117 GO
118 DECLARE @minorVer int
119 SET @minorVer = '2'
120 EXEC editHeaderSequencelistingCount @minorVer
121
122
123
124 -- Provide the list of the sequencelisting or total count of sequencelisting are in the system have not been reviewed yet where sequencelisting is loaded
125 CREATE VIEW unprocessedSequencelisting AS
126 SELECT * FROM dbo.document_version where document_decision_id IS NULL
127 GO
128
129 CREATE PROCEDURE unprocessedSequencelistingProc(@afterDate DATETIME)
130 AS
131 BEGIN
132 SELECT * from unprocessedSequencelisting WHERE unprocessedSequencelisting.create_ts >= @afterDate
133 GO
134
135 DECLARE @afterDate DATETIME
136 SET @afterDate = '2019-01-01'
137 EXEC unprocessedSequencelistingProc @afterDate

```

The bottom pane shows the 'Results' tab with a single row of data:

document_version_id	content_id	bio_sequence_format_id	source_system_id	submission_id	submission_timestamp	document_code_id	patent_application_no	document_decision_id	major_version_no
3	cd12345b	3	2	s0091827125	2019-01-15 00:00:00.000	2	16000178	NULL	2

The status bar at the bottom indicates 'Query executed successfully.' and '1 rows'.

10. Project Summary:

This report comprises of the system design for the proposed system and starting from their functions, dependencies and relationships between the various entities and attributes. The analysis of the prototype is included in this report. The prototype has been completed successfully giving with robust impression that the proposed data model can solve USPTO SCORE business problem and it also can be example for other AIS to start thinking about moving to SQL server with proper data model design which will provide data independency and flexibility to make a query and free to design application based on data.

11.

Appendix:

DML Script:

```
/*
Author : Debasis Chatterjee
Course : IST659 M403
Term   : March 20, 2019
*/

--CREATE DATABASE [BIOSEQ]
--GO

IF EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME='biosequence_user')
BEGIN
    DROP TABLE biosequence_user
END
GO
IF EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME='stnd_document_code')
BEGIN
    DROP TABLE stnd_document_code
END
GO
IF EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME='stnd_bio_sequence_format')
BEGIN
    DROP TABLE stnd_bio_sequence_format
END
GO
IF EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME='stnd_document_decision')
BEGIN
    DROP TABLE stnd_document_decision
END
GO
IF EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME='stnd_source_system')
BEGIN
    DROP TABLE stnd_source_system
END
GO
IF EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME='stnd_export_system')
BEGIN
    DROP TABLE stnd_export_system
END
GO
IF EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME='document_version')
BEGIN
    DROP TABLE document_version
END
GO
```

```

IF EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME='document_subversion')
BEGIN
    DROP TABLE document_subversion
END
GO
IF EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME='document_export')
BEGIN
    DROP TABLE document_export
END
GO

-- Creating the BioSequence User Table
CREATE TABLE biosequence_user(
    -- Columns for the user table
    bio_user_id int identity,
    patron_id    varchar(50) not null,
    first_name   varchar(50) not null,
    last_name    varchar(50) not null,
    email_id     varchar(50),
    -- Constraints on the User Table
    CONSTRAINT pk_biosequence_user PRIMARY KEY (bio_user_id),
    CONSTRAINT u1_biosequence_user UNIQUE(patron_id),
    CONSTRAINT u2_biosequence_user UNIQUE(email_id)
)
GO

CREATE TABLE stnd_document_code(
    document_code_id    int identity,
    document_code        varchar(25)    NOT NULL,
    description_tx       varchar(255),
    begin_effective_dt   datetime        NOT NULL,
    end_effective_dt     datetime,
    create_ts            datetime        DEFAULT CURRENT_TIMESTAMP NOT NULL,
    create_user_id       int             NOT NULL,
    last_mod_ts          datetime        DEFAULT CURRENT_TIMESTAMP NOT NULL,
    last_mod_user_id     int             NOT NULL,
    is_locked            tinyint         DEFAULT 0 NOT NULL,
    CONSTRAINT pk_stnd_document_code PRIMARY KEY (document_code_id),
    CONSTRAINT u1_stnd_document_code UNIQUE (document_code),
    CONSTRAINT fk1_stnd_document_code FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),
    CONSTRAINT fk2_stnd_document_code FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)
)
GO

CREATE TABLE stnd_bio_sequence_format(
    bio_sequence_format_id    int identity,
    bio_sequence_format_nm    varchar(50)    NOT NULL,
    description_tx            varchar(255),
    begin_effective_dt        datetime        NOT NULL,

```

```

end_effective_dt      datetime,
create_ts             datetime      DEFAULT CURRENT_TIMESTAMP NOT NULL,
create_user_id        int           NOT NULL,
last_mod_ts           datetime      DEFAULT CURRENT_TIMESTAMP NOT NULL,
last_mod_user_id      int           NOT NULL,
is_locked              tinyint      DEFAULT 0 NOT NULL,
CONSTRAINT pk_stnd_bio_sequence_format PRIMARY KEY (bio_sequence_format_id),
CONSTRAINT u1_stnd_bio_sequence_format UNIQUE (bio_sequence_format_nm),
CONSTRAINT fk1_stnd_bio_sequence_format FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),
CONSTRAINT fk2_stnd_bio_sequence_format FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)
)
GO

```

```

CREATE TABLE stnd_document_decision(
  document_decision_id int identity,
  decision_cd          varchar(4)    NOT NULL,
  description_tx        varchar(255),
  begin_effective_dt    datetime     NOT NULL,
  end_effective_dt      datetime,
  create_ts             datetime      DEFAULT CURRENT_TIMESTAMP NOT NULL,
  create_user_id        int           NOT NULL,
  last_mod_ts           datetime      DEFAULT CURRENT_TIMESTAMP NOT NULL,
  last_mod_user_id      int           NOT NULL,
  is_locked              tinyint      DEFAULT 0 NOT NULL,
  CONSTRAINT pk_stnd_document_decision PRIMARY KEY (document_decision_id),
  CONSTRAINT u1_stnd_document_decision UNIQUE (decision_cd),
  CONSTRAINT fk1_stnd_document_decision FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),
  CONSTRAINT fk2_stnd_document_decision FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)
)
GO

```

```

CREATE TABLE stnd_source_system(
  source_system_id int identity,
  source_system_name varchar(10)    NOT NULL,
  description_tx        varchar(255),
  begin_effective_dt    datetime     NOT NULL,
  end_effective_dt      datetime,
  create_ts             datetime      DEFAULT CURRENT_TIMESTAMP NOT NULL,
  create_user_id        int           NOT NULL,
  last_mod_ts           datetime      DEFAULT CURRENT_TIMESTAMP NOT NULL,
  last_mod_user_id      int           NOT NULL,
  is_locked              tinyint      DEFAULT 0 NOT NULL,
  CONSTRAINT pk_stnd_source_system PRIMARY KEY (source_system_id),
  CONSTRAINT u1_stnd_source_system UNIQUE (source_system_name),
  CONSTRAINT fk1_stnd_source_system FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),

```

```

        CONSTRAINT fk2_stnd_source_system FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)
)
GO

```

```

CREATE TABLE stnd_export_system(
    exported_system_id      int identity,
    exported_system_nm      varchar(50) NOT NULL,
    description_tx          varchar(255),
    begin_effective_dt      datetime NOT NULL,
    end_effective_dt        datetime,
    create_ts              datetime DEFAULT CURRENT_TIMESTAMP NOT NULL,
    create_user_id          int NOT NULL,
    last_mod_ts            datetime DEFAULT CURRENT_TIMESTAMP NOT NULL,
    last_mod_user_id        int NOT NULL,
    is_locked              tinyint DEFAULT 0 NOT NULL,
    CONSTRAINT pk_stnd_export_system PRIMARY KEY (exported_system_id),
    CONSTRAINT u1_stnd_export_system UNIQUE (exported_system_nm),
    CONSTRAINT fk1_stnd_export_system FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),
    CONSTRAINT fk2_stnd_export_system FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)
)
GO

```

```

CREATE TABLE document_version(
    document_version_id    int identity,
    content_id             varchar(36) NOT NULL,
    bio_sequence_format_id int NOT NULL,
    source_system_id        int NOT NULL,
    submission_id          varchar(50),
    submission_timestamp    datetime NOT NULL,
    document_code_id        int NOT NULL,
    patent_application_no   varchar(17) NOT NULL,
    document_decision_id    int,
    major_version_no        INT DEFAULT -1 NOT NULL,
    sequence_quantity       int,
    create_user_id          int NOT NULL,
    create_ts              datetime DEFAULT CURRENT_TIMESTAMP NOT NULL,
    last_mod_ts            datetime DEFAULT CURRENT_TIMESTAMP NOT NULL,
    last_mod_user_id        int NOT NULL,
    is_locked              tinyint DEFAULT 0 NOT NULL,
    is_deleted             tinyint DEFAULT 0 NOT NULL,
    CONSTRAINT pk_document_version PRIMARY KEY (document_version_id),
    CONSTRAINT u1_document_version UNIQUE (content_id),
    CONSTRAINT fk1_document_version FOREIGN KEY (bio_sequence_format_id) References
stnd_bio_sequence_format(bio_sequence_format_id),
    CONSTRAINT fk2_document_version FOREIGN KEY (source_system_id) References
stnd_source_system(source_system_id),
    CONSTRAINT fk3_document_version FOREIGN KEY (document_code_id) References
stnd_document_code(document_code_id),

```

```

        CONSTRAINT fk4_document_version FOREIGN KEY (document_decision_id) References
stdnd_document_decision(document_decision_id),
        CONSTRAINT fk5_document_version FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),
        CONSTRAINT fk6_document_version FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)
)
GO

```

```

CREATE TABLE document_subversion(
    document_subversion_id      int identity,
    document_version_id         int                NOT NULL,
    minor_version_no            int                NOT NULL,
    create_ts                   DATETIME           DEFAULT CURRENT_TIMESTAMP NOT NULL,
    last_mod_ts                 DATETIME           DEFAULT CURRENT_TIMESTAMP NOT NULL,
    last_mod_user_id            int                NOT NULL,
    is_locked                   tinyint            DEFAULT 0 NOT NULL,
    is_deleted                   tinyint            DEFAULT 0 NOT NULL,
    CONSTRAINT pk_document_subversion PRIMARY KEY (document_subversion_id),
    CONSTRAINT fk1_document_subversion FOREIGN KEY (document_version_id) References
document_version(document_version_id),
    CONSTRAINT fk2_document_subversion FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)
)
GO

```

```

CREATE TABLE document_export(
    document_export_id          int identity,
    document_version_id         int                NOT NULL,
    document_subversion_id      int                NOT NULL,
    exported_system_id          int                NOT NULL,
    export_status               tinyint            NOT NULL,
    export_ts                   datetime           NOT NULL,
    create_ts                   datetime           DEFAULT CURRENT_TIMESTAMP NOT NULL,
    create_user_id              int                NOT NULL,
    last_mod_ts                 datetime           DEFAULT CURRENT_TIMESTAMP NOT NULL,
    last_mod_user_id            int                NOT NULL,
    is_locked                   tinyint            DEFAULT 0 NOT NULL,
    is_deleted                   tinyint            DEFAULT 0 NOT NULL,
    CONSTRAINT pk_document_export PRIMARY KEY (document_export_id),
    CONSTRAINT fk1_document_export FOREIGN KEY (document_version_id) References
document_version(document_version_id),
    CONSTRAINT fk2_document_export FOREIGN KEY (document_subversion_id) References
document_subversion(document_subversion_id),
    CONSTRAINT fk3_document_export FOREIGN KEY (exported_system_id) References
stdnd_export_system(exported_system_id),
    CONSTRAINT fk4_document_export FOREIGN KEY (create_user_id) References
biosequence_user(bio_user_id),
    CONSTRAINT fk5_document_export FOREIGN KEY (last_mod_user_id) References
biosequence_user(bio_user_id)
)

```


GO

DQL

Insertion of the data script:

```
/*
Author : Debasis Chatterjee
Course : IST659 M403
Term   : March 28, 2019
*/

--CREATE DATABASE [BIOSEQ]
--GO

-- Insert into biosequence_user
INSERT INTO biosequence_user
    (patron_id, first_name, last_name, email_id) VALUES
    ('312312', 'Debasis', 'Chatterjee', 'dchatter@syr.edu')
GO
INSERT INTO biosequence_user
    (patron_id, first_name, last_name, email_id) VALUES
    ('312313', 'Dale', 'Thompson', 'edthomps@syr.edu')
GO
INSERT INTO biosequence_user
    (patron_id, first_name, last_name, email_id) VALUES
    ('312314', 'Kimberly', 'Pendelberry', 'kpendelb@syr.edu')
GO
INSERT INTO biosequence_user
    (patron_id, first_name, last_name, email_id) VALUES
    ('312315', 'Abdullah', 'Mamdouh', 'amawaysh@syr.edu')
GO
INSERT INTO biosequence_user
    (patron_id, first_name, last_name, email_id) VALUES
    ('312316', 'James', 'Robertson', 'jrober12@syr.edu')
GO
-- Insert into stnd_document_code
INSERT INTO stnd_document_code(document_code, description_tx, begin_effective_dt,
create_user_id, last_mod_user_id) values
    ('1', '1', '1', '1', '1');
GO
INSERT INTO stnd_document_code(document_code, description_tx, begin_effective_dt,
create_user_id, last_mod_user_id) values
    ('1', '1', '1', '1', '1');
GO
```

```

-- Insert into stnd_bio_sequence_format
INSERT INTO stnd_bio_sequence_format (bio_sequence_format_nm, description_tx, begin_effective_dt,
create_user_id, last_mod_user_id) values
('ST23', 'Sequence Listing
23', '1970-01-01', '1', '1');
GO
INSERT INTO stnd_bio_sequence_format (bio_sequence_format_nm, description_tx, begin_effective_dt,
create_user_id, last_mod_user_id) values
('ST25', 'Sequence Listing
25', '1970-01-01', '1', '1');
GO
INSERT INTO stnd_bio_sequence_format (bio_sequence_format_nm, description_tx, begin_effective_dt,
create_user_id, last_mod_user_id) values
('ST26', 'Sequence Listing
26', '1970-01-01', '1', '1');
GO

-- Insert into stnd_document_decision
INSERT INTO stnd_document_decision (decision_cd, description_tx, begin_effective_dt,
create_user_id, last_mod_user_id) values
('CRFE', 'COMPUTER READABLE FORMAT EFFECTIVE', '1970-01-01',
'1', '1');
GO
INSERT INTO stnd_document_decision (decision_cd, description_tx, begin_effective_dt,
create_user_id, last_mod_user_id) values
('CRFD', 'COMPUTER READABLE FORMAT DEFERRED', '1970-01-01', '1',
'1');
GO

-- Insert into stnd_source_system
INSERT INTO stnd_source_system (source_system_name, description_tx, begin_effective_dt,
create_user_id, last_mod_user_id) VALUES
('EFSWEB', 'ELECTRONIC FILING SYSTEM', '1970-01-01',
'1', '1')
GO
INSERT INTO stnd_source_system (source_system_name, description_tx, begin_effective_dt,
create_user_id, last_mod_user_id) VALUES
('SLIC', 'SEQUENCELISTING CONTROL', '1970-01-01',
'1', '1')
GO
INSERT INTO stnd_source_system (source_system_name, description_tx, begin_effective_dt,
end_effective_dt, create_user_id, last_mod_user_id) VALUES
('POWER', 'PATENT POWER', '1970-01-01',
'2030-12-31', '1', '1')
GO
INSERT INTO stnd_source_system (source_system_name, description_tx, begin_effective_dt,
end_effective_dt, create_user_id, last_mod_user_id) VALUES
('IB', 'INTERNATIONAL BUREAU', '1970-01-01',
'2030-12-31', '1', '1')
GO
-- Insert into stnd_export_system

```

```

INSERT INTO stnd_export_system (exported_system_nm, description_tx, begin_effective_dt,
end_effective_dt, create_user_id, last_mod_user_id) VALUES
('IDC', 'INITIAL DATA CAPTURE', '1970-01-01',
'2030-12-31', '1', '1')
GO
INSERT INTO stnd_export_system (exported_system_nm, description_tx, begin_effective_dt,
end_effective_dt, create_user_id, last_mod_user_id) VALUES
('FDC', 'FINAL DATA CAPTURE', '1970-01-01',
'2030-12-31', '1', '1')
GO
INSERT INTO stnd_export_system (exported_system_nm, description_tx, begin_effective_dt,
end_effective_dt, create_user_id, last_mod_user_id) VALUES
('PGC', 'INTERNATIONAL BUREAU', '1970-01-01',
'2030-12-31', '1', '1')
GO
INSERT INTO stnd_export_system (exported_system_nm, description_tx, begin_effective_dt,
end_effective_dt, create_user_id, last_mod_user_id) VALUES
('PPD', 'PROJECTED PUBLICATION DATE', '1970-01-01',
'2030-12-31', '1', '1')
GO
INSERT INTO stnd_export_system (exported_system_nm, description_tx, begin_effective_dt,
end_effective_dt, create_user_id, last_mod_user_id) VALUES
('ABBSS', 'AUTOMATED BIOTECHNOLOGY SEQUENCE SEARCH SYSTEM',
'1970-01-01', '2030-12-31', '1', '1')
GO
-- Insert into document_version-----
-----
INSERT INTO document_version (content_id, bio_sequence_format_id, source_system_id, submission_id,
submission_timestamp, document_code_id, patent_application_no, document_decision_id, major_version_no,
sequence_qtquantity, create_user_id, create_ts, last_mod_ts, last_mod_user_id) VALUES
('cd123456', '1', '1', '1', 's0091827123', '2018-12-
28', '1', '14289012', '1', '1', '102',
'2', '2019-01-31', '2019-01-31', '2')
GO
--need to correct the document_code_id to 1
INSERT INTO document_version (content_id, bio_sequence_format_id, source_system_id, submission_id,
submission_timestamp, document_code_id, patent_application_no, document_decision_id, major_version_no,
sequence_qtquantity, create_user_id, create_ts, last_mod_ts, last_mod_user_id) VALUES
('cd12345a', '2', '2', '2', 's0091827124', '2019-01-
15', '2', '16000178', '2', '1', '32189',
'5', '2019-03-21', '2019-03-21', '5')
GO
INSERT INTO document_version (content_id, bio_sequence_format_id, source_system_id, submission_id,
submission_timestamp, document_code_id, patent_application_no, document_decision_id, major_version_no,
sequence_qtquantity, create_user_id, create_ts, last_mod_ts, last_mod_user_id) VALUES
('cd12345b', '3', '2', '2', 's0091827125', '2019-01-
15', '2', '16000178', '2', '7675655', '5',
'2019-03-25', '2019-03-25', '5')
GO

```

```

INSERT INTO document_version (content_id, bio_sequence_format_id, source_system_id, submission_id,
submission_timestamp, document_code_id, patent_application_no, document_decision_id, major_version_no,
sequence_quantity, create_user_id, create_ts, last_mod_ts, last_mod_user_id) VALUES
('cd12345c', '3', '3', 's0091827127', '2017-05-
23', '2', '62064677', '1', '1', '18',
'1', '2019-02-28', '2019-02-28', '1'
)
GO
-- Insert into document_subversion-----
-----
INSERT INTO document_subversion (document_version_id, minor_version_no, create_ts, last_mod_ts,
last_mod_user_id) VALUES
('1', '0', '2019-01-31', '2019-01-31', '2')
GO
INSERT INTO document_subversion (document_version_id, minor_version_no, create_ts, last_mod_ts,
last_mod_user_id) VALUES
('1', '1', '2019-02-01', '2019-02-01', '3')
GO
INSERT INTO document_subversion (document_version_id, minor_version_no, create_ts, last_mod_ts,
last_mod_user_id) VALUES
('1', '2', '2019-02-01', '2019-02-01', '3')
GO
INSERT INTO document_subversion (document_version_id, minor_version_no, create_ts, last_mod_ts,
last_mod_user_id) VALUES
('2', '0', '2019-03-21', '2019-03-21', '5')
GO
INSERT INTO document_subversion (document_version_id, minor_version_no, create_ts, last_mod_ts,
last_mod_user_id) VALUES
('2', '1', '2019-03-22', '2019-03-21', '5')
GO
INSERT INTO document_subversion (document_version_id, minor_version_no, create_ts, last_mod_ts,
last_mod_user_id) VALUES
('3', '0', '2019-03-25', '2019-03-25', '4')
GO
INSERT INTO document_subversion (document_version_id, minor_version_no, create_ts, last_mod_ts,
last_mod_user_id) VALUES
('4', '0', '2019-02-28', '2019-02-28', '1')
GO
INSERT INTO document_subversion (document_version_id, minor_version_no, create_ts, last_mod_ts,
last_mod_user_id) VALUES
('4', '1', '2019-02-28', '2019-02-28', '2')
GO
-- Insert into document_export-----
-----
INSERT INTO document_export (document_version_id, document_subversion_id, exported_system_id, export_status,
export_ts, create_ts, create_user_id, last_mod_ts, last_mod_user_id) VALUES
('1', '11', '5', '1',
'2019-02-02', '2019-02-02', '3', '2019-02-02', '3'
)
GO

```

```

INSERT INTO document_export (document_version_id, document_subversion_id, exported_system_id, export_status,
export_ts,      create_ts,      create_user_id, last_mod_ts, last_mod_user_id) VALUES
('1' ,      '11',      '1',      '1',
'2019-03-02', '2019-03-02', '3',      '2019-03-02', '3'
)
GO
INSERT INTO document_export (document_version_id, document_subversion_id, exported_system_id, export_status,
export_ts,      create_ts,      create_user_id, last_mod_ts, last_mod_user_id) VALUES
('1' ,      '11',      '2',      '1',
'2019-03-03', '2019-03-03', '3',      '2019-03-03', '3'
)
GO
INSERT INTO document_export (document_version_id, document_subversion_id, exported_system_id, export_status,
export_ts,      create_ts,      create_user_id, last_mod_ts, last_mod_user_id) VALUES
('4' ,      '16',      '5',      '1',
'2019-03-01', '2019-03-01', '3',      '2019-03-01', '3'
)
GO
INSERT INTO document_export (document_version_id, document_subversion_id, exported_system_id, export_status,
export_ts,      create_ts,      create_user_id, last_mod_ts, last_mod_user_id) VALUES
('4' ,      '16',      '2',      '1',
'2019-03-25', '2019-03-25', '3',      '2019-03-25', '3'
)
GO
INSERT INTO document_export (document_version_id, document_subversion_id, exported_system_id, export_status,
export_ts,      create_ts,      create_user_id, last_mod_ts, last_mod_user_id) VALUES
('4' ,      '16',      '3',      '1',
'2019-03-26', '2019-03-26', '3',      '2019-03-26', '3'
)
GO
INSERT INTO document_export (document_version_id, document_subversion_id, exported_system_id, export_status,
export_ts,      create_ts,      create_user_id, last_mod_ts, last_mod_user_id) VALUES
('4' ,      '16',      '4',      '1',
'2019-03-28', '2019-03-28', '3',      '2019-03-28', '3'
)
GO

```

Select queries to find data:

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [bio_user_id]
      ,[patron_id]
      ,[first_name]
      ,[last_name]
      ,[email_id]
FROM [BIOSEQ].[dbo].[biosequence_user]

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [document_code_id]
      ,[document_code]
      ,[description_tx]
      ,[begin_effective_dt]
      ,[end_effective_dt]
      ,[create_ts]

```

```

        ,[create_user_id]
        ,[last_mod_ts]
        ,[last_mod_user_id]
        ,[is_locked]
FROM [BIOSEQ].[dbo].[stnd_document_code]

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [bio_sequence_format_id]
        ,[bio_sequence_format_nm]
        ,[description_tx]
        ,[begin_effective_dt]
        ,[end_effective_dt]
        ,[create_ts]
        ,[create_user_id]
        ,[last_mod_ts]
        ,[last_mod_user_id]
        ,[is_locked]
FROM [BIOSEQ].[dbo].[stnd_bio_sequence_format]

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [document_decision_id]
        ,[decision_cd]
        ,[description_tx]
        ,[begin_effective_dt]
        ,[end_effective_dt]
        ,[create_ts]
        ,[create_user_id]
        ,[last_mod_ts]
        ,[last_mod_user_id]
        ,[is_locked]
FROM [BIOSEQ].[dbo].[stnd_document_decision]

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [source_system_id]
        ,[source_system_name]
        ,[description_tx]
        ,[begin_effective_dt]
        ,[end_effective_dt]
        ,[create_ts]
        ,[create_user_id]
        ,[last_mod_ts]
        ,[last_mod_user_id]
        ,[is_locked]
FROM [BIOSEQ].[dbo].[stnd_source_system]

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [exported_system_id]
        ,[exported_system_nm]
        ,[description_tx]
        ,[begin_effective_dt]
        ,[end_effective_dt]

```

```

        ,[create_ts]
        ,[create_user_id]
        ,[last_mod_ts]
        ,[last_mod_user_id]
        ,[is_locked]
FROM [BIOSEQ].[dbo].[stnd_export_system]

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [document_version_id]
        ,[content_id]
        ,[bio_sequence_format_id]
        ,[source_system_id]
        ,[submission_id]
        ,[submission_timestamp]
        ,[document_code_id]
        ,[patent_application_no]
        ,[document_decision_id]
        ,[major_version_no]
        ,[sequence_quantity]
        ,[create_user_id]
        ,[create_ts]
        ,[last_mod_ts]
        ,[last_mod_user_id]
        ,[is_locked]
        ,[is_deleted]
FROM [BIOSEQ].[dbo].[document_version]

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [document_subversion_id]
        ,[document_version_id]
        ,[minor_version_no]
        ,[create_ts]
        ,[last_mod_ts]
        ,[last_mod_user_id]
        ,[is_locked]
        ,[is_deleted]
FROM [BIOSEQ].[dbo].[document_subversion]

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [document_export_id]
        ,[document_version_id]
        ,[document_subversion_id]
        ,[exported_system_id]
        ,[export_status]
        ,[export_ts]
        ,[create_ts]
        ,[create_user_id]
        ,[last_mod_ts]
        ,[last_mod_user_id]
        ,[is_locked]
        ,[is_deleted]

```

```
FROM [BIOSEQ].[dbo].[document_export]
```

DB queries to support Business questions behind the forms/reports –

--Complete overview report of Bio-Sequence Exports

```
CREATE VIEW biosequence_exports_projection AS
SELECT  dbo.document_version.content_id,
        dbo.document_version.submission_id,
        dbo.document_version.patent_application_no,
        dbo.document_version.major_version_no,
        dbo.document_version.sequence_quantity,
        dbo.document_subversion.minor_version_no,
        dbo.document_subversion.last_mod_ts,
        dbo.document_export.export_status,
        dbo.document_export.export_ts,
        dbo.stnd_document_decision.decision_cd,
        dbo.stnd_bio_sequence_format.bio_sequence_format_nm,
        dbo.stnd_export_system.exported_system_nm,
        dbo.stnd_source_system.source_system_name,
        dbo.biosequence_user.first_name,
        dbo.biosequence_user.last_name
FROM    dbo.biosequence_user
        INNER JOIN (dbo.stnd_source_system
                    INNER JOIN (dbo.stnd_export_system
                                INNER JOIN (dbo.stnd_bio_sequence_format
                                            INNER JOIN (dbo.stnd_document_decision
                                                        INNER JOIN
(dbo.document_version
        INNER JOIN (dbo.document_subversion
                    INNER JOIN dbo.document_export
                                ON dbo.document_subversion.document_subversion_id =
dbo.document_export.document_subversion_id)
                    ON (dbo.document_version.document_version_id = dbo.document_export.document_version_id)
                    AND (dbo.document_version.document_version_id = dbo.document_subversion.document_version_id))
                                ON
dbo.stnd_document_decision.document_decision_id = dbo.document_version.document_decision_id)
                                ON
dbo.stnd_bio_sequence_format.bio_sequence_format_id = dbo.document_version.bio_sequence_format_id)
                                ON dbo.stnd_export_system.exported_system_id =
dbo.document_export.exported_system_id)
                                ON dbo.stnd_source_system.source_system_id =
dbo.document_version.source_system_id)
                                ON dbo.biosequence_user.bio_user_id = dbo.document_export.create_user_id;
```


GO

```
SELECT * FROM biosequence_exports_projection
```

---How many documents are certified or rejected in last one month

```
SELECT dbo.document_version.major_version_no, dbo.stnd_document_decision.decision_cd,  
MAX(dbo.document_subversion.minor_version_no)  
FROM (dbo.stnd_document_decision  
INNER JOIN dbo.document_version ON dbo.stnd_document_decision.document_decision_id =  
dbo.document_version.document_decision_id)  
INNER JOIN dbo.document_subversion ON dbo.document_version.document_version_id =  
dbo.document_subversion.document_version_id  
GROUP BY dbo.document_version.major_version_no, dbo.stnd_document_decision.decision_cd;
```

---How many documents are certified or rejected in last one month By application number

```
CREATE VIEW count_process_seq_by_month AS  
SELECT dbo.document_version.patent_application_no, dbo.document_version.major_version_no,  
dbo.stnd_document_decision.decision_cd, MAX(dbo.document_subversion.minor_version_no) as  
processed_sequencelisting_no  
FROM (dbo.stnd_document_decision  
INNER JOIN dbo.document_version ON dbo.stnd_document_decision.document_decision_id =  
dbo.document_version.document_decision_id)  
INNER JOIN dbo.document_subversion ON dbo.document_version.document_version_id =  
dbo.document_subversion.document_version_id  
WHERE dbo.document_subversion.last_mod_ts >= DATEADD(mm, DATEDIFF(mm, 0, GETDATE())-1, 0) AND  
dbo.document_subversion.last_mod_ts >= DATEADD(mm, DATEDIFF(mm, 0, GETDATE()), 0)  
GROUP BY dbo.document_version.patent_application_no, dbo.document_version.major_version_no,  
dbo.stnd_document_decision.decision_cd;  
GO
```

```
CREATE FUNCTION dbo.count_certify_Reject_by_Last_month()
```

```
RETURNS int AS
```

```
BEGIN
```

```
DECLARE @returnValue int
```

```
SELECT @returnValue = count_process_seq_by_month.processed_sequencelisting_no FROM  
count_process_seq_by_month
```

```
RETURN @returnvalue
```

```
END
```

Go

```
SELECT
```

```
dbo.count_certify_Reject_by_Last_month() as total_processed_sequence
```

```
FROM count_process_seq_by_month
```

Go

-- How many ST26 was loaded into system in last one year by projected year

```
CREATE PROCEDURE loadCountST26(@seqFormat VARCHAR(10))
```

```
AS
```

```
BEGIN
```

```
SELECT COUNT(dbo.document_version.document_version_id) as total_count_ST26,
```

```

        dbo.stnd_bio_sequence_format.bio_sequence_format_nm, YEAR(dbo.document_version.create_ts) as
projected_year
FROM dbo.stnd_bio_sequence_format
INNER JOIN dbo.document_version ON dbo.stnd_bio_sequence_format.bio_sequence_format_id =
dbo.document_version.bio_sequence_format_id
WHERE dbo.stnd_bio_sequence_format.bio_sequence_format_nm=@seqFormat
and dbo.document_version.create_ts > DATEADD(YEAR,-1, GETDATE())
GROUP BY dbo.stnd_bio_sequence_format.bio_sequence_format_nm, YEAR(dbo.document_version.create_ts);
END
Go
DECLARE @seqFormat VARCHAR(10)
SET @seqFormat = 'ST26'
EXEC loadCountST26 @seqFormat

-- How many certified sequencelisting have been exported to ABSS after march 1st 2019.
CREATE FUNCTION dbo.exported_seq_from_date(@fromDate DATETIME, @exportedSystem VARCHAR(10))
RETURNS int AS
BEGIN
    DECLARE @returnValue int
    SELECT @returnValue = COUNT(*) FROM biosequence_exports_projection
    WHERE dbo.biosequence_exports_projection.exported_system_nm = @exportedSystem and
dbo.biosequence_exports_projection.export_ts >= @fromDate
    RETURN @returnValue
END--Complete overview report of Bio-Sequence Exports

CREATE VIEW biosequence_exports_projection AS
SELECT  dbo.document_version.content_id,
        dbo.document_version.submission_id,
        dbo.document_version.patent_application_no,
        dbo.document_version.major_version_no,
        dbo.document_version.sequence_quantity,
        dbo.document_subversion.minor_version_no,
        dbo.document_subversion.last_mod_ts,
        dbo.document_export.export_status,
        dbo.document_export.export_ts,
        dbo.stnd_document_decision.decision_cd,
        dbo.stnd_bio_sequence_format.bio_sequence_format_nm,
        dbo.stnd_export_system.exported_system_nm,
        dbo.stnd_source_system.source_system_name,
        dbo.biosequence_user.first_name,
        dbo.biosequence_user.last_name
FROM    dbo.biosequence_user
INNER JOIN (dbo.stnd_source_system
            INNER JOIN (dbo.stnd_export_system
                        INNER JOIN (dbo.stnd_bio_sequence_format
                                    INNER JOIN (dbo.stnd_document_decision
                                                INNER JOIN
(dbo.document_version

        INNER JOIN (dbo.document_subversion

```

```

INNER JOIN dbo.document_export

ON dbo.document_subversion.document_subversion_id =
dbo.document_export.document_subversion_id)

ON (dbo.document_version.document_version_id = dbo.document_export.document_version_id)

AND (dbo.document_version.document_version_id = dbo.document_subversion.document_version_id))
ON
dbo.stnd_document_decision.document_decision_id = dbo.document_version.document_decision_id)
ON
dbo.stnd_bio_sequence_format.bio_sequence_format_id = dbo.document_version.bio_sequence_format_id)
ON
ON dbo.stnd_export_system.exported_system_id =
dbo.document_export.exported_system_id)
ON
ON dbo.stnd_source_system.source_system_id =
dbo.document_version.source_system_id)
ON
ON dbo.biosequence_user.bio_user_id = dbo.document_export.create_user_id;

GO

SELECT * FROM biosequence_exports_projection

---How many documents are certified or rejected in last one month
SELECT dbo.document_version.major_version_no, dbo.stnd_document_decision.decision_cd,
MAX(dbo.document_subversion.minor_version_no)
FROM (dbo.stnd_document_decision
INNER JOIN dbo.document_version ON dbo.stnd_document_decision.document_decision_id =
dbo.document_version.document_decision_id)
INNER JOIN dbo.document_subversion ON dbo.document_version.document_version_id =
dbo.document_subversion.document_version_id
GROUP BY dbo.document_version.major_version_no, dbo.stnd_document_decision.decision_cd;

---How many documents are certified or rejected in last one month By application number
CREATE VIEW count_process_seq_by_month AS
SELECT dbo.document_version.patent_application_no, dbo.document_version.major_version_no,
dbo.stnd_document_decision.decision_cd, MAX(dbo.document_subversion.minor_version_no) as
processed_sequencelisting_no
FROM (dbo.stnd_document_decision
INNER JOIN dbo.document_version ON dbo.stnd_document_decision.document_decision_id =
dbo.document_version.document_decision_id)
INNER JOIN dbo.document_subversion ON dbo.document_version.document_version_id =
dbo.document_subversion.document_version_id
WHERE dbo.document_subversion.last_mod_ts >= DATEADD(mm,DATEDIFF(mm,0,GETDATE())-1,0) AND
dbo.document_subversion.last_mod_ts >= DATEADD(mm,DATEDIFF(mm,0,GETDATE()),0)
GROUP BY dbo.document_version.patent_application_no, dbo.document_version.major_version_no,
dbo.stnd_document_decision.decision_cd;
GO

CREATE FUNCTION dbo.count_certify_Reject_by_Last_month()
RETURNS int AS

```

```

BEGIN
    DECLARE @returnValue int
    SELECT @returnValue = count_process_seq_by_month.processed_sequencelisting_no FROM
count_process_seq_by_month
    RETURN @returnvalue
END
Go

SELECT
dbo.count_certify_Reject_by_Last_month() as total_processed_sequence
FROM count_process_seq_by_month
Go

-- How many ST26 was loaded into system in last one year by projected year
CREATE PROCEDURE loadCountST26(@seqFormat VARCHAR(10))
AS
BEGIN
SELECT COUNT(dbo.document_version.document_version_id) as total_count_ST26,
        dbo.stnd_bio_sequence_format.bio_sequence_format_nm, YEAR(dbo.document_version.create_ts) as
projected_year
FROM dbo.stnd_bio_sequence_format
INNER JOIN dbo.document_version ON dbo.stnd_bio_sequence_format.bio_sequence_format_id =
        dbo.document_version.bio_sequence_format_id
WHERE dbo.stnd_bio_sequence_format.bio_sequence_format_nm=@seqFormat
and dbo.document_version.create_ts > DATEADD(YEAR,-1, GETDATE())
GROUP BY dbo.stnd_bio_sequence_format.bio_sequence_format_nm, YEAR(dbo.document_version.create_ts);
END
Go
DECLARE @seqFormat VARCHAR(10)
SET @seqFormat = 'ST26'
EXEC loadCountST26 @seqFormat

-- How many certified sequencelisting have been exported to ABSS after march 1st 2019.
CREATE FUNCTION dbo.exported_seq_from_date(@fromDate DATETIME, @exportedSystem VARCHAR(10))
RETURNS int AS
BEGIN
    DECLARE @returnValue int
    SELECT @returnValue = COUNT(*) FROM biosequence_exports_projection
    WHERE dbo.biosequence_exports_projection.exported_system_nm = @exportedSystem and
        dbo.biosequence_exports_projection.export_ts >= @fromDate
    RETURN @returnvalue
END
Go
DECLARE @fromDate DATETIME
SET @fromDate = '2019-03-01'
DECLARE @exportedSystem VARCHAR(10)
SET @exportedSystem = 'ABBSS'
SELECT
        dbo.exported_seq_from_date(@fromDate, @exportedSystem) as total_abbss_exported_sequence
FROM count_process_seq_by_month
Go

```

--How many sequencelisting header have been edited in last one year

```
CREATE PROCEDURE editHeaderSequenceListingCount(@minorVer int)
AS
BEGIN
SELECT COUNT(dbo.document_subversion.document_subversion_id) as total_no_edited_sequencelisting
FROM dbo.document_subversion
WHERE dbo.document_subversion.minor_version_no = @minorVer
and dbo.document_subversion.create_ts > DATEADD(WEEK,-8, GETDATE())
END
Go
DECLARE @minorVer int
SET @minorVer = '2'
EXEC editHeaderSequenceListingCount @minorVer
```

-- Provide the list of the sequencelisting or total count of sequencelisting are in the system have not been reviewed yet where sequencelisting is loaded after Jan 1, 2019

```
CREATE VIEW unprocessedSequenceListing AS
SELECT * FROM dbo.document_version where document_decision_id IS NULL
Go
```

```
CREATE PROCEDURE unprocessedSequenceListingProc(@afterDate DATETIME)
AS
BEGIN
SELECT * from unprocessedSequenceListing WHERE unprocessedSequenceListing.create_ts >= @afterDate
END
Go
DECLARE @afterDate DATETIME
SET @afterDate = '2019-01-01'
EXEC unprocessedSequenceListingProc @afterDate
```

```
Go
DECLARE @fromDate DATETIME
SET @fromDate = '2019-03-01'
DECLARE @exportedSystem VARCHAR(10)
SET @exportedSystem = 'ABBSS'
SELECT
dbo.exported_seq_from_date(@fromDate, @exportedSystem) as total_abbss_exported_sequence
FROM count_process_seq_by_month
Go
```

--How many sequencelisting header have been edited in last one year

```
CREATE PROCEDURE editHeaderSequenceListingCount(@minorVer int)
AS
BEGIN
SELECT COUNT(dbo.document_subversion.document_subversion_id) as total_no_edited_sequencelisting
FROM dbo.document_subversion
WHERE dbo.document_subversion.minor_version_no = @minorVer
```

```

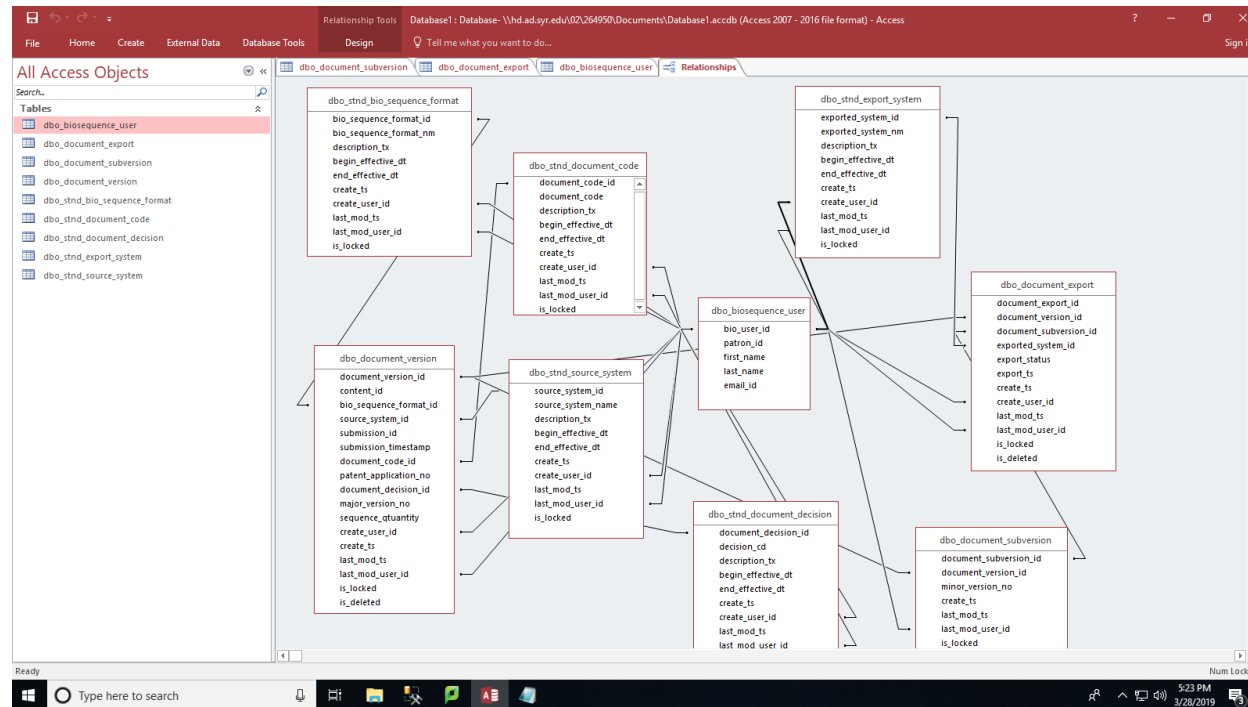
and dbo.document_subversion.create_ts > DATEADD(WEEK,-8, GETDATE())
END
Go
DECLARE @minorVer int
SET @minorVer = '2'
EXEC editHeaderSequenceListingCount @minorVer

-- Provide the list of the sequencelisting or total count of sequencelisting are in the system have not been
reviewed yet where sequencelisting is loaded after Jan 1, 2019
CREATE VIEW unprocessedSequenceListing AS
SELECT * FROM dbo.document_version where document_decision_id IS NULL
Go

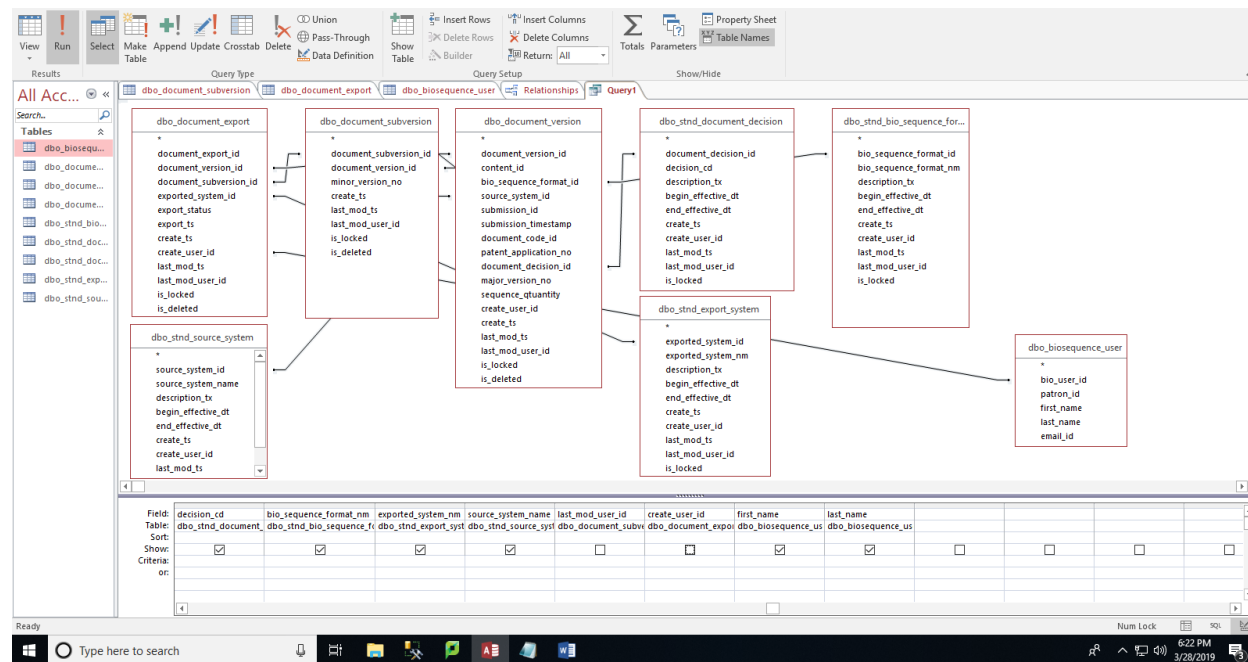
CREATE PROCEDURE unprocessedSequenceListingProc(@afterDate DATETIME)
AS
BEGIN
SELECT * from unprocessedSequenceListing WHERE unprocessedSequenceListing.create_ts >= @afterDate
END
Go
DECLARE @afterDate DATETIME
SET @afterDate = '2019-01-01'
EXEC unprocessedSequenceListingProc @afterDate

```

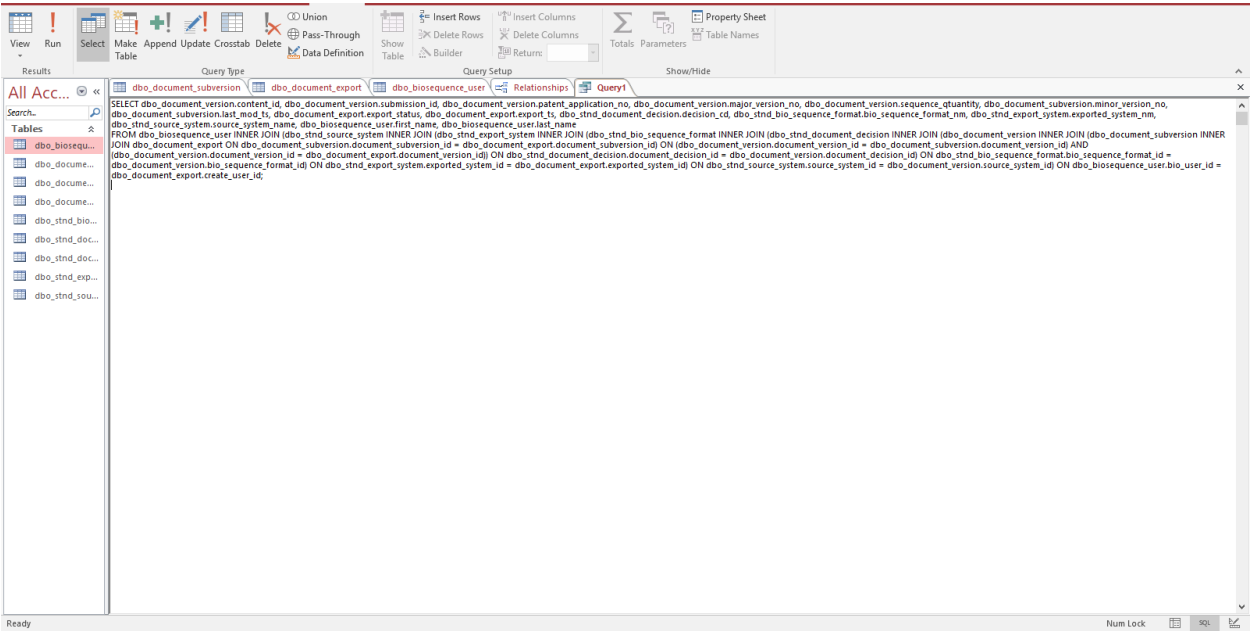
Relationship made in MS Access for Query design and Form



Query Design



Sql Query View



12.

Glossary	
USPTO	UNITED STATES AND PATENT TRADEMARK OFFICE
WIPO	World Intellectual Property Organization
ST.23	Standard 23
ST.25	Standard 25
ST.26	Standard 26
ERD	Entity relationship diagram
DBMS	Database management system
EFS	Electronic filing system
OIPE	Office of Initial Patent Examination
STIC	Scientific and Technical Information Center
ABSS	Automated Biotechnology Sequence Search
CRFE	Computer readable format effective
CRFD	Computer readable format differed