

Z3950 Gateway サンプルアプリケーション概説書

Version 2.2

2001年08月31日
コンパックコンピュータ株式会社
ソリューション企画推進本部/Cache'推進部

変更履歴

日付	ページ	内容	担当
2000/10/27		V2.0 初版	柴田
2000/11/30		V2.1 改訂	柴田
2000/12/11		^ZGSYS 項目追加	柴田
2001/08/28		V2.2 改訂 主要追記 2.4.9 文字コード変換追加 2.5.8 DB 表示単位登録・修正・削除 3.9 ZGLIBC 3.18 ZGMAN5 5.7-5.10 EUC2EACC,EACC2EUC,MARC8,MGWAPP 6.1.11,6.1.16 DB 表示単位関連画面	柴田

目次

1	概要	1
1.1	ソフトウェア環境	1
1.2	アプリケーション動作環境	1
1.3	サンプルアプリケーションの構成	2
1.4	WEBLINK 設定	2
1.4.1	サーバ設定	2
1.4.2	WEB からのアプリケーション指定変数	2
1.5	処理概要	3
1.5.1	初期化の処理	3
1.5.2	要求の処理	3
1.5.3	応答獲得の処理	4
2	機能	5
2.1	Gateway 機能	5
2.2	Checker 機能	6
2.3	Gateway・Checker プロセスの起動と停止	6
2.3.1	開始	6
2.3.2	停止	6
2.4	Client プロセスとのインターフェース	7
2.4.1	初期化	7
2.4.2	SEARCH 要求	7
2.4.3	SEARCH 応答	8
2.4.4	PRESENT 要求	8
2.4.5	PRESENT 応答	9
2.4.6	CLOSE 要求	9
2.4.7	データ獲得/更新関連	10
2.4.8	その他	12
2.4.9	文字コード変換	14
2.4.10	フォーマット変換	15
2.4.11	プログラムの流れ	17
2.5	システム管理	18
2.5.1	DB 登録・修正・削除機能	18
2.5.2	ゲットウェイ設定	19
2.5.3	開始・停止(WEB)	20
2.5.4	クライアント接続情報監視	20
2.5.5	検索キー登録・修正	20
2.5.6	DB 毎 BIB-1 属性登録	20
2.5.7	DB 毎 一覧表示/詳細表示(TEXT)項目登録	20
2.5.8	DB 表示単位登録・修正・削除	21
2.6	動作環境初期化	21
3	プログラム・ラベル一覧	22
3.1	ZGINI,ZGINI1	22
3.2	ZGSS	22
3.3	ZGCHKR	22
3.4	ZGORGD	23
3.5	ZGSND	23
3.6	ZGAPI	24
3.7	ZGAPI1	24
3.8	ZGLIBN	25
3.9	ZGLIBC	25
3.10	ZGCCNV	26
3.11	ZGCOM(ZGWCOM)	26
3.12	ZGCCOM(CATCOM)	26
3.13	ZGMAN	27
3.14	ZGMAN1	27
3.15	ZGMAN2	27
3.16	ZGMAN3	27

3.17	ZGMAN4.....	28
3.18	ZGMAN5.....	28
4	プログラム処理.....	29
4.1	Checker プロセス.....	29
4.2	開始.....	30
4.3	停止.....	31
4.4	Gateway プロセス.....	32
4.5	Gateway I/F.....	39
4.5.1	初期処理/ハンドルの確保(INIT^ZGSND).....	39
4.5.2	SEARCH 要求(SRCQ^ZGSND).....	39
4.5.3	SEARCH 応答(SRCR^ZGSND).....	41
4.5.4	PRESENT 要求(PRSQ^ZGSND).....	42
4.5.5	PRESENT 応答(PRSR^ZGSND).....	43
4.5.6	CLOSE 要求(CLOSE^ZGSND).....	45
4.6	Gateway 検索システム.....	47
4.6.1	メイン.....	47
4.6.2	ターゲット選択・検索式入力[Search Menu].....	48
4.6.3	検索処理の実行・検索結果一覧出力[Result list].....	48
4.6.4	データ一覧出力[Data list].....	49
4.6.5	データ詳細出力[Detail Data].....	52
4.6.6	検索履歴[Search History].....	57
4.6.7	終了[Close].....	58
5	グローバル仕様.....	59
5.1	ZGSYS.....	59
5.2	ZGJOB.....	63
5.3	ZGQRY.....	65
5.4	ZGRD.....	67
5.5	ZGLOG.....	69
5.6	ZGDBG.....	70
5.7	EUC2EACC.....	71
5.8	EACC2EUC.....	71
5.9	MARC8.....	71
5.10	MGWAPP.....	72
6	画面一覧.....	73
6.1	システム管理.....	73
6.1.1	管理メニュー.....	74
6.1.2	DB情報新規登録.....	75
6.1.3	DB情報修正・削除登録.....	77
6.1.4	ゲートウェイ設定変更.....	79
6.1.5	ゲートウェイ開始/停止/監視.....	80
6.1.6	クライアント接続情報.....	82
6.1.7	検索キー登録・修正.....	84
6.1.8	DB毎 BIB-1 属性登録.....	85
6.1.9	DB毎 一覧表示項目登録.....	86
6.1.10	DB毎 詳細表示 (TEXT) 項目登録.....	86
6.1.11	DB表示単位新規登録.....	87
6.1.12	DB表示単位登録確認.....	88
6.1.13	DB表示単位選択.....	88
6.1.14	DB表示単位修正・削除登録.....	89
6.1.15	その他.....	90
6.2	検索システム.....	91
6.2.1	DB選択・検索条件入力.....	92
6.2.2	検索結果一覧.....	93
6.2.3	データ一覧.....	94
6.2.4	詳細データ表示.....	96
6.2.5	検索履歴.....	99
6.2.6	その他.....	99

図表目次

図 1-1 初期化の処理	3
図 1-2 要求の処理	3
図 1-3 応答獲得の処理	4
図 2-1 プログラムの流れ	17
図 6-1 管理メニュー	74
図 6-2 DB新規登録	75
図 6-3 DB登録確認	76
図 6-4 DB修正・削除登録[DB選択]	77
図 6-5 DB修正・削除登録	78
図 6-6 ゲートウェイ設定変更	79
図 6-7 ゲートウェイ開始/停止（停止状態）	80
図 6-8 ゲートウェイ開始/停止（開始済）	81
図 6-9 クライアント接続情報	82
図 6-10 検索情報	83
図 6-11 検索キー登録・修正	84
図 6-12 DB毎 BIB-1 属性登録(Z39.50)	85
図 6-13 DB毎 BIB-1 属性登録(NII-CATP)	85
図 6-14 DB毎 一覧表示項目登録	86
図 6-15 DB毎 詳細表示（TEXT）項目登録	86
図 6-16 DB表示単位新規登録	87
図 6-17 DB表示単位登録確認	88
図 6-18 DB表示単位選択	88
図 6-19 DB表示単位修正・削除登録	89
図 6-20 エラーメッセージ出力	90
図 6-21 DB選択・検索条件入力	92
図 6-22 検索結果一覧（検索中）	93
図 6-23 検索結果一覧（検索完了）	93
図 6-24 データ一覧 MARC21(USMARC)	94
図 6-25 データ一覧 NII-CAT	94
図 6-26 レコード取得中	95
図 6-27 詳細データ表示 MARC21(USMARC) (TEXT)	96
図 6-28 詳細データ表示 MARC21(USMARC) (MARC)	96
図 6-29 詳細データ表示 SUTRS (MARC)	97
図 6-30 詳細データ表示 NII-CATP (TEXT)	98
図 6-31 詳細データ表示 NII-CATP (MARC)	98
図 6-32 検索履歴	99
図 6-33 エラーメッセージ出力	99

1 概要

本サンプルアプリケーションは、DSM 上で動作し、Z39.50 及び CAT-P クライアントソフトウェアを統合した WWW ゲートウェイ機能を提供します。

1.1 ソフトウェア環境

本サンプルアプリケーションは以下のソフトウェア環境で動作します。

- OpenVMS Ver 7.1 以上
- TCP/IP Service for OpenVMS Alpha V4.1 以上
- DSM for OpenVMS Alpha V6.4 以上
- Z39.50 クライアント API
- CAT-P クライアント API
- Cache' WebLink V4.2 以上
- 文字コード変換 KCV 関数¹

1.2 アプリケーション動作環境

- 動作 DSM 環境
この WWW ゲートウェイアプリケーションは、以下の3つのDSMボリュームセットから構成される1つのDSM環境で動作します。²

DSM UCI,VOL 名	string length	用途 (作成グローバル)
XXX,XXX	512	Z39.50 GATEWAY DSM ルーチン、WEBLINK アプリケーション(^MGWAPP) コンフィグ情報(^ZGSYS)、デバッグ情報(^ZGDBG)
XXX,XXX	512	オンラインログ(^ZGLOG)、要求ファイル(^ZGQRY)、Gateway プロセス情報(^ZGJOB)
XXX,XXX	512	結果ファイル(^ZGRD)

- プロセスと必要な資源
開始により、2種類のプロセス(Gateway プロセスと Checker プロセス)が、DSM のジョブとして作成されます。各ジョブは、ソースバッファとシンボルテーブルサイズについて以下のオプション付きで起動されます。

```
/SOURCE_BUFFER_SIZE=40000  
/SYMBOL_TABLE_SIZE=6000000
```

```
Gateway プロセス名   ZG_DBn_i           (n=ターゲットDB番号,i=DB番号内の通番)  
Checker プロセス名  ZG_CHECKER
```

- 必要な DSM ライセンス数
Gateway プロセスは、各ターゲットDB毎に最大起動数が設定できますが、必要なライセンス数は
必要なライセンス数 = 各ターゲットDBの最大起動数 の合計 + Checker プロセス用の 1
となります。³

注：クライアント WEB からの WEBLINK アクセスにより自動起動されるプロセスに消費されるライセンス数がこれ以外に必要です。

¹ KCV 関数 は、JIS,EUC,SJIS<->UTF8 文字コードの変換を行うために、「今田 庸介 Imada's Kanji Code Converter Library 'imdkcvlib'」を OpenVMS に移植し、ECALL 化し、DSM から利用可能にしたものです。

² [UCI,VOL]へのグローバルの割当てや初期ゲートウェイ設定は、ST^ZGINI で行います。このルーチンを修正することで、[UCI,VOL]を変更することが可能です。

³ ただし Gateway プロセスは、固定起動設定と自動起動設定があるので開始時には、固定起動の全 Gateway プロセスと Checker プロセスが起動されます。

1.3 サンプルアプリケーションの構成

3 プログラム・ラベル一覧で一覧した複数の M プログラムを DSM^%RS 形式で、1 ルーチンセーブファイル [ZGV2.RS]として提供されます。

このファイルを Gateway プロセス動作 DSM 環境にリストアして下さい。

1.4 WEBLINK 設定

1.4.1 サーバ設定

このサンプルアプリケーションを利用するために必要な WEBLINK Server(Weblink の MGWLPN 変数に設定する値)を登録し、Server Access 用のパラメータを設定します。

Server Access パラメータ	値
Connection Method	TCP Socket
Server Type	Digital Standard M
Map_To_Server_Variable	FALSE
Maximum_String_Length	256
IP_Address	DSM DB サーバの ip アドレス
Base_TCP_PORT	
PCD_NameSpace	KMC,ZGA
Default_User_NameSpace	KMC,ZGA
Maximum Server Session	ライセンス数による
Server_response_timeout	300 秒

1.4.2 WEB からのアプリケーション指定変数

このサンプルアプリケーションの初期化ルーチンにより、WEBLINK からこのサンプルアプリケーションを参照できるように^MGWAPP グローバルに以下エントリを作成します。

^MGWAPP(n)	データ	
Z39V2	ST^ZGAPI	検索メニュー用 (EUC 表示用)
Z39MAN	ST^ZGMAN	管理メニュー用
Z39V2U	ST^ZGAPI	検索メニュー用 (UTF-8 表示用)

実際に検索メニューにアクセスする場合は、以下のような指定となります。

<http://<HOST>/scripts/mgwms32.dll?MGWLPN=server&MGWAPP=Z39V2>

初期化ルーチンで定義される上記値以外に、管理メニューの「DB 表示単位登録・修正削除」で管理される DB 表示単位・表示順・表示文字コードを区別するための ID も、^MGWAPP グローバルに追加されます。

^MGWAPP(n)	データ	
表示単位 ID	ST^ZGAPI	DB 表示単位・表示順・文字コードを区別してアクセスするための変更可能なエントリ

1.5 処理概要

WWW サンプルアプリケーションが、Gateway プロセス経由でターゲット要求を出す基本的な処理の流れを示します。

1.5.1 初期化の処理

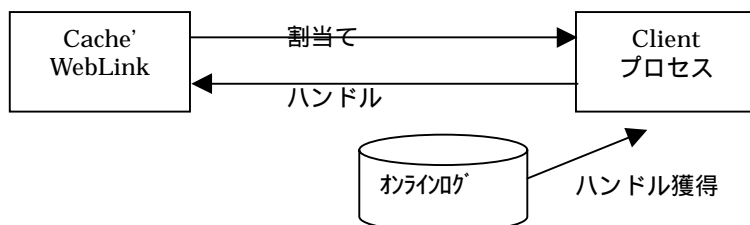


図 1-1 初期化の処理

- Cache' WebLink よりアプリケーション起動要求が行われ、Client プロセスが割り当てられます。
- Client プロセス は クライアント・ハンドル⁴ を取得します。

1.5.2 要求の処理

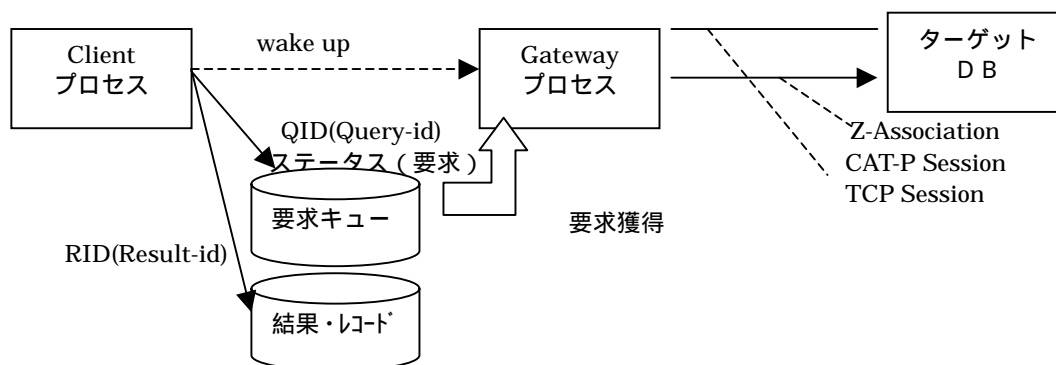


図 1-2 要求の処理

⁴ セッションを識別するための内部 ID。以後の Gateway プロセスへの要求に利用します。

- Client プロセスは、クエリに対して、Query-ID(QID)を獲得し、又同時に各ターゲット DB 毎の結果・レコードを管理する Result-id (RID) を獲得します。プロセスフリーキューから空き Gateway プロセスを獲得、各ターゲット DB 毎の要求キューへステータス(要求)でキューイングし、Gateway プロセスを WAKE UP します。
- Gateway プロセスは、特定の ターゲット DB 毎に起動(複数)され、要求キューにエントリされたクエリを参照し、各ターゲット DB にクエリを発行します。

1.5.3 応答獲得の処理

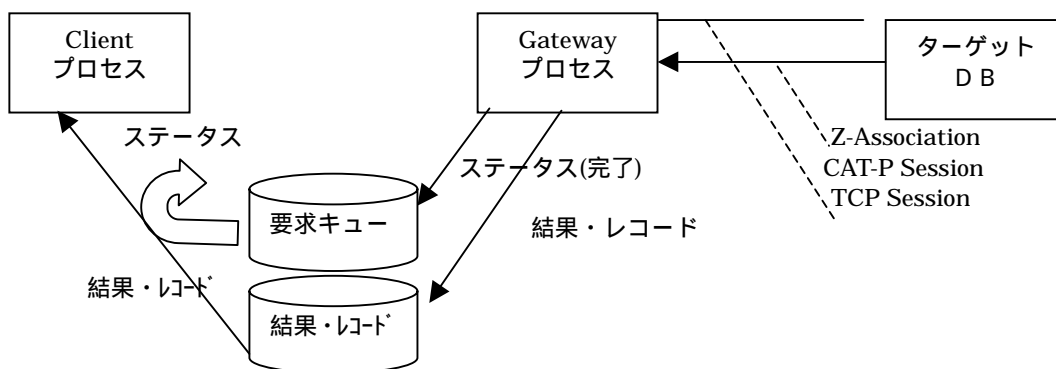


図 1-3 応答獲得の処理

- Gateway プロセスは ターゲット DB より検索結果又はレコードを取得し、RID に対応する結果・レコードファイルに出力し、QID を要求キューへステータス(完了)でキューイングします。
- Client プロセスは、定期的に要求キューチェックし、QID のステータス(完了)になった時点で、RID より結果・レコードファイルより検索結果やレコードを獲得します。

2 機能

以下に、本サンプルアプリケーションの機能について説明します。

2.1 Gateway 機能

Gateway は、DSM のユーザライセンスに制限がある環境において、複数のクライアントより Z39.50 ターゲット及び NACSIS-CAT へのネゴシエーションを一括で行うためのプロセスです。

ターゲット/DB 毎の Gateway プロセスが、起動処理 (Gateway・Checker プロセスの起動と停止)⁵により作成されます。(複数可)

稼動していない Gateway プロセスは、DB 毎のプロセスフリーキューにより管理され、Client プロセスからの Wakeup 指示を待ちます。

各 Z39.50 ターゲット、CAT-P のセッション管理を行います。ターゲットタイプを Z39.50 系、CAT-P に分けて管理します。

[Z39.50]

State-Aware な接続形態であるため、TCP セッションを保持します。Gateway プロセスの停止指示があった場合または、Checker プロセスの解放指示により一定時間アクセスがなかった場合に、Z-association のクローズ要求を発行します。

[CAT-P]

http を基本としているため、TCP セッションを保持しません。Z39.50 同様、Gateway プロセスの停止指示があった場合または、Checker プロセスの解放指示により一定時間アクセスがなかった場合、CAT-P サーバに対して、全ての Handle の開放要求を発行します。

Z39.50 系の場合で、z39.50 client パラメータファイルを作成し、個別に recsyn を指定します。

自動起動 Gateway プロセスは、アソシエーション解放実行回数が一定回数を越えた場合は終了処理を行い停止します。

要求キューファイルからステータス(要求)の QUERY-ID を獲得することによってターゲット DB に対する処理を開始します。

同一クライアント同一 DB に対する要求(QUERY-ID)は、前回担当 Gateway プロセスが担当します。(ただしタイマーによるアソシエーション解放が行われた場合は初回と同様フリープロセスを確保して行います。)

要求キューファイルからステータス(要求)の QUERY-ID が獲得できない(存在しない)場合、プロセスフリーキューに自ら登録し、ハイパネートします。

1 Gateway プロセスにより、複数クライアント要求に対応します。Z39.50 に named result set がサポートされていない場合は、reference-id を変更することにより 32⁶クライアントまで対応します。

クライアントからの最初の要求時に Reference-id を作成し、要求キューファイルに登録します。CAT-P の場合はクライアントハンドルが対応し、Z39.50 の場合は named result set サポート有無により Gateway プロセスの pid またはクライアントハンドルのいずれかとなります。

ターゲット接続管理-Z39.50/CAT-P による INIT 済み Reference-id をグローバル上で管理することにより、(再)INIT 処理必要有無を判断し INIT 処理を行い各 Target とセッションを確立します。

Result set 名は、named result set をターゲットがサポートする場合は、query-id と同一値となります。サポートしない場合は、"default"⁷となります。

ターゲットのエンコーディングにより SEARCH QUERY 文をコード変換します。⁸

SEARCH、PRESENT の各メソッドについてネゴシエーションを実行します。Z39.50、CAT-P の各 API を Call します。

アソシエーション解放済み QUERY-ID に対する PRESENT 要求については、再度 SEARCH を行った後、PRESENT を実行します。

⁵ Checker プロセスによって、起動される場合もあります。

⁶ Z39.50 Client API の最大 32 reference-id の制限事項によります。

⁷ 複数クライアント間の重複をさけるために reference-id を使って区別します。また同一クライアントにおける直前の SEARCH 以外の PRESENT 要求は再度 SEARCH 要求を実施した後、PRESENT 要求を行うことで対応します。

⁸ 提供している変換は、EUC->JIS、EUC->EACC、EUC->SJIS、EUC->UTF8 です。PRESENT の受信結果についての逆変換は Gateway では行わず、web アプリケーション側で出力時に行います

Z39 で SEARCH/PRESENT が失敗した場合、アソシエーションを解放し、再度 INIT 処理が次回行えるように確立済み Reference-id をグローバルから削除します。

Z39.50/CAT-P SEARCH/PRESENT コマンド実行結果を結果ファイルへ出力し、QUERY-ID 要求キューにステータス(完了)をキューイングします。

異常終了した場合、ジョブ再起動フラグ値によっては、Gateway プロセスを再起動します。

2.2 Checker 機能

Checker は、以下の機能を実現するプロセスです。

起動処理により他の Gateway プロセスと同時にシステムで 1 プロセス起動されます。

Z39.50 用 Gateway プロセスの最終処理時刻を監視し、タイマによってアソシエーション解放指示を各 Gateway プロセスへ出します。

定時起動によって、ゲートウェイ設定パラメータで指定した保存期間を残してログファイル(クライアント、要求、結果)を削除します。

自動起動と設定されている DB に対するクライアント検索要求を監視し、該当 DB の Gateway プロセスを起動します。

異常終了した場合、ジョブ再起動フラグ値によっては、Checker プロセスを再起動します。

2.3 Gateway・Checker プロセスの起動と停止

Gateway プロセス、Checker プロセスは、以下の DSM プログラムまたはシステム管理メニューからの開始/停止指示を実行することで、起動、停止を行います。

2.3.1 開始

ターゲット情報(^ZGSYS)にもとずきターゲット DB 毎の Gateway プロセス(DB 毎に複数有り)を起動します。

起動例)

```
$ DSM START^ZGSS
```

CHECKER プロセスを起動します。

2.3.2 停止

Gateway プロセスと CHECKER プロセスに対して停止指示を出します。

停止例)

```
$ DSM STOP^ZGSS
```

ゲートウェイ設定パラメータで指定した保存期間を残してログファイル(クライアント、要求、結果)を削除します。

2.4 Clientプロセスとのインターフェース

Client プロセスから Gateway プロセスにアクセスするためのプログラミングインターフェースです。複数の Gateway に対するアクセスなどの機能を提供します。

2.4.1 初期化

クライアント・ハンドルの獲得とオンラインログの初期化を行います。

Calling Seq.

S STS=\$\$INIT^ZGSND(%CGIEVAR,APP,DAY,.,HANDLE)

Parameters

Input

CGIVAR %CGIEVAR (WEBLINK var)

APP %KEY("MGWAPP") アプリケーション ID

Output

DAY -クライアント・ログインシステム日 (+\$H)

HANDLE -クライアント・ハンドル

戻り値

STS ステータス

0: 成功

-5: リトライオーバー

-4: 異常終了

2.4.2 SEARCH 要求

要求キューファイルより Query-id(QID)を獲得し QID による検索要求管理を行います。また ターゲット DB 毎の Result-id(RID)の獲得し、結果ファイルを初期化します。

SEARCH 要求のため DB 毎のフリーGateway プロセスをプロセスフリーキューから獲得し、WAKE UP させます。

プロセスフリーキューに空きプロセスが存在しない場合は、要求キューにステータス(要求)でキューイングのみ行います。

Calling Seq.

S STS=\$\$SRCQ^ZGSND(DAY,HANDLE,.,DBLIS,.,QUERY,.,QID)

Parameters

Input

DAY クライアント・ログインシステム日 (+\$H)

HANDLE クライアント・ハンドル

DBLIS DBLIS= 選択件数

DBLIS(i) = DB 番号

QUERY QUERY(DB 番号) = 行数

QUERY(DB 番号,n) = 検索式

Output

QID Query id

戻り値

STS ステータス

0: 成功

-1: ターゲット情報なし

-2: Z39.50/CAT-P 用検索式なし (パラメータエラー)

-3: ロックタイムアウト

-4: 異常終了

2.4.3 SEARCH 応答

SEARCH 要求に対する DB 毎の完了ステータスを取得します。

Calling Seq.

S STS=\$\$SRCR^ZGSND(DAY,HANDLE,QID,.DBSTS)

Parameters

Input

DAY クライアント・ログインシステム日 (+\$H)

HANDLE クライアント・ハンドル

QID Query id

Output

DBSTS DB 毎のステータス報告

DBSTS(DB 番号) = ステータス^ヒット件数^メッセージ

0: 完了(エラーなし)

1: 完了(エラー完了)

2: 未完了

戻り値

STS ステータス

0: 成功

1: 部分完了

2: 未完了

-2: パラメータエラー

-3: ロックタイムアウト

-4: 異常終了

2.4.4 PRESENT 要求

QID で照会ターゲット DB に対する PRESENT 要求を管理します。

前回 SEARCH 要求を行った Gateway プロセスに対して PRESENT 要求を要求キューへステータス(要求)をキューイングします。

前回 SEARCH 要求を行った Gateway プロセスが存在しない場合は、SEARCH 要求と同様にフリーキューから空きプロセスを獲得し、WAKE UP させます。空きプロセスが存在しない場合は、要求キューにキューイングのみ行います。

Calling Seq.

S STS=\$\$PRSQ^ZGSND(DAY,HANDLE,FROMTO,RANGE,QID,.DBLIS)

Parameters

Input

DAY クライアント・ログインシステム日 (+\$H)

HANDLE クライアント・ハンドル

FROMTO -range start-end

RANGE -range

QID Query id

DBLIS -PRESENT する DB 番号 (既に SEARCH した DB リスト内)

DBLIS = 件数

DBLIS(i) = DB 番号

Output

なし

戻り値

STS ステータス

0: 成功

1: 未完了

-2: パラメータエラー

-3: ロックタイムアウト

-4: 異常終了

-6: SEARCH 実行プロセスが存在せず

2.4.5 PRESENT 応答

PRESENT 要求に対する DB 毎の完了ステータスを取得します。

Calling Seq.

S STS=\$\$PRSR^ZGSND(DAY,HANDLE,QID,RNO,.,DBLIS,.,DBSTS)

Parameters

Input

DAY クライアント・ログインシステム日 (+SH)
 HANDLE クライアント・ハンドル
 QID Query id
 RNO 最終獲得レコード番号
 DBLIS -PRESENT した DB 番号 (既に SEARCH した DB リスト内)
 DBLIS = 件数
 DBLIS(i) = DB 番号

Output

DBSTS DB 毎のステータス
 DBSTS(DB 番号) = ステータス^ヒット件数^メッセージ
 0: 完了
 1: 完了 (エラー)
 2: 未完了

戻り値

STS ステータス
 0: 成功
 1: 部分完了
 2: 未完了
 -2: パラメータエラー
 -3: ロックタイムアウト
 -4: 異常終了

2.4.6 CLOSE 要求

過去に各ターゲット DB に要求した QUERY-ID で、CHECKER プロセスによる解放がされていないセッションについて CLOSE 要求用 QUERY-ID を確保し各 Gateway プロセスに解放指示をキューイングします。
 オンラインログに CLOSE 要求送信済み時刻を記録します。

Calling Seq.

S STS=\$\$CLOSE^ZGSND(DAY,HANDLE)

Parameters

Input

DAY クライアント・ログインシステム日 (+SH)
 HANDLE クライアント・ハンドル

戻り値

STS ステータス
 0: 成功
 -2: パラメータエラー
 -3: ロックタイムアウト
 -4: 異常終了
 -7: CLOSE 済み

2.4.7 データ獲得/更新関連

2.4.7.1 ^ZGQRY データ更新

pQRY^ZGSND と pQRYp^ZGSND は、QUERY-ID/DB 番号で管理されている ^ZGQRY グローバルを更新します。

DB 番号指定 : ^ZGQRY("M",QID,"db",DBNO,ITEM)の値
DB 番号 NULL : ^ZGQRY("M",QID,ITEM)の値

Calling Seq.

D pQRY^ZGSND(QID,DBNO,ITEM,DATA)
D pQRYp^ZGSND(QID,DBNO,ITEM,.VAR)

Parameters

Input

QID Query id
DBNO ターゲットのDB番号
ITEM 項目名
DATA 値
or
VAR 値を渡す変数 (ポインタ)。 Merge コマンドで更新される。

2.4.7.2 ^ZGQRY データ獲得

gQRY^ZGSND と gQRYp^ZGSND は、QUERY-ID/DB 番号で管理されている ^ZGQRY グローバルからデータ獲得します。

DB 番号指定 : ^ZGQRY("M",QID,"db",DBNO,ITEM)の値
DB 番号 NULL : ^ZGQRY("M",QID,ITEM)の値

Calling Seq.

S Value=\$\$gQRY^ZGSND(QID,DBNO,ITEM)
D gQRYp^ZGSND(QID,DBNO,ITEM,.VAR)

Parameters

Input

QID Query id
DBNO ターゲットのDB番号
ITEM 項目名

Output

Value 項目名に対応する値
or
VAR 値を得る変数 (ポインタ)。 Merge コマンドで得る。

2.4.7.3 レコード番号存在チェック

fRno^ZGSND は、Result -ID で管理されているレコードについて、レコード番号で指定するレコードが獲得済みか存在チェックします。

Calling Seq.

S Value=\$fRno^ZGSND(RID,RNO)

Parameters

Input

RID	Result ID
RNO	レコード番号

戻り値

Value	ステータス
	0 : 未獲得
	>0 : 獲得済み

2.4.7.4 レコードデータ獲得

gRnoDt^ZGSND は、Result -ID で管理されているレコードについて、レコード番号で指定するレコードデータを獲得します。

Calling Seq.

D gRnoDt^ZGSND(RID,RNO,DATA)

Parameters

Input

RID	Result ID
RNO	レコード番号

Output

DATA	レコードデータ
	DATA = 件数
	DATA(i) = データ

MERGE コマンドで^ZGRD("M",RID,2,"r")以下の値を得る。

2.4.8 その他

Query-id/Result-id で管理されているデータ獲得以外に ZGLIBN ルーチンでその他の情報を提供する関数を用意しています。

lcktime	^ZGSYS("CONFIG")に設定されている LOCK 時のタイムアウト時間を得る S timeout=\$lcktime^ZGLIBN(to) to タイムアウト対象("client","gateway") timeout ロックタイムアウト値
gUCIVOL	^ZGQRY,^ZGRD,^ZGJOB,^ZGLOG 各グローバルの[UCI,VOL]情報を得る D gUCIVOL^ZGLIBN 出力変数 %ZGUQRY,%ZGVQRY ^ZGQRY %ZGURD,%ZGVRD ^ZGRD %ZGULOG,%ZGVLOG ^ZGLOG %ZGUJOB,%ZGVJOB ^ZGJOB
getjpi	\$&ZLIB.%GETJPI (エラーなし) S val=\$getjpi^ZGLIBN(pid,item,prn) pid プロセス ID item プロセス属性 prn プロセス名(PID を指定しない場合) val 値 (プロセスが存在する場合) NULL (プロセスが存在しない場合)
wakeup	\$&ZLIB.%WAKEUP (エラーなし) S sts=\$wakeup^ZGLIBN(pid) pid WAKEUP させるプロセス ID sts ステータス(1 = 成功、 0 = エラー)
cZGOprc	Gateway プロセスをジョブ起動する S pid=\$cZGOprc^ZGLIBN(dbn,ii,logdir,dsmopt) dbn D B 番号(^ZGSYS("TARGET"),DBn) ii JOB 番号 logdir 標準出力ログファイルの作成ディレクトリ dsmopt J o b コマンドオプションパラメータ pid g wPID (PID = プロセス作成完了、 0 = 未完了)
cZGCprc	Checker プロセスをジョブ起動する D cZGCprc^ZGLIBN(logdir,dsmopt) logdir 標準出力ログファイルの作成ディレクトリ dsmopt J o b コマンドオプションパラメータ
pDBGtinf	トレース ON の場合にトレースログに値をセットする D pDBGtinf^ZGLIBN(location,value) location ^ZGDBG("T",location) サブスクリプトの値 value location にセットするデータ値
pDBGpinf	トレース ON の場合にトレースログに指定した変数の値 (ポインタ版) をセットする D pDBGpinf^ZGLIBN(location,.var) location ^ZGDBG("P",location) サブスクリプトの値 var ^ZGDBG("P",location,カウンタ)以下に MERGE される変数名

isRJB	異常終了時に Gateway プロセス、Checker プロセスを再起動するかを示す値を得る S flag=\$\$isRJB^ZGLIBN() flag ^ZGSYS("CONFIG","gateway","createjob") の値 (1=再起動、 0 = 再起動せず)
isDBG	デバック情報を記録するかを示す値を得る S flag=\$\$isDBG^ZGLIBN() flag ^ZGDBG("T")の値 (1=記録する、 0 = 記録しない)
gMGWLPN	Weblink のM環境名を得る S MGWLPN=\$\$gMGWLPN^ZGLIBN() MGWLPN %KEY("MGWLPN")の値
gWEBSRV	Web サーバ名を得る S WEBSRV=\$\$gWEBSRV^ZGLIBN() WEBSRV %CGIEVAR("SERVER_NAME")の値
gMGWLIB	cgi ライブラリ名を得る。(mgwms32.dll など) S MGWLIB=\$\$gMGWLIB^ZGLIBN() MGWLIB %KEY("MGWLIB")の値
gMGW	"http://"_\$\$gWEBSRV_\$\$gMGWLIB の値を得る S MGW=\$\$gMGW^ZGLIBN() MGW "http://"_\$\$gWEBSRV_\$\$gMGWLIB
isACC	外部ターゲットにアクセスできるかを示す値を得る S sts=\$\$isACC^ZGLIBN(cmd) cmd 実行コマンド(INIT,SEARCH,PRESENT) sts 外部ターゲットにアクセスできるかを示す値。 cmd NULL の場合 ^ZGDBG("NOACCESS")の値 (1=アクセス可、 0 = 不可) cmd 指定の場合^ZGDBG("NOACCESS",cmd)の値。
flush	flush^%mgw をコールする D flush^ZGLIBN()
pNRS2JOB	^ZGJOB("J",DBn,PID,"nrs")を更新する D pNRS2JOB^ZGLIBN(dbno,pid,nrs) dbno D B 番号 pid Gateway プロセスのプロセス ID nrs name result set サポート有無 (1 = 有、 0 = 無)
pNRS2SYS	^ZGSYS("TARGET",DBn,"nrs")を更新する D pNRS2SYS^ZGLIBN(dbno,nrs) dbno D B 番号 nrs name result set サポート有無 (1 = 有、 0 = 無)
isNRS	^ZGJOB("J",DBn,PID,"nrs")の値を得る S nrs=\$\$isNRS^ZGLIBN(dbno,pid) dbno D B 番号 pid Gateway プロセスのプロセス ID nrs named result set サポート値 (1 = 有、 0 = 無)

gNRSFLG	INIT の結果による Named result set サポート値を得る S nrs=\$\$gNRSFLG^ZGLIBN() nrs INIT の結果による Named result set サポート値 (1 = 有、 0 = 無) \$\$(\$E(\$G(%Z3RES("z","options")),15)="y":1,1:0)
isADB	指定 D B に検索が可能を得る S sts=\$\$isADB^ZGLIBN(dbno) dbno D B 番号 sts ターゲット D B にアクセス可能 (1 = 可能、 0 = 不可) 1) Gateway プロセスが起動されている場合 : 1 2) 自動起動 Gateway プロセスの場合でプロセスが存在しない 場合でかつ ^ZGSYS("TARGET",dbno,"count") が 1 以上で Checker プロセスが存在する場合 : 1 3) それ以外 : 0
gDB	DB リストを得る D gDB^ZGLIBN(dblis,qid,mgwapp) 引数なし ^ZGSYS("TARGET",dbno) の D B 番号を返す。 qid QUERY-ID (^ZGQRY("M",QID,"db") の D B 番号を得る場合に指定する。) 指定されない場合、 ^ZGSYS("TARGET") 以下の D B 番号を返す。 mgwapp D B 表示 ID この値が指定された場合、 ^ZGSYS("MGWAPP",,"dbs") の D B 番号 を返す。 dblis 出力 D B 番号リスト 1) qid 指定、 dblis = 件数 引数なしの場合 dblis(DB 番号)="" 2) mgwapp 指定 dblis = 件数 dblis(i=1:1:件数)=D B 番号 表示順のためサブスクリプトにいれない。
gDBitem	^ZGSYS("TARGET") の D B 毎の情報を得る S val=\$\$gDBitem^ZGLIBN(dbno,item) dbno DB 番号 item 項目名 val item に対応する ^ZGSYS("TARGET",dbno,item) の値

2.4.9 文字コード変換

文字コード変換するための間数を ZGCCNV ルーチンで提供しています。

DeCode(dat,code)	他コード -> EUC S Data=\$\$DeCode^ZGCCNV(Data{,Code}) dat : 変換対象文字列 code : 変換対象文字列の文字セット EUC,SJIS,JIS,UTF8,EACC return value : EUC 変換後文字列
EnCode(dat,code)	EUC -> 他コード S Data=\$\$EeCode^ZGCCNV(Data{,Code}) dat : 変換対象文字列 code : 変換後文字列の文字セット EUC,SJIS,JIS,UTF8,EACC return value : 指定文字セットの変換後文字列

CnvCode(dat,fcode,tcode,max) 他コード -> 他コード
 S Data=\$\$CnvCode^ZGCCNV(Data,{FCode,TCode,max})
 dat : 変換対象文字列
 fcode : 変換前文字列の文字セット
 tcode : 変換後文字列の文字セット EUC,SJIS,JIS,UTF8,EACC
 max : 変換後最大バイト数
 return value : 指定文字セットの変換後文字列

Length(dat,tcode,fcode) コード変換後の長さを得る
 S len=\$\$Length^ZGCCNV(Data,TCode,{Fcode})
 dat : 変換対象文字列
 tcode : 変換後文字列の文字セット EUC,SJIS,JIS,UTF8,EACC
 fcode : 変換前文字セット [EUC] EUC,SJIS,JIS,UTF8,EACC
 return value : 指定文字セットの変換後文字列の長さ (バイト)

EXTRACT(Istr,Imax,Ifcode,Itcode,Obytes,Owords,Flag)
 S Ostr=\$\$EXTRACT^ZGCCNV(Istr,Imax,Ifcode,Itcode,.Obytes,.Owords,Flag)
 文字変換付き文字列 EXTRACT
 指定文字セット(Ifcode)の入力文字列(Istr)を指定文字セット(Itcode)に変換して、
 先頭文字から最大バイト数(Imax) まで文字列(Ostr) を切り出す。
 切り出された文字列のバイト数(Obytes) と実際に変換対象となった文字数(Owords)
 を返す。
 EACC の場合 EACC から他のコード(UTF8 以外)に変換する時 EUC に変換後、
 そのコードに変換する。EACC -> SJIS は、EACC -> EUC -> SJIS となる。
 EACC <-> UTF8, EACC <-> EUC は直接変換する。

Istr : 切り出し対象入力文字列
 Imax : 切り出し最大バイト数
 default = 512、最大 512 バイトまで指定可
 Ifcode : 入力文字列の文字セット
 default = EUC EUC,SJIS,JIS,UTF8,EACC,UCS
 Itcode : 出力文字列の文字セット
 default = EUC EUC,SJIS,JIS,UTF8,EACC,UCS
 Flag : Encode/DeCode からの呼び出し = 1
 Obytes : 出力文字のバイト数
 Owords : 変換対象文字数
 return value : 切り出し文字列

2.4.10 フォーマット変換

ZGCOM ルーチンで、MARC21(USMARC)、SUTRS の各レコード形式を HTML 形式で表示するためのフォーマット変換関数を提供しています。

MARC(P1,P2,P3,P4,P5) MARC21 形式のデータより項目単位にデータを取り出す。
 P1 - MARC データ
 P2 - DSM データ
 P2(tag)= 行数
 P2(tag,n) = データ
 P3 - タグ出現順序情報
 P3=回数
 P3(n) = タグ番号
 P4 - MARC21 CharSet 情報による EnCode
 P5 - Leader CharSet 情報により EnCode(UTF8-EACC)を切換えるか
 1:切換える、0:切換えない

SUTRS(P1,P2,P3,P4) SUTRS 形式のデータより項目単位にデータを取り出す。
 P1 - SUTRS データ
 P2 - DSM データ

P2(tag)= 行数
 P2(tag,n) = データ
 P3 - タグ出現順序情報
 P3=回数
 P3(n) = タグ番号
 P4 - optsyn
 SUTRS でもターゲット毎に編集しなければならない場合を考慮

Craws(P1,P2,P3,P4,P5) ; Format change (raw data / SUTRS)

P1 - フィールド名
 P2 - 変換前データ
 P3 - 変換後データ
 P4 - データ encode
 P5 - 出力 encode

Craw(P1,P2,P3,P4,P5) ; Format change (raw data / MARC21)

P1 - タグ番号
 P2 - 変換前データ
 P3 - 変換後データ
 P4 - データ encode
 P5 - 出力 encode

Cful(P1,P2,P3); Format change (text data / MARC21)

P1 - 変換前データ
 P2 - データ encode
 P3 - 出力 encode
 戻り値 - 変換後データ

Cfuls(P1,P2,P3) Format change (text data / SUTRS)

P1 - 変換前データ
 P2 - データ encode
 P3 - 出力 encode
 戻り値 - 変換後データ

2.4.11 プログラムの流れ

上記プログラム I/F を利用したアプリケーションプログラムの処理の流れを示します。

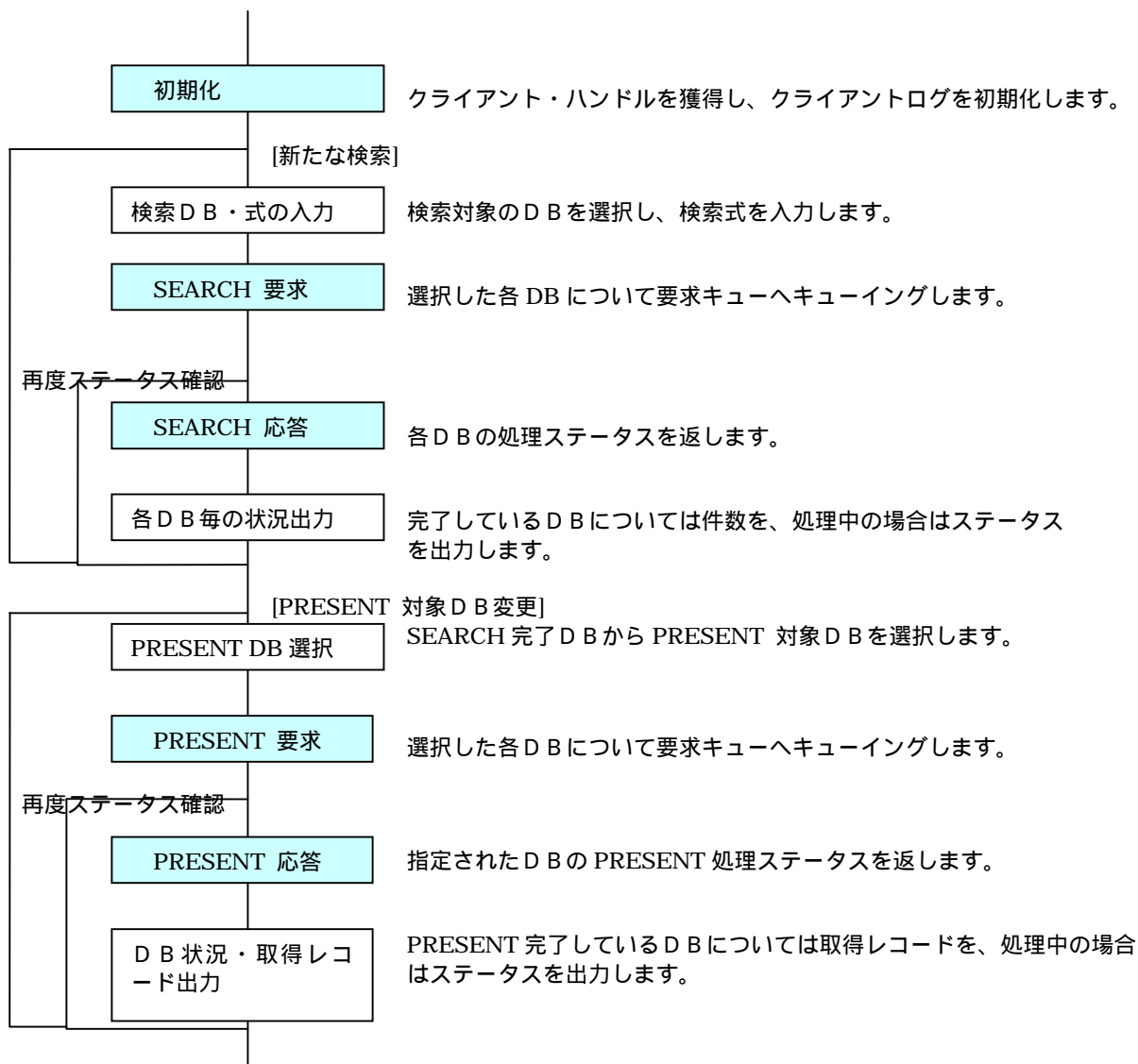


図 2-1 プログラムの流れ

注) フローでわかるように、各応答 I/F はターゲットDBの応答ステータスを返す仕組みになっています。処理中のDBが存在する場合は、アプリケーションプログラム側で、定期的に応答 I/F を呼び出しDBのステータスを監視する必要があります。逆にこのような I/F なので他のDBが検索中であっても検索完了してDBについてはPRESENT要求を発行することができます。

2.5 システム管理

2.5.1 DB登録・修正・削除機能

DB番号で管理されるDB情報を管理します。

新たなDB情報（以下項目）を^ZGSYS("TARGET")に登録します。

項目名	内容
DB番号	ターゲットDBを識別するための通番(1以上)
DB名称	表示用の値
DB ID名	Z39.50 Client API のパラメータとしてのデータベース名
IPアドレス	Z39.50 Client API のパラメータとしてのIPアドレス
ポート番号	Z39.50 Client API のパラメータとしてのポート番号 210 (省略値)
ユーザ名	Z39.59/CAT-P Client API のパラメータとしてのユーザ名
パスワード	Z39.59/CAT-P Client API のパラメータとしてのパスワード
Z39.50/CAT-P 識別	ターゲットDB向けプロトコルがZ39.50かCAT-Pを示す値 0: Z39.50 (省略値) 1: CAT-P
Named Result Set サポート (NRSS)	ターゲットDBがNamed Result Setを提供しているかを示す値 0: 未提供 1: 提供 (省略値)
レコードタイプ	Z39.50 Prefrecsyn 値。 MARC21(USMARC)、SUTRS、OTHERのいずれかを選択。 Z39.50/CAT-P 識別=1の場合はOTHER固定となる。
特別レコード編集	PrefrecsynがSUTRSの場合の特別編集指定。 none または フィールド: 形式
起動方法	開始時に起動するか、クライアント要求時に自動的に起動されるかを示す値 0: 固定起動 開始時 (省略値) 1: 自動起動 クライアント要求時
最大起動数	1 (省略値)
クライアント数/1プロセス	1 Gateway プロセスあたりの最大担当クライアント数。 0: NRSS=1の場合無制限を意味する。NRSS=0の場合は32と同値 (省略値) >0: 最大担当クライアント数
コード	ターゲットのエンコードを指定する。 EUC、EACC、JIS7、SJIS、UTF8、無変換のいずれかを指定。 EUC (SDEC) は、Gateway で利用しているコード、コードによってSEARCH/PRESENT時に変換が行われる。
MARC21 Leader Character Code 対応	PrefrecsynがMARC21の場合に、LeaderのCharacter Codeの値によりMARC8(EACC)とUTF8にコードを変更するかを指定する。 1: Leader Character Codeを有効として変更する。 0: 無効として上記指定コードを利用する。
ユーザ検索対象	アプリケーションからターゲットDBの指定が可能かを示す値。 0: 可能 (省略値) 1: 禁止

DB番号指定で既存DB情報の修正および、DB情報の削除を行います。

2.5.2 ゲットウェイ設定

Gateway プロセス、Checker プロセス、クライアントプログラム I/F が利用するパラメータを管理します。
以下項目を^ZGSYS("CONFIG")に登録します。

項目名	内容
解放間隔(アソシエーション)	Gateway プロセスの最終処理時刻からこの間のセッションを維持するための値 (秒)。60~1800 までの値。 300 (省略値)
解放・自動起動監視間隔	Checker プロセスのが解放処理および自動起動監視を行う間隔 (秒)。1~60 までの値。 3 (省略値)
ログ・ディレクトリ	Gateway プロセス、Checker プロセスの Job ログファイルを作成するディレクトリを示す値。 "" (省略値)
ログ・保存期間	オンラインログ・クライアント要求・結果ファイルなどグローバルの保存期間を示す値 (日)。2~60 までの値。 7 (省略値)
ログ・クリア開始時刻	ログ削除処理を行う時刻。00:00:00~23:59:59 までの時刻かログクリア禁止を示す 99:99:99。 06:00:00 (省略値)
ログイン・リトライ	クライアント接続競合によりクライアントハンドルが同一値となった場合に、何回獲得リトライするかを示す値。1~1000 までの値。 10 (省略値)
SEARCH 監視間隔	SEARCH 応答を確認する間隔 (秒)。0.1~10 までの値 0.5 (省略値)
SEARCH 監視リトライ	SEARCH 完了まで SEARCH 応答を最大何回行うかを示す値。1~60 までの値。 15 (省略値)
PRESENT 監視間隔	PRESENT 応答を確認する間隔 (秒)。0.1~10 までの値 0.5 (省略値)
PRESENT 監視リトライ	PRESENT 完了まで PRESENT 応答を最大何回行うかを示す値。1~60 までの値。 15 (省略値)
自動GWプロセス最大数	クライアント要求により Checker プロセスにより起動される GW プロセスの合計数。1~の値。 5 (省略値)
自動GWプロセス停止リトライ	自動起動 GW プロセスがアソシエーション解放を何回行うと停止するかを示す値。1~60 の値。 5 (省略値)
クライアント・ロックタイムアウト	クライアント I/F でのグローバル・ロック・タイムアウト値 (秒)。1~60 の値。 3 (省略値)
GWプロセス・ロックタイムアウト	GW プロセスがグローバル・ロック・タイムアウト値 (秒)。1~60 の値。V2.0 プログラムは参照していない。 5 (省略値)
GW・CHECKER 異常終了時再起動	Gateway プロセス、Checker プロセスが異常終了した場合に自動再起動を行うかを示す値。 0:再起動しない 1:再起動する (省略値)
検索履歴出力行数	0 は、履歴を出力しないの意味、1 以上は 1 ページに出力する検索履歴行数を示す値。 10(省略値)
検索履歴出力 DB 数	1 検索式あたりで出力する最大ターゲット DB 数。 10 (省略値)
所蔵情報出力件数	NII-CAT で B HOLD/SHOLD 情報を 1 ページに最大何件出力するかを示す値。 10 (省略値)

2.5.3 開始・停止(WEB)

Gateway プロセス、Checker プロセスの起動・停止を管理します。

開始ボタンによりターゲット情報 (^ZGSYS) にもとずきターゲット/DB 毎の Gateway プロセス(DB 毎に複数有り)と CHECKER プロセスを起動します。⁹

停止ボタンにより全 Gateway プロセスと CHECKER プロセスに対して停止指示を出します。

DB 毎のチェックボックスをチェックした上で停止指示することで、特定DB についてのみ停止指示が行えます。

DB 毎のチェックボックスをチェックした上で開始指示することで、特定DB についてのみ開始指示が行えます。Checker プロセスは、最初の DB 毎の Gateway プロセス起動と同時に起動されます。

2.5.4 クライアント接続情報監視

クライアント・ログよりクライアント接続情報を出力します。

ログイン接続したシステム日・開始時刻範囲を指定し、クライアント・ログより接続情報 (接続元、接続時刻、最終処理時刻、表示単位、直近の QUERY-ID リスト) を出力します。

QUERY-ID リストより、QUERY-ID に対応した以下項目を出力します。

- 検索式 (Z39.50/CAT-P)
- DB 番号/ID 名
- DB 毎のヒット件数
- 検索応答時間
- 獲得レコード数

2.5.5 検索キー登録・修正

アプリケーションから各ターゲットDB について検索を行うときに指定できるキー (インデックス) 表示名を登録します。

以下項目を ^ZGSYS("KEY") に登録します。

- No
- 表示名

2.5.6 DB 毎 BIB-1 属性登録

DB 毎に上記検索キーと実際の BIB-1 属性(Z39.50 の場合)やフィールド名 (NII-CAT の場合) を関係付けて登録します。

DB が Z39.50 の場合は、以下項目を ^ZGSYS("QUERY") に登録します。

- No
- BIB-1 属性(Use,Relation,Position,Structure,Truncation,Completeness)

DB が NII-CATP の場合は、以下項目を ^ZGSYS("QUERY") に登録します。

- No
- フィールド名

2.5.7 DB 毎 一覧表示/詳細表示(TEXT)項目登録

DB 毎に Data list 表示時と詳細表示時に、表示される項目を登録します。

DB が Z39.50、Prefrecsyn が MARC21(USMARC)の場合は、以下項目を ^ZGSYS("FORM") に登録します。

- 区切り文字
- MARC のタグ番号 (連続する3桁の数字でタグ番号と認識する。)

DB が Z39.50、Prefrecsyn が SUTRS の場合は、以下項目を ^ZGSYS("FORM") に登録します。

- 区切り文字
- フィールド名 (入力時は、フィールド名+コロン ':' で、フィールドと認識する。)

⁹ DB 情報を修正した場合や、プロセスが異常終了した場合など、再開指示で足りないプロセスが起動できます。

DBがNII-CATの場合は、以下項目を^ZGSYS("FORM")に登録します。

- 区切り文字
- フィールド名 (入力時は、連続する英数字で、フィールドと認識する。)

2.5.8 DB表示単位登録・修正・削除

SEARCH/PRESENTを行えるターゲットDB(複数)を、DB表示単位IDで管理します。

この表示単位IDは、URLで指定するMGWAPP変数の値となります。

新たなDB表示単位情報(以下項目)を^ZGSYS("MGWAPP")と^MGWAPPに登録します。

^ZGSYS("MGWAPP",表示単位ID)

項目名	内容
DB番号リスト("dbs")	検索画面に表示するDB番号リスト。 存在するDB番号を表示順にカンマで区切り指定する。
DB表示単位名称	表示単位IDの名称(注釈)
表示文字コード	WEB表示文字コード EUC (euc-jp) SJIS (shift_jis) UTF8 (utf-8)

^MGWAPP(表示単位ID) = ST^ZGAPI

登録済みのDB表示単位指定で、既存情報を修正します。または、^ZGSYS("MGWAPP")と^MGWAPPから表示単位IDを削除します。

2.6 動作環境初期化

Gateway動作用DSM環境を初期化します。

^ZGQRY,^ZGLOG,^ZGJOB,^ZGRD グローバルを特定に[UCI,VOL]に作成します。

^ZGSYSに以下のデフォルト情報をセットします。

- ターゲット情報(^ZGSYS("TARGET"))
- DB表示単位情報(^ZGSYS("MGWAPP"))
- ゲートウェイ設定情報(^ZGSYS("CONFIG"))
- 検索キー情報(^ZGSYS("KEY"))
- 検索キー対応BIB-1情報(^ZGSYS("QUERY"))
- 表示項目情報(^ZGSYS("FORM"))
- BIB-1属性(^ZGSYS("BIB-1"))
- NII-CAT DBフィールド情報(^ZGSYS("NII-CAT"))
- MARCタグ番号(^ZGSYS("MARC"))

^MGWAPPに検索アプリケーション(Z39V2、Z39V2U)とシステム管理アプリケーション(Z39MAN)のシンボルとルーチンとの対応をセットします。

起動例)

```
$ DSM ST^ZGINI
```

3 プログラム・ラベル一覧

提供されるプログラムとそのプログラム内主要ラベルの一覧を以下示します。

3.1 ZGINI,ZGINI1

^ZGSYS のテンプレートと、^ZGQRY、^ZGLOG、^ZGJOB、^ZGRD のグローバル初期化。

ラベル	内容
ZGINI	Z-origin gateway ^ZGSYS initialize
ST	main entry
INIT	create ^ZGSYS,^ZGQRY,^ZGLOG,^ZGJOB,^ZGRD
TARGET	^ZGSYS("TARGET") initialize
CONFIG	^ZGSYS("CONFIG"),^ZGSYS("MGWAPP") initialize
KEY	^ZGSYS("KEY") initialize
BIB1	^ZGSYS("BIB-1") initialize
NII	^ZGSYS("NII-CAT") initialize
FORM	^ZGSYS("FORM") initialize

ラベル	内容
ZGINI1	Z-origin gateway ^ZGSYS initialize 2
MARC	^ZGSYS("MARC") initialize

3.2 ZGSS

Gateway プロセス、Checker プロセスの起動・停止。

ラベル	内容
ZGSS	Z-origin gateway start/stop
START	Gateway プロセス, Checker プロセスの起動
STOP	Gateway プロセス, Checker プロセスの停止
STOP2	個別 DB Gateway でのみの停止
START2	個別 D B Gateway でのみの起動

3.3 ZGCHKR

Checker プロセス用処理。

ラベル	内容
ZGCHKR	Z-origin gateway checker
ST	main entry
chkFJ	Free queue の存在チェック
gGWlst	自動起動の Gateway プロセスリスト作成
clslogg	^ZGRD,^ZGQRY,^ZGLOG 削除

3.4 ZGORGD

Gateway プロセス用処理。

ラベル	内容
ZGORGD	Z-origin gateway daemon
ST	main entry
LOOP	main loop
EXIT	停止
TRAP	異常終了
Finish	終了処理
CMD	Z39.50/CAT-P SEARCH/PRESENT/CLOSE 処理
putRD	RID 情報 ^ZGRD 更新
chkQID	未処理 QID 存在確認
gQID	未処理 QID 獲得
pQIDsts	QID ステータスの更新
CLOSE	アソシエーション解放
gRefid	get reference-id
gRsn	get result set name
ZINIT	Z - Initial
ZCLS	Z - Close
ZSRC	Z - Search
ZPRS	Z - Present
ZSCN	Z - Scan
CSRC	CAT-P - Search
CPRS	CAT-P - Present
ZERR	Z39.50 API error message logging
CERR	CAT-P API error message logging
EnCode	EUC->他コード変換
DeCode	他コード->EUC 変換
SETdum	for test (Set dummy response)

3.5 ZGSND

ZGAPI から Gateway プロセスへ要求を発行するための I/F。

ラベル	内容
ZGSND	Z-origin gateway requester
INIT	Get gateway request handler
SRCQ	Search Request
findFJ	フリープロセスを得る
getQID	QID 確保
getRID	RID 確保
SRCR	Search Result
PRSQ	Present Query
PRSR	Present Result
CLOSE	Release Gateway request handler
gQRY	^ZGQRY("M",QID,"db",DB 番号,item) を得る
gQRYp	^ZGQRY("M",QID,"db",DB 番号,item) を得る(MERGE)
pQRY	^ZGQRY("M",QID,"db",DB 番号,item) を更新
pQRYp	^ZGQRY("M",QID,"db",DB 番号,item) を更新(MERGE)
fRno	^ZGRD("M",RID,2,"r",レコード番号)の存在確認
gRnoDt	^ZGRD("M",RID,2,"r",レコード番号)のデータ獲得

3.6 ZGAPI

検索システム (HTML/WEBLINK) 処理 (その1)

ラベル	内容
ZGAPI	Z-Gateway CGI 1
ST	Main entry
W	表示エンコード付 WRITE (EUC->表示コード)
WW	表示エンコード付 WRITE (結果コード->表示コード)
HPerr	Error message
HP3	データ一覧出力等
HP2	検索処理の実行、検索結果一覧出力等
ChkOpr	演算子とキーワードの組み合わせチェック V2.0
MkQry	検索式の編集 V2.0
HP21	検索式の編集 V1.1
HP1	ターゲット選択、検索式の入力等
HP11	DB リスト表示
Head	Write Header
Style	Write Style
Color	Write Color
Title	Write Title
INIT	Initialize
KEY	Search Index list
LOG	%KEY の保存

3.7 ZGAPI1

検索システム (HTML/WEBLINK) 処理 (その2)

ラベル	内容
ZGAPI1	Z-Gateway CGI 2
W	表示エンコード付 WRITE (EUC->表示コード)
WW	表示エンコード付 WRITE (結果コード->表示コード)
HP0	クローズ処理
HP5	検索履歴画面
HPprs	レコード取得中画面
hidden	%KEY("Z*")変数 hidden 変数出力
addN	データリスト改ページ処理
CPAGE	カレントのレコード番号よりデータリストの表示範囲作成
HP4	データ詳細出力等
HP4err	HP 4 用エラーメッセージ出力
NiiHold	Nii-CAT の BHOLD/SHOLD 処理
holdsrq	*hold への SEARCH 要求/結果受信
holdprq	*hold への PRESENT 要求/結果受信
hp45	NII-CAT (BHOLD/SHOLD)Convert Text
hp46	NII-CAT(BHOLD/SHOLD)Convert Marc
hp41	Z39.50 Convert Text
hp42	Z39.50 Convert Marc
hp43	NII-CAT(BOOK/SERIAL)Convert Text
hp44	NII-CAT(BOOK/SERIAL)Convert Marc

3.8 ZGLIBN

ZGAPI、ZGSND、ZGCHKR、ZGORGD など利用されているサブルーチン群1。

ラベル	内容
ZGLIBN	Z-origin gateway library
lcktime	get lock timeout value
gUCIVOL	^ZGQRY, ^ZGRD, ^ZGLOG などグローバル配置%ZGU*, %ZGV* 変数獲得
getjpi	%GETJPI (no error)
wakeup	%WAKE (no error)
cZGOprc	create origin dameon
cZGCprc	create checker process
pDBGtinf	Debug Trace 情報の記録
pDBGpinf	Debug Parameter 情報の記録
isRJB()	再起動情報獲得
isDBG()	トレースフラグ獲得
pDBGon	トレースセット
pDBGoff	トレースオフ
gMGWLPN	Weblink の M の環境名を返す
gWEBSRV	WebServer 名を返す ex. bobo.tko.dec.com
isACC	外部の TARGET にアクセスできるかどうか
pNRS2JOB	プロセス毎の Named Result Set の使用可能フラグを設定
pNRS2SYS	Named Result Set の使用可能フラグを設定
isNRS	Named Result Set の使用可能フラグの値を返す
gNRSFLG	INIT の結果, Named Result Set を使用できるかどうかを返す
isADB	利用可能 DB ?
gDB	DB lis を得る
gDBitem	DB 情報を得る
gCNF	^ZGSYS("CONFIG") 情報を得る
isOptr	Option syntax record 存在チェック
isOptm	Option syntax record Marc 出力サポートチェック

3.9 ZGLIBC

ZGAPI、ZGSND、ZGCHKR、ZGORGD など利用されているサブルーチン群2。

ラベル	内容
ZGLIBC	Z-origin gateway library for Cahe' vs DSM
UPCASE	大文字変換
HD	10 進数変換
DH	16 進数変換
BL	バイト長

3.10ZGCCNV

UTF8、SJIS、JIS、EACC と EUC 間の相互変換など文字コードに関する関数。

ラベル	内容
ZGCCNV	Z39.50 Gateway/Target Tools (Charactor Conversion)
CharCode	ポート毎のエンコード定義を得る
DeCode	他コードから EUC への変換エントリ
EnCode	EUC から他コードへの変換エントリ
CnvCode	他コードから他コードへの変換エントリ
Length	コード変換後の長さ(バイト)を得る
EXTRACT	文字変換付き文字列 EXTRACT
EACC2EUC	EACC から EUC への変換
EUC2EACC	EUC から EACC への変換
EACC2UTF	EACC から UTF-8 への直接変換(CJK の CK 表示対応)
UTF2EACC	UTF-8 から EACC への直接変換
EACC2EAC	EACC から EACC へ出力バ 1 長を指定した EXTRACT 対応
JIS72EUC	JIS7 から EUC への変換(未使用)
EUC2JIS7	EUC から JIS7 への変換(未使用)

3.11ZGCOM

Z-Target の返送データのフォーマット変換等。

ラベル	内容
ZGCOM	Z39.50 Gateway Tools
MARC	MARC21 形式のデータより項目単位にデータを取り出す、Leader CharSet 情報より EnCode 情報を得る
SUTRS	SUTRS 形式のデータより項目単位にデータを取り出す(optrecsyn 利用)
Craw	Format change (raw data MARC21) データ Encode -> 出力 EnCode への変換
Craws	Format change (raw data SUTRS) データ Encode -> 出力 EnCode への変換
Cful	Format change (text data MARC21) データ Encode -> 出力 EnCode への変換
Cfuls	Format change (text data SUTRS) データ Encode -> 出力 EnCode への変換

3.12ZGCCOM

NACSIS CAT の返送データのフォーマット変換。

ラベル	内容
ZGCCOM	Z39.50 Gateway Tools (NACSIS-CAT)
MARC	NACSIS 形式のデータより項目単位にデータを取り出す
Craw	Format change (raw data) データ Encode -> 出力 EnCode への変換
Cful	Format change (text data) データ Encode -> 出力 EnCode への変換

3.13 ZGMAN

システム管理 メニュー、DB登録、修正削除登録 (HTML/WEBLINK) 処理。

ラベル	内容
ZGMAN	Z-Gateway management 1
ST	main entry
HPerr	error message
DelDB	DB 情報削除
AddDB	DB 情報登録
ChkDB	DB 情報型チェック
ListDB	DB 情報出力(確認用)
HP22	DB 修正・削除登録処理,登録確認
HP21	DB 修正・削除入力画面出力
HP20	DB 選択
HP11	DB 新規登録処理,登録確認
HP10	DB 新規登録入力画面出力
MENU	メニュー
INIT	Initialize

3.14 ZGMAN1

システム管理 ゲートウェイパラメータ設定 (HTML/WEBLINK) 処理。

ラベル	内容
ZGMAN1	Z-Gateway management 2
AddSYS	パラメータ情報設定
ChkSYS	パラメータ型チェック
ListSYS	登録内容出力(確認用)
HP31	パラメータ設定変更 [登録->登録確認/エラー出力]
HP30	パラメータ設定 DB 登録 [入力画面出力]

3.15 ZGMAN2

システム管理 開始・停止、クライアント接続情報 (HTML/WEBLINK) 処理。

ラベル	内容
ZGMAN2	Z-Gateway management 3
HP51	QUERY-ID 情報
HP50	クライアント接続情報
HP40	ゲートウェイ開始 / 停止

3.16 ZGMAN3

検索キー、DB 毎 BIB-1 属性登録 (HTML/WEBLINK) 処理。

ラベル	内容
ZGMAN3	Z-Gateway management 4
AddQRY	BIB-1 属性設定
ChkQRY	BIB-1 属性型チェック
ListQRY	BIB-1 属性登録内容出力(確認用)
HP72	BIB-1 属性登録確認 [登録->登録確認/エラー出力]
HP71	BIB-1/フィールド登録 [入力画面出力]
HP70	BIB-1 属性登録 DB 選択
AddKEY	検索キー設定
ChkKEY	検索キー型チェック
ListKEY	検索キー登録内容出力(確認用)
HP61	検索キー登録確認 [登録->登録確認/エラー出力]
HP60	検索キー登録 [入力画面出力]

3.17 ZGMAN4

DB 毎一覧・詳細(TEXT)表示の項目設定(HTML/WEBLINK) 処理。

ラベル	内容
ZGMAN4	Z-Gateway management 5
AddFORM	フォーム項目情報設定
Elm	入力項目の分解
ChkFORM	フォーム型チェック
ListFORM	フォーム登録内容出力(確認用)
HP82	フォーム登録確認(一覧/詳細) [登録->登録確認/エラー出力]
HP81	フォーム登録(一覧/詳細) [入力画面出力]
HP80	フォーム登録 DB 選択(一覧/詳細)

3.18ZGMAN5

DB 表示単位、登録、修正削除登録(HTML/WEBLINK) 処理。

ラベル	内容
ZGMAN5	Z-Gateway management 6
AddDBD	DB 表示単位情報登録/修正/削除
ChkDBD	DB 表示単位チェック
ListDB	DB 表示単位情報出力(確認用)
HP122	DB 表示単位修正・削除登録処理
HP121	DB 表示単位修正・削除入力画面出力
HP120	DB 表示単位選択
HP111	DB 表示単位新規登録処理,登録確認
HP110	DB 表示単位新規登録入力画面出力

4 プログラム処理

次の主要プログラムについて処理内容を示します。

- Checker プロセス(ST^ZGCHKR)
- 開始(START^ZGSS)
- 停止(STOP^ZGSS)
- Gateway プロセス(ST^ZGORGD)
- Gateway I/F(*^ZGSND)
- Gateway 検索システム(ST^ZGAPI)

4.1 Checker プロセス

- [初期処理]
 - 1) ^ZGSYS からコンフィグ情報を得る。
 - アソシエート解放時間
 - CHECKER プロセス SLEEP 時間
 - ログ保存期間
 - ログクリア開始時刻
 - Gateway 自動起動プロセスの最大数
 - ログファイルディレクトリ
 - ジョブ起動オプション
 - 2) ^ZGJOB("CHECKER","pid")に自プロセス ID(\$J)をセットする。
- [メインループ開始/停止確認]
 - 1) ^ZGJOB("SYSTEM","off")を確認し、1であれば終了処理へ。
- [アソシエーション解放]
 - 1) 各DBを担当する全 Gateway プロセスを^ZGJOBより得て、^ZGJOB("J",DBn,pid)をロックする。
 - 2) ^ZGJOB に登録されているPID、プロセス名より該当プロセスが存在するかチェックする。
 - 3) 存在しない場合は、^ZGJOB("J",DBn,PID)、^ZGJOB("F",DBn,PID)を削除し、^ZGJOB("J",DBn)をロック/アンロックし、DB毎のプロセスカウンタ(^ZGJOB("J",DBn))を減算する。
 - 4) 存在する場合で、以下の条件に合致する場合は、ターゲットタイムアウト CHECKER フラグ (^ZGJOB("J",DBn,pid,"chk"))フラグを立て Gateway プロセスを WakeUP する。
 - ・ (現在時刻 (\$ H¹⁰) 最終処理時刻(\$H)) > アソシエーション解放時間
 - ・ 現在処理中でない (フリーリスト^ZGJOB("F",DBn)に登録されている) 場合
 - 5) ロック解除し、次のプロセスの処理へ移る。
- [ログクリア]
 - 1) 当日のシステム日付でログクリアを未実施、かつログクリア開始時刻をすぎた場合、ログクリア処理を開始する。ログクリア開始時刻が 999999 の場合は、ログクリアを禁止する。
 - 2) 削除対象システム日 = 現システム日 (+\$H) ログ保存期間 (日) について、それ以前のシステム日をもつ以下各グローバルについて削除を行う。グローバルを削除する前に、各グローバル(^ZGRD、^ZGQRY、^ZGLOG) をロックし、削除後アンロックする。

^ZGRD("M",RID)	削除対象システム日 (以前) の最大RID まで削除
^ZGRD("C",+\$H)	削除対象システム日以前
^ZGQRY("M",QID)	削除対象システム日 (以前) の最大QID まで削除
^ZGQRY("C",+\$H)	削除対象システム日以前
^ZGQRY("Q", DBno, ステータス,QID)	削除対象システム日 (以前) の最大QID まで削除
^ZGLOG(+\$H)	削除対象システム日以前
^ZGJOB("CHECKER","log",+\$H)	削除対象システム日以前

¹⁰ システム日付が2日にまたがる場合もある。

- 3) ロックタイムアウトせず、全グローバルについて完了した場合は、ログクリア完了フラグ
(^ZGJOB("CHECKER","log",+SH)) をセットする。

- [自動起動 Gateway 確認]
 - 1) ^ZGSYS をロック/アンロックし、自動起動 Yes のターゲット DB 番号リストを得る。
 - 2) 自動起動ターゲットの現在起動済 Gateway プロセス数を計算する。
 - 3) 自動起動プロセス合計が、自動起動 Gateway プロセスの最大値を越えてなければ、Gateway プロセスを各 DB 毎の最大プロセス数とクライアント要求数の小さい数だけ起動する。

ジョブ起動時のパラメータ

プロセス名 : ZG_DBn_Jobno
 ログファイル名 : log directory _ "プロセス名".log
 DSM オプション : ジョブ起動オプションの値(/SOURCE=40000/SYMBOL=6000000)
 パラメータ : DBn_^"_Jobno

- [Sleep]
 - 1) CHECKER プロセス SLEEP 時間だけ HANG し、メインループに戻る。
- [停止]
 - 1) ^ZGJOB("CHECKER","pid")を削除し、HALT する。
- [エラー終了]
 - 1) ^ZGJOB("CHECKER","pid")を削除する。
 - 2) エラーメッセージを標準出力に出力する。
 - 3) プロセス名を ZGE_SJ に変更し、Checker プロセスを再起動する。
 - 4) HALT する。

4.2 開始

- [初期処理]
 - 1) ^ZGSYS をロックし¹¹、^ZGSYS から Gateway/Checker プロセス用 dir、Gateway プロセスのジョブ起動オプションを得る。
 - 2) ロックでタイムアウトが発生した場合は、メッセージを出力し終了する。
- [システム停止フラグクリア]
 - 1) システム停止フラグ(^ZGJOB("SYSTEM","off")を削除し、0 をセットする。
- [Gateway プロセス起動]
 - 1) ターゲット DBn 毎に Gateway プロセス数/自動起動フラグを得て、自動起動以外は、^ZGJOB("J",DBn,pid)にエントリされているプロセス(プロセス名)が存在しないか%GETJPI で確認し、存在しなければ起動する。

Gateway プロセス

プロセス名 : "ZG_"_DBn_"_"_Jobno
 ログファイル名 : log directory _ "プロセス名".log
 DSM オプション : ジョブ起動オプションの値 (/SOURCE=40000/SYMBOL=6000000)
 パラメータ : DBn_^"_Jobno

n は、^ZGSYS のターゲットの DB 番号を示す。Jobno は、ターゲット DB 毎の Gateway プロセスの通番。

- [Checker プロセス起動]
 - 1) Checker プロセス (プロセス名) が存在しないか%GETJPI で確認し、起動する。

Checker プロセス

プロセス名 : "ZG_CHECKER"

ログファイル名 : log directory _ "ZG_CHECKER".log

DSM オプション : ジョブ起動オプション (/SOURCE=40000/SYMBOL=6000000)

- [終了]
 - 1) 起動後、開始時刻をセット(^ZGJOB("SYSTEM","start")する。
 - 2) ^ZGSYS をアンロックする。

4.3 停止

- [初期処理]
 - 1) ^ZGSYS をロックし、^ZGSYS からログファイル保存期間、ログクリア開始時刻、Gateway/Checker プロセス用 dir を得る。
 - 2) ロックでタイムアウトが発生した場合は、メッセージを出力し終了する。
- [プロセス停止]
 - 1) ^ZGJOB をロック/アンロックし、^ZGJOB("J",DBn,pid)および^ZGJOB("CHECKER","pid")の各プロセスリストを得る。
 - 2) この時、この pid に対応するプロセスが存在しないか、プロセス名が異なる場合は、該当プロセスが異常終了したものとし、^ZGJOB("J",DBn,pid),^ZGJOB("F",DBn,pid)を削除し、DB 毎のプロセスカウンタ (^ZGJOB("J",DBn)) を減算する。
 - 3) ^ZGJOB ("SYSTEM","off") に 1 をセットする。
 - 4) Gateway プロセスリストに存在プロセス ID が、^ZGJOB("F",DBn) フリーリストに存在する場合、Wakeup する。全 Gateway が停止し、^ZGJOB("J",DBn,pid)が削除されるまで監視する。
 - 5) Checker プロセスが存在する場合、Checker プロセスが消え、^ZGJOB("CHECKER","pid")が削除されるまで監視する。
- [要求キューステータスクリア]
 - 1) ^ZGQRY("Q",DBn,ステータス)の各DB毎のステータス処理中(1)の QID について ^ZGQRY("M",QID,"db",DBn,"sts")を異常終了 (-1) に変更し、^ZGQRY("Q",DBn,ステータス,QID)を削除する。
- [ログクリア]
 - 1) ログクリア開始時刻が 999999 でなければ (禁止でなければ)、CHECKER プロセスのグローバルのログクリア処理を呼び出す。
 - 2) プロセス用 DIR に存在する ZG_DBn_jobno.LOG;*と ZG_CHECKER.LOG;*、Z3950ECALL_*.LOG.*についてログ保存期間以前の全ファイル (バージョン) について削除を行う。
 - 3) 現在の DIR に存在する CATP*.LOG.*についてログ保存期間以前の全ファイル (バージョン) について削除を行う。
- [パラメータファイルクリア]
 - 1) 各ターゲットDBについて Z39.50 クライアントパラメータファイル名を ^ZGSYS("TARGET",DBn,"pfile")から得て、ファイルが存在すれば削除する。
- [終了]
 - 1) ^ZGJOB("SYSTEM","stop")に停止時刻(\$H)をセットし、^ZGSYS をアンロックする。

4.4 Gateway プロセス

- [初期処理]
 - 1) ジョブ起動パラメータよりターゲットの DBno と Job 番号を得る。
 - 2) ^ZGSYS("CONFIG")から以下のパラメータを得る。
 - アソシエーション解放間隔
 - ログディレクトリ
 - ジョブ起動オプション
 - 自動起動 Gateway プロセス用 CHECKER WAKEUP 回数
 - 3) ^ZGSYS("TARGET",DBno)からの以下 DB 情報を得る
 - DB ID
 - アドレス
 - ポート番号
 - ユーザ名
 - パスワード
 - name result set サポートフラグ
 - 自動起動フラグ
 - cat-p/Z39.50 識別フラグ
 - エンコーディング
 - Preferd record syntax
 - Option record syntax
 - 4) ターゲット DB が Z39.50 で、Z39.50 パラメータファイルを作成する。
 まず、論理名 Z3950INI_DIR に現在の装置 + ディレクトリをアサインする。
 パラメータファイル名をターゲットアドレスから、例えば A.B.C.D なら A_B_C_D.INI とし作成する。
 ファイルには、prefrecsyn= ターゲット recsyn の 1 レコードのみ記述する。¹²
 SUTRS の場合、SIMPLETEXT に変更する、MARC21 の場合は USMARC に変更して prefrecsyn を定義する。
 - 5) ^ZGJOB("J",DBno,pid)をロック/アンロックし、初期値を設定する。
 - ^ZGJOB("J",DBno,pid)の値
 - "prn" = プロセス名
 - "jobno" = Job 番号
 - "sts" = 0(未処理)
 - "chk" = 0
 - "lst" = 初期処理時の SH
 - "start" = 自動起動フラグ
 - "cat-p" = cat-p/Z39.50 識別フラグ
 - "nrs" = name result set サポートフラグ
 - "req" = NULL
 - "qid" = 0
 - "pro" = DB ID^アドレス^ポート番号^ユーザ名^パスワード
 - "clients" = 0
 - 6) ^ZGJOB("J",DBno)をロック/アンロックし、DB 毎のプロセス数をカウントアップする。
 - 7) アソシエーション解放回数をクリアする。(自動起動 Gateway プロセス用)
- [メインループ]

¹² 同一アドレスで、複数の DB をサポートしており各々の record syntax が異なる場合、後から起動された Gateway プロセスによりパラメータファイルが上書きされる状態が発生する。この場合は、local host にターゲットアドレスに対応して、各 DB 毎の別名を登録し、その値を DB 登録することで、別名.INI ファイルが作成され、それで recsyn を区別することで解決する。

- [停止確認]
 - 1) 自動起動 Gateway プロセスの場合は、アソシエーション解放回数が CHECKER WAKEUP 回数と等しくなった場合、停止処理へ移動。
 - 2) システム停止フラグ (^ZGJOB("SYSTEM","off")) が立っている場合、停止処理へ移動。
 - 3) 個別DB毎システム停止フラグ (^ZGJOB("SYSTEM","off",DBno)) が立っている場合、停止処理へ移動。
- [処理要求獲得]
 - 1) ^ZGJOB("F",DBn)をロックする。
 - 2) ^ZGQRY("Q",DBn)をロックし、Client 要求キュー(^ZGQRY("Q",DBn,0)に QID が存在し、Gateway プロセスの pid 値(^ZGQRY("Q",DBn,0,QID))が NULL または自 PID の場合にはその QID 獲得する。¹³client 要求キューの QID ステータスを処理中に変更する。))
 - 3) Client 要求キューをアンロックする。
- [ハイバネート処理/要求が存在しない場合]
 - 1) フリープロセスキュー (^ZGJOB("F",DBn,PID)) に自分を登録する。
 - 2) ^ZGJOB("J",DBn,PID)をロック/アンロックし、^ZGJOB("J",DBn,PID,"sts")に処理ステータス未処理(0)をセットする。
 - 3) ^ZGJOB("F",DBn)をアンロックする。
 - 4) %HIBER する。
 - 5) WAKEUP 後、^ZGJOB("J",DBn,PID)をロック/アンロックし、アソシエーション解放指示がないか、^ZGJOB("J",DBn,PID,"chk")をチェックする。1 なら「アソシエーション解放処理」コール
- [要求処理/要求が存在する場合]
 - 1) フリープロセスキュー(^ZGJOB("F",DBn,PID)より自分を削除する。
 - 2) ^ZGQRY("M",QID,"db",DBn)をロック/アンロックし、QID 処理ステータスに 1 (処理中)をセットする。
^ZGQRY("M",QID,"db",DBn,"sts")=1
^ZGQRY("M",QID,"db",DBn,"sts",コマンド番号)=1
 - 3) ^ZGJOB("J",DBn,PID)をロックし、以下の値をセットする。
 - ・ 処理ステータス(^ZGJOB("J",DBn,PID,"sts")に 1 をセット
 - ・ 現在処理中 QID (^ZGJOB("J",DBn,pid,"req"))に獲得 QID をセット
 - ・ 最終処理時刻 (^ZGJOB("J",DBn,PID,"lst")) に\$H をセットする。
 - 4) 以下初期パラメータ情報を REQPAR 変数にセットする。

REQPAR 添え字	値
"addr"	ターゲットDBのアドレス
"port"	ターゲットDBのポート番号
"user"	ターゲットDBのユーザ名
"pwd"	ターゲットDBのパスワード
"nrss"	ターゲットDBの named result set サポートフラグ
"cat-p"	ターゲットDBの CAT-P/Z39.50 識別フラグ
"dbname"	ターゲットDBの DB ID
"qid"	獲得 QID
"handle"	クライアントハンドル^ZGQRY("M",QID,"handle")の値
"cmdno"	ターゲットDBに対するコマンド番号^ZGQRY("M",QID,"db",DBn,"cmdno")の値
"refid"	リファレンス ID^ZGQRY("M",QID,"db",DBn,"refid")
"rid"	result set id ^ZGQRY("M",QID,"db",DBn,"rid")
"stime"	要求開始時刻(\$H)
"encode"	ターゲットDBのコード
"prefrecsyn"	ターゲットDBがサポートしている record syntax
"optrecsyn"	アプリケーション出力 時特別レコード編集形式

¹³ 同一ターゲットDBに対して複数の Gateway プロセスが起動されている場合は、最初にアソシエーションし、Search を行った Gateway プロセスが以後のそのクエリに対する複数回の PRESENT を担当する。

- 5) リファレンス ID が NULL の場合は、以下規則で獲得後、クライアントと Gateway プロセスの対応を識別するための情報を ^ZGJOB、^^ZGQRY にセットする。

CAT-P/Z39.50 フラグ	named result set サポート	reference-id
Z39.50	有	Gateway Process の PID
	無	client handle
CAT-P		client handle

^ZGQRY("M",QID,"db",DBn,"pid")=PID
 ^ZGQRY("M",QID,"db",DBn,"refid")=獲得リファレンス ID
 クライアントハンドルが、^ZGJOB("J",DBn,PID,"clients",クライアントハンドル)に存在しなければ、担当クライアント数 (^ZGJOB("J",DBn,PID,"clients")) を加算し、
 ^ZGJOB("J",DBn,PID,クライアントハンドル)を登録する。

- 6) ^ZGJOB("J",DBn)、^ZGJOB("F",DBn)をアンロックする。¹⁴

- 7) コマンド毎の情報を REQPAR 変数にセットする。

REQPAR 添え字	値
"query"	検索式 SEARCH/PRESENT の場合 ^ZGQRY("M",QID,"db",DBn,"query")
"fromto"	レコード獲得範囲 Start-end PRESENT の場合 ^ZGQRY("M",QID,"db",DBn,"fromto")

- 8) ^ZGRD("M",RID,"sts")を処理中(1)にセットする。

• [INIT 処理確認]

- 1) アソシエーション済みフラグをセット(1)する。今回 INIT フラグをクリア(0)する。
- 2) コマンド番号が 0 の場合は、以下の処理を行い [Result set name 処理]へ移る。
 ^ZGJOB("J",DBn,PID,"refid",refid,handle)が存在しない場合、アソシエーション済みフラグをクリア(0)する。
- 3) ^ZGJOB("J",DBn,PID,"refid",refid)に REQPAR("refid")が登録されていない場合は、今回 INIT フラグに 1 をセットする。
- 4) Z39.50 で named result set 未サポートハンドルリスト
 (^ZGJOB("J",DBn,PID,"defaultsrn",ハンドル)) に handle が存在する場合、今回 INIT フラグを 1 にセットする。¹⁵
- 5) CAT-P の場合 [Result set name 獲得]の処理へ移る
- 6) 「Z39.50 INIT 処理」を呼び出す。INIT 処理失敗で、アソシエーション済みフラグをクリア(0)する。
- 7) INIT の named result set support 戻り値より ^ZGSYS の named result set 情報と異なる場合、REQPAR("nrss")を変更し、さらに以下のグローバルを更新する。
 ^ZGSYS("TARGET",DBn,"nrs") = 戻り値
 ^ZGJOB("J",DBn,PID,"nrs")= 戻り値
- 8) CAT-P または Z39.50 で INIT 完了した場合 ^ZGJOB("J",DBn,PID,"refid",refid)に REQPAR("refid")、クライアントハンドルを登録する。
 ^ZGJOB("J",DBn,PID,"refid",refid,handle)=""

• [Result set name 獲得]

- 1) ^ZGRD より、Result set name を REQPAR 変数にセットする。
- 2) result set name が NULL の場合は、以下の規則で作成する。

¹⁴ PRESENT は前回実行 Gateway が担当することやプロセスフリーキューによるプロセスの割当てなどのため、フリーキューの情報と、担当クライアント情報、要求キューの情報は、ロックにより排他制御し一体で更新しなければならない。

¹⁵ 前回実行 SEARCH/PRESENT QID と異なる QID の PRESENT 要求が来た場合に再度 SEARCH 要求を実行させるため。

REQPAR 添え字	値	
"rsn"	result set name ^ZGRD("M",RID,"rsn")	
CAT-P/Z39.50 フラグ	named result set サポート	result set name
Z39.50	有	query-id
	無	"default"
CAT-P		query-id

- [CLOSE 処理]
 コマンド番号が0の場合にはZ39.50/CAT-P 共通の CLOSE 処理を行う。
 1) ^ZGJOB("J",DBn,PID)をロックする。
 2) ^ZGJOB("J",DBn,PID,"refid",refid,handle)を削除する。
 3) ^ZGJOB("J",DBn,PID,"refid",refid,handle)を削除する。
 4) named result set 未サポートの場合、未サポートハンドルのリスト (^ZGJOB("J",DBn,PID,"default",handle)) を削除する。
 5) ^ZGJOB("J",DBn,PID,"clients",handle)が存在すれば、 ^ZGJOB("J",DBn,PID,"clients",handle)を削除し、クライアント担当数(^ZGJOB("J",DBn,PID))を減算する。
 6) ^ZGJOB("J",DBn,PID)をアンロックする。
- [Z39.50 SEARCH/PRESENT/CLOSE 処理呼び出し]
 1) アソシエーション済みでなければ、[コマンド結果 RID 更新処理]へ移る。
 2) CLOSE の場合で ^ZGJOB("J",DBn,PID,"refid",refid)以下に他のクライアントハンドルが存在しなければ、「Z39.50 用 CLOSE 処理」を呼び出す。
 3) SEARCH の場合「Z39.50 用 SEARCH 処理」を呼び出す。
 4) PRESENT、今回 INIT フラグ = 1 の場合、「Z39.50 SEARCH 処理」、「コマンド結果 RID 更新」、「Z39.50 PRESENT 処理」を連続して呼び出す。¹⁶
 SEARCH 処理を呼び出す前後で REQPAR("cmdno")の値を 1 - > 2 へ変更する。
 SEARCH 処理、コマンド結果 RID 更新後、REQPAR("stime")に現在日時(SH)でリセットする。
 5) PRESENT、今回 INIT フラグ = 0 の場合、「Z39.50 PRESENT 処理」を呼び出す。
- [CAT-P SEARCH/PRESENT/CLOSE 処理呼び出し]
 1) アソシエーション済みでなければ、[コマンド結果 RID 更新処理]へ移る。
 2) CLOSE の場合で ^ZGJOB("J",DBn,PID,"refid",refid)以下に他のクライアントハンドルが存在しなければ、「CAT-P 用 CLOSE 処理」を呼び出す。
 3) SEARCH の場合「CAT-P 用 SEARCH 処理」を呼び出す。
 4) PRESENT、今回 INIT フラグ = 1 の場合、「CAT-P SEARCH 処理」、「コマンド結果 RID 更新」、「CAT-P PRESENT 処理」を連続して呼び出す。
 SEARCH 処理を呼び出す前後で REQPAR("cmdno")の値を 1 - > 2 へ変更する。
 SEARCH 処理、コマンド結果 RID 更新後、REQPAR("stime")に現在日時(SH)でリセットする。
 5) PRESENT、今回 INIT フラグ = 0 の場合、「CAT-P PRESENT 処理」を呼び出す。
- [コマンド失敗による再アソシエーション準備]
 Z39.50 で SEARCH/PRESENT コマンドが失敗した場合、再度 INIT 処理を行うため、以下の処理を行う。
 1) REQPAR("refid")を指定してアソシエーションを解放する。CLOSE^%Z3COM
 2) リファレンス ID に対するアソシエーションを実行済みを示すエントリを削除する。
 ・ ^ZGJOB("J",DBn,PID,"refid",リファレンス ID)を削除
 ・ ^ZGQRY("M",QID,"db",DBn,"refid") = NULL セット

¹⁶ アソシエーション解放間隔のタイムアウトにより解放されている場合を考慮し、再度 SEARCH した後で、PRESENT を行う。

- [コマンド結果 RD 更新]
 - 1) RID に対応する ^ZGRD("M",RID)をロックする。
 - 2) Z39.50 の場合、%Z3RES("z")、%Z3RES("a")、%Z3RES("s")の内容を、
^ZGRD("M",RID,コマンド番号,"z"),^(("a")、^(("s")、にコピーする。
 - 3) CAT-P の場合、%CATPsts("c")、%CATPsts("a")の内容を、
^ZGRD("M",RID,コマンド番号,"c"),^(("a")にコピーする。
 - 4) PRESENT の場合、上記以外に Z39.50 の場合は、%Z3RES("r")の内容を、CAT-P の場合は、
%CATPsts("r")の内容を ^ZGRD("M",RID,コマンド番号,"r")にレコード No を考慮してコピーする。
例)
rc = fromto 開始 no - 1
i = 1:1:%Z3RES("numberofRecordsReturned") D
.S rc = rc + 1 M ^ZGRD("M",RID,"r",2,rc)=%Z3RES("r",refid,i)

rc = fromto 開始 no - 1
i = 1:1:%CATPsts("Number-of-records-returned") D
.S rc = rc + 1 M ^ZGRD("M",RID,"r",2,rc)=%CATPsts("r",i)
 - 5) rc の値が、^ZGRD("M",RID,2,"r")より大きければ
 - ^ZGRD("M",RID,2,"r")=rc
 - ^ZGQRY("M",QID,"db",DBn,"get")=rc
 - ^ZGQRY("M",QID,"db",DBn,"next")=rc+1
 - 6) result set name と、処理完了ステータス(2)を、^ZGRD("M",RID,"rsn")、^ZGRD("M",RID,"sts")
にセットする。
 - 7) ^ZGRD("M",RID,コマンド番号,"time")=REQPAR("stime")_ "^_要求終了時刻(\$H)
 - 8) ^ZGRD("M",RID)をアンロックする。
- [コマンド結果 QID 更新]
 - 1) ^ZGQRY("M",QID,"db",DBn)をロックする。
 - 2) QID ステータスを完了(2)にセットし、
^ZGQRY("M",QID,"db",DBn,"sts")=2
^ZGQRY("M",QID,"db",DBn,"sts",コマンド番号)=2
要求キューファイル(^ZGQRY("Q",DBn,ステータス,QID))を更新する。
 - 3) ^ZGQRY("M",QID,"db",DBn)をアンロックする。
- [終了処理]
 - 1) ^ZGJOB("J",DBn,PID)をロックする。
 - 2) named result set 未帰 帰の場合で、返す 番号が 1 又は 2 の場合
^ZGJOB("J",DBn,PID,"defaultrsn",handle)をセットする。
 - 3) 処理済 QID を ^ZGJOB("J",DBn,PID,"QID",QID)に登録し、処理済数(^ZGJOB("J",DBn,PID,"QID"))を
更新する。
 - 4) プロセス処理ステータスを未処理(0)を ZGJOB("J",DBn,PID,"sts")にセットする。
 - 5) 処理中 QID(^ZGJOB("J",DBn,PID,"req"))を NULL クリアする。
 - 6) 最終処理時刻(^ZGJOB("J",DBn,PID,"lst"))に現在時刻(\$H)をセットする。
 - 7) ^ZGJOB("J",DBn,PID)をアンロックする。
 - 8) メインループへ
- [停止]
 - 1) 「アソシエーション解放処理」をコールする。
 - 2) ^ZGJOB("F",DBn)をロック/アンロックし、^ZGJOB("F",DBn,PID)を削除する。
 - 3) ^ZGJOB("J",DBn)をロック/アンロックし、^ZGJOB("J",DBn,PID)を削除し、DB 毎プロセス
数(^ZGJOB("J",DBn))を減算する。
 - 4) HALT する。

- [Z39.50 INIT 処理]
 - 1) REQPAR 変数から以下の入力パラメータにより%Z3COM ルーチンをコールする。
Target addr,port,username,password,reference-id
D INIT^%Z3COM(refernce-id,addr,port,username,password)
- [アソシエーション解放処理]
 - 1) アソシエーション解放回数をカウントアップする。
 - 2) ^ZGJOB("J",DBn,PID,"refid")以下にリファレンス ID が登録されていれば、ターゲットが CAT-P/Z39.50 により、各プロトコルの CLOSEALL 処理をコール。
Z39.50 の場合：CLOSEALL^%Z3COM
CAT-P の場合：CLOSEALL^%CATP
 - 3) ^ZGJOB("J",DBn,PID)をロックする。
 - 4) ^ZGJOB("J",DBn,PID,"qid")に登録されている全 QID について、^ZGQRY("M",QID,"db",DBn)をロック/アンロックし、処理 Gateway プロセス ID(^ZGQRY("M",QID,"db",DBn,"pid"))およびリファレンス ID(^ZGQRY("M",QID,"db",DBn,"refid"))を NULL クリアする。
 - 5) 処理済 QID(^ZGJOB("J",DBn,PID,"qid"),QID)を削除する。
 - 6) 担当クライアント (^ZGJOB("J",DBn,PID,"clients"))を削除する。
 - 7) アソシエーション済みリファレンス ID (^ZGJOB("J",DBn,PID,"refid"))を削除する。
 - 8) named result set 未サポートハンドリスト (^ZGJOB("J",DBn,PID,"defaultsrn"))を削除する。
 - 9) 処理中 QID(^ZGJOB("J",DBn,PID,"req"))を NULL クリアする。
 - 10) アソシエーション指示フラグ(^ZGJOB("J",DBn,PID,"chk"))を 0 クリアする。
 - 11) ^ZGJOB("J",DBn,PID)をアンロックする。
- [Z39.50 CLOSE 処理]
 - 1) 入力パラメータ refernce-id
 - 2) %Z3COM 呼び出しを行う。CLOSE^%Z3COM(refid)
 - 3) %Z3RES("a")が 0 以外の場合、「エラーメッセージ出力」を行い戻る。
- [CAT-P CLOSE 処理]
 - 1) 入力パラメータ refernce-id
 - 2) %CATP 呼び出しを行う。CLOSE^%CATP(refid)
 - 3) %CATPsts("a")が 0 以外の場合、「エラーメッセージ出力」を行い戻る。
- [Z39.50 SEARCH 処理]
 - 1) 入力パラメータ query 文,refernce-id,dbname、result set name、encode
 - 2) query 文を encode によりコード変換する。(EnCode^ZGCCNV)
 - 3) %Z3COM 呼び出しを行う。SEARCH^%Z3COM(refid,DB-id,query,rsn)
 - 4) %Z3RES("a")が 0 以外の場合、「エラーメッセージ出力」を行い戻る。
 - 5) 正常終了の場合、%Z3RES("z","resultCount")の値を数値化 (+ 指定) し戻る。
- [CAT-P SEARCH 処理]
 - 1) 入力パラメータ username,password,query 文,refernce-id,dbname,result set name,encode
 - 2) query 文を encode によりコード変換する。(EnCode^ZGCCNV)
 - 3) 以下の%CATP 呼び出しを行う。
CLOSE^%CATP(refid)
OPEN^%CATP(refid,userid,password,"","")
SEARCH^%CATP(refid,db-id,query,rsn)
 - 4) %CATPsts("a")が 0 以外の場合、「エラーメッセージ出力」を行い戻る。
 - 5) 正常終了の場合、%CATPsts("c","resultCount")の値を数値化 (+ 指定) し戻る。

- [Z39.50 PRESENT 処理]
 - 1) 入力パラメータ fromto,reference-id,result set name
 - 2) 以下の%Z3COM¹⁷呼び出しを行う。
PRESENT^%Z3COM(refid,fromto,rsn,"","")
 - 3) %Z3RES("a")が0以外の場合、「エラーメッセージ出力」を行いつつ戻る。

- [CAT-P PRESENT 処理]
 - 1) 入力パラメータ fromto,reference-id,result set name
 - 2) レコード範囲指定から、開始レコードと取得レコード数を計算する。
 - 3) 以下の%CATP 呼び出しを行う。
PRESENT^%CATP(refid,rsn,start record,record num,"F","")
 - 4) %CATPsts("a")が0以外の場合、「エラーメッセージ出力」を行いつつ戻る。

- [エラー処理]
 - 1) エラーメッセージを標準出力に出力する。
 - 2) 「アソシエーション解放処理」をコールする。
 - 3) ^ZGJOB("F",DBn)をロック/アンロックし、^ZGJOB("F",DBn,PID)を削除する。
 - 4) ^ZGJOB("J",DBn)をロック/アンロックし、^ZGJOB("J",DBn,PID)を削除し、DB 毎プロセス数(^ZGJOB("J",DBn))を減算する。
 - 5) プロセス名をZGE_SJに変更し、旧プロセス名に対応するDBn, Jobno の Gateway プロセスを起動する。
 - 6) HALT する。

- [エラーメッセージ出力]
 - 1) Z39.50 の場合、標準出力に%Z3RES 変数を ZWRITE する。
 - 2) CAT-P の場合、標準出力に%CATPsts 変数を ZWRITE する。

¹⁷ ZGORGD より Call される Z39.50 及び CAT-P クライアント API プログラムについては、各 API のユーザーズガイドを参照してください。

4.5 Gateway I/F

4.5.1 初期処理/ハンドルの確保(INIT^ZGSND)

Calling Seq.

S STS=\$\$INIT^ZGSND(,%CGIEVAR,,DAY,,HANDLE)

Parameters

Input

CGIVAR %CGIEVAR (WEBLINK var)

Output

DAY クライアント・ログインシステム日 (+\$H)

HANDLE クライアント・ハンドル

戻り値

STS ステータス
 0: 成功
 -5: リトライオーバー
 -4: 異常終了

- [ハンドル獲得/^ZGLOG 初期化]
 - 1) ^ZGSYS("CONFIG","login","retry")よりログインハンドル獲得リトライ回数を得る。
 - 2) クライアントハンドルを以下の式で計算する。
 $handle = SP(SH, ",", 2) * 100 + SR(100)$
 - 3) ^ZGLOG(+\$H,handle)をロックし、^ZGLOG(+\$H,handle)が存在すれば、アンロックし0.5秒 HANGして再度ハンドルを計算してトライする。
 - 4) トライ数が、ログインハンドル獲得リトライ回数を越えた場合、ステータスにリトライオーバーのエラーをセットし、戻る。
 - 5) ^ZGLOG(+\$H,handle,"in-time")に現在のシステム時刻(\$H)をセットする。
 - 6) ^ZGLOG(+\$H,handle,"client")に%CGIEVAR
 の"REMOTE_ADDRESS","REMOTE_HOST","REMOTE_IDENT","REMOTE_USER"の値をセットする。
 - 7) ^ZGLOG をアンロックし、ステータスに正常終了をセットし、戻る。

4.5.2 SEARCH 要求(SRCQ^ZGSND)

Calling Seq.

S STS=\$\$SRCQ^ZGSND(DAY,HANDLE,,DBLIS,,QUERY,,QID)

Parameters

Input

DAY クライアント・ログインシステム日 (+\$H)

HANDLE クライアント・ハンドル

DBLIS DBLIS= 選択件数
 DBLIS(i) = DB 番号

QUERY QUERY(DBno) = 行数
 QUERY(DBno,n) = 検索式

Output

QID Query id

戻り値

STS ステータス
 0: 成功
 -1: ターゲット情報なし
 -2: Z39.50/CAT-P 用検索式なし (パラメータエラー)
 -3: ロックタイムアウト
 -4: 異常終了
 -7: CLOSE 処理済み

- [パラメータチェック]
 - 1) DBn より ^ZGSYS("TARGET")より cat-p/Z39.50 フラグを得て cat-p/Z39.50 毎 DB 数をカウントする。
 - 2) cat-p/Z39.50 毎 DB 数より query 文に各プロトコル用の検索式がセットされているかチェックする。
 - 3) ^ZGLOG(day,handle,"cls-time")が存在する場合、エラーとして戻る。
 - 4) cat-p/Z39.50 毎 DB 数が共に 0 の場合や、その他パラメータチェックエラーの場合は、ステータスをエラーに更新し戻る。

- [QID 獲得/^ZGQRY 初期化]
 - 1) ^ZGQRY("C")をロック/アンロックし、その間に QID を採番し、^ZGQRY("C",+SH)をその QID で更新する。
 - 2) ^ZGQRY("M",QID)の初期化を行う。


```

^ZGQRY("M",QID,"handle")=client handle
"day")=login day
"queryflag")=ターゲット queryflag (クライアント検索禁止フラグ)
"db")=DB 数
^ZGQRY("M",QID,"db",DBn,"pro")=^ZGSYS("TARGET",DBn)のターゲット接続情報
^ZGQRY("M",QID,"db",DBn,"sts")=0 (要求中)
^ZGQRY("M",QID,"db",DBn,"sts",1)=0
^ZGQRY("M",QID,"db",DBn,"query")=行数
^ZGQRY("M",QID,"db",DBn,"query",n)=検索式 (逆ポーランド形式)
^ZGQRY("M",QID,"db",DBn,"cmdno")=1
          
```

- [RID 獲得/^ZGRD 初期化]
 - 1) ^ZGRD("C")をロック/アンロックし、その間にターゲット DB 数分の result-id を獲得する。
 - 2) ^ZGRD("M",RID)の初期化を行う。


```

^ZGRID("M",RID,"sts")=0
"qid")=QID
"time")=SH
"cat-p")=ターゲットの cat-p/Z39.50 フラグ値
"db")=DB 番号
"query")=Z39.50/CAT-P いずれかの query 文
^ZGQRY("M",QID,"db",DBn,"rid")=RID
          
```

- [フリープロセス確保]
 - 1) DB リストに存在する各 DB 番号について順に担当 Gateway プロセスを確保する。
 - 2) ^ZGJOB("F",DBn)、^ZGQRY("Q",DBn)をロックする
 - 3) ^ZGJOB("F",DBn)に登録されている Gateway プロセスの PID を順に次のチェックを行う。
 - DB 毎の担当クライアント最大数が 0 (無制限)で、named result set 有の場合
登録されている最初の PID を担当プロセスとする。
 - DB 毎の担当クライアント最大数が 0 (無制限)で、named result set 無の場合
クライアント最大数を 3 2 として最大数が設定されている場合の処理を行う。
 - DB 毎の担当クライアント最大数が設定されている場合は
^ZGJOB("J",DBn,PID)をロック/アンロックし、
^ZGJOB("J",DBn,PID,"clients")より現在担当クライアント数とその値以下の場合や、
パラメータのハンドルが既存担当リスト (^ZGJOB("J",DBn,PID,"clients",
client handle))に登録されている場合は、そのプロセスとする。
 - 4) PID が確保できれば、^ZGQRY("Q",DBn,0,QID)=PID とセットし、そのプロセスを%WAKE し、
^ZGJOB("F",DBn,PID)エンタリを削除する。
 - 5) PID 存在しなければ、^ZGQRY("Q",DBn,0,QID)=""をセットする。
 - 6) ^ZGQRY("Q",DBn),^ZGJOB("F",DBn)をアンロックする。

- [ログ更新終了]
 - 1) ^ZGLOG(+Sh,ハンドル,"out-time")にシステム時刻(SH)をセットする。
 - 2) ^ZGLOG(+Sh,ハンドル,"cur")に獲得した qid をセットする。
 - 3) ステータスに正常終了をセットし戻る。

4.5.3 SEARCH 応答(SRCR^ZGSND)

Calling Seq.

S STS=\$\$SRCR^ZGSND(DAY,HANDLE,QID,.DBSTS)

Parameters

Input

DAY クライアント・ログインシステム日 (+\$H)

HANDLE クライアント・ハンドル

QID Query id

Output

DBSTS DB 毎のステータス報告

DBSTS(DB 番号) = ステータス^ヒット件数^メッセージ

0: 完了(エラーなし)

1: 完了(エラー完了)

2: 未完了

戻り値

STS ステータス

0: 成功

1: 部分完了

2: 未完了

-2: パラメータエラー

-3: ロックタイムアウト

-4: 異常終了

-7: CLOSE 済み

• [パラメータチェック]

- 1) QID が ^ZGLOG(DAY,HANDLE,"cur") と一致するか、 ^ZGLOG(DAY,HANDLE,"qid",QID) に登録されているかをチェックする。どちらでもない場合は、ステータスをパラメータエラーとし戻る。
- 2) ^ZGLOG(day,handle,"cls-time") が存在する場合、エラーとして戻る。

• [処理ステータス獲得]

- 1) ^ZGQRY("M",QID) をロックし、 ^ZGQRY("M",QID,"db") に登録されている各 DB 番号毎に処理を行う。
- 2) 各 DB の Result set ID を得て、コマンドステータス(^ZGRD("M",RID,1,"a")) を得る。
- 3) 各 DB 毎・コマンド毎のステータスを得る。(^ZGQRY("M",QID,"db",DBn,"sts",1))
- 4) ヒット件数、メッセージを得る。

Z39.50 の場合

ヒット件数 ^ZGRD("M",RID,1,"z","resultCount")

メッセージ ^ZGRD("M",RID,1,"z","errorMessage") : ^("errorCode")

メッセージ2 ^ZGRD("M",RID,1,"z","errorMessage")

CAT-P の場合

ヒット件数 ^ZGRD("M",RID,1,"c","resultCount")

メッセージ ^ZGRD("M",RID,1,"a","m")

メッセージ2 ^ZGRD("M",RID,1,"c","errorMessage")

5) DBSTS の作成

各DB毎・コマンド毎のステータス	DBSTS のステータス	ヒット件数	メッセージ
0, 1	2	NULL	NULL
2, 3	0	ヒット件数	コマンドステータス = 0 なら NULL それ以外メッセージ

6) 各DB毎・コマンド毎のステータスが3の場合は、クライアント獲得済みなので次のDB番号へ移る

7) 2の場合以下のグローバルを更新する。

^ZGQRY("M",QID,"db",DBn,"mess") = メッセージ

^("mess2") = メッセージ2

^("hits") = ヒット件数

8) 各DB毎・コマンド毎のステータス、QID キューを2から3へ変更する。

```
^ZGQRY("M",QID,"db",DBn,"sts")=3
```

```
^ZGQRY("M",QID,"db",DBn,"sts",1)=3
```

9) 全DB処理完了後、^ZGQRY("M",QID)をアンロックする。

• [ログ更新終了]

1) ^ZGLOG(+Sh,ハンドル,"out-time")にシステム時刻(SH)をセットする。

2) ^ZGLOG(+Sh,ハンドル,"cur")をクリアし、^ZGLOG(+Sh,ハンドル,"qid",QID)に処理済みQIDをセットする。

3) 正常終了時のステータスを次のようにセットし戻す。

条件	ステータス
検索完了DB数 = 要求DB数	0
要求DB数 > 検索完了DB数 > 0	1
検索完了DB数 = 0	2

4.5.4 PRESENT 要求(PRSQ^ZGSND)

Calling Seq.

```
S STS=$$PRSQ^ZGSND(DAY,HANDLE,FROMTO,RANGE,QID,.DBLIS)
```

Parameters

Input

DAY クライアント・ログインシステム日 (+SH)

HANDLE クライアント・ハンドル

FROMTO -range start-end

RANGE -range

QID Query id

DBLIS -PRESENT する DB 番号 (既に SEARCH したDBリスト内)

DBLIS = 件数

DBLIS(i) = DB 番号

Output

なし

戻り値

STS ステータス

0: 成功

1: 未完了

-2: パラメータエラー

-3: ロックタイムアウト

-4: 異常終了

-6: SEARCH 実行プロセスが存在せず

-7: CLOSE 済み

• [パラメータチェック]

1) レコード範囲指定の存在、開始・終了の大小チェック、QIDが^ZGLOG(DAY,HANDLE,"qid",QID)に登録されているかチェックし、エラーの場合は戻る。

2) ^ZGLOG(day,handle,"cls-time")が存在する場合、エラーとして戻る。

3) 今回 PRESENT 要求DBについて現在処理中でないかチェックする。パラメータのDBリストについてDB毎にチェックする。

DB 毎ステータス	戻りステータス
< 2	1
DB 毎 SEARCH コマンドステータス	
< 3	1
DB 毎 PRESENT コマンドステータス	
存在し、< 3	1

- 4) 0 以外のステータスの場合は、戻る
- [QID 更新]
 - 1) ^ZGQRY("M",QID)の処理処理
 - ^ZGQRY("M",QID,"sts")=0
 - ^ZGQRY("M",QID,"db",DBn,"cmdno")=2
 - ^ZGQRY("M",QID,"db",DBn,"fromto")=レコード範囲
 - ^ZGQRY("M",QID,"db",DBn,"sts")=0 (要求中)
 - ^ZGQRY("M",QID,"db",DBn,"sts",2)=0
 - 2) ^ZGQRY("M",QID)をアンロックする。
 - [フリープロセス確保]

PRESENT DB リストの各DBについて、担当 Gateway プロセスを確保する。

 - 1) ^ZGJOB("F",DBn)、^ZGQRY("Q",DBn)をロックする。
 - 2) ^ZGQRY("M",QID,"db",DBn,"pid")より前回実行 Gateway PID を得る。
 - 3) そのプロセス ID が Gateway プロセスとして存在しなければ、新たに PID を確保するため、SEARCH 要求時と同様なチェックを行う。(フリープロセスの確保|40ページ)
 - 4) ^ZGQRY("Q",DBn,0,QID)=PID をセットする。
 - 5) Gateway PID が ZGJOB("Q",DBn)のフリープロセスキューに存在すれば、%WAKE し、^ZGJOB("F",DBn,PID)エントリを削除する。
 - 6) ^ZGJOB("F",DBn)、^ZGQRY("Q",DBn)をアンロックする。
 - [ログ更新]
 - 1) ^ZGLOG(+Sh,ハンドル,"out-time")にシステム時刻(SH)をセットする。
 - 2) ^ZGLOG(+Sh,ハンドル,"cur")に QID をセットする。
 - 3) ステータスに正常終了をセットし戻る。

4.5.5 PRESENT 応答(PRSR^ZGSND)

Calling Seq.

S STS=\$\$PRSR^ZGSND(DAY,HANDLE,QID,RNO,.,DBLIS,.,DBSTS)

Parameters

Input

DAY クライアント・ログインシステム日 (+SH)
 HANDLE クライアント・ハンドル
 QID Query id
 RNO 最終獲得レコード番号
 DBLIS -PRESENT した DB 番号 (既に SEARCH したDBリスト内)
 DBLIS = 件数
 DBLIS(i) = DB 番号

Output

DBSTS DB 毎のステータス
 DBSTS(DB 番号) = ステータス^ヒット件数^メッセージ
 0: 完了
 1: 完了(エラー)
 2: 未完了

戻り値

STS ステータス
 0: 成功
 1: 部分完了
 2: 未完了
 -2: パラメータエラー
 -3: ロックタイムアウト
 -4: 異常終了
 -7: CLOSE 済み

- [パラメータチェック]
 - 1) QID が ^ZGLOG(DAY,HANDLE,"qid",QID) に登録されているかをチェックする。登録されていない場合は、ステータスをパラメータエラーとし戻る。
 - 2) ^ZGLOG(day,handle,"cls-time") が存在する場合、エラーとして戻る。

- [処理ステータス獲得]
 - 1) ^ZGQRY("M",QID) をロックし、パラメータの DB リストで指定されている各 DB 番号毎に処理を行う。
 - 2) 各 DB の Result set ID を得て、コマンドステータス (^ZGRD("M",RID,2,"a")) を得る。
 - 3) 各 DB 毎・コマンド毎のステータスを得る。 (^ZGQRY("M",QID,"db",DBn,"sts",2))
 - 4) ヒット件数、メッセージを得る。

[Z39.50 の場合]
 ヒット件数 ^ZGRD("M",RID,1,"z","resultCount")
 メッセージ ^ZGRD("M",RID,2,"a","m")
 メッセージ 2 ^ZGRD("M",RID,2,"z","errorMessage")

[CAT-P の場合]
 ヒット件数 ^ZGRD("M",RID,1,"c","resultCount")
 メッセージ ^ZGRD("M",RID,2,"a","m")
 メッセージ 2 ^ZGRD("M",RID,2,"c","errorMessage")

5) DBSTS の作成

条件	DBSTS ステータス	ヒット件数	メッセージ
各 DB 毎・コマンド毎ステータス = 0 又は 1、 最終レコードが獲得済みでない	2	NULL	NULL
各 DB 毎・コマンド毎ステータス = 3	0	ヒット件数	NULL
各 DB 毎・コマンド毎ステータス = 3 コマンドステータス < 0	1	ヒット件数	メッセージ

- 6) 各 DB 毎・コマンド毎のステータスが 2 以外の場合は、次の DB 番号へ移る
- 7) 2 の場合以下のグローバルを更新する。
 ^ZGQRY("M",QID,"db",DBn,"mess") = メッセージ
 ^("mess2") = メッセージ 2
- 8) 各 DB 毎・コマンド毎のステータス、QID キューを 2 から 3 へ変更する。
 ^ZGQRY("M",QID,"db",DBn,"sts")=3
 ^ZGQRY("M",QID,"db",DBn,"sts",2)=3
- 9) 全 DB 処理完了後、^ZGQRY("M",QID) をアンロックする。

- [ログ更新]
 - 1) ^ZGLOG(+Sh,ハンドル,"out-time") にシステム時刻(SH) をセットする。
 - 2) ^ZGLOG(+Sh,ハンドル,"cur") をクリアし、^ZGLOG(+Sh,ハンドル,"qid",QID) に処理済み QID をセットする。
 - 3) 正常終了時のステータスを次のようにセットし戻る。

条件	ステータス
PRESENT 完了 DB 数 = 要求 DB 数	0
要求 DB 数 > PRESENT 完了 DB 数 > 0	1
PRESENT 完了 DB 数 = 0	2

4.5.6 CLOSE 要求(CLOSE^ZGSND)

Calling Seq.

S STS=\$\$CLOSE^ZGSND(DAY,HANDLE)

Parameters

Input

DAY クライアント・ログインシステム日 (+\$H)

HANDLE クライアント・ハンドル

戻り値

STS ステータス
 0: 成功
 -2: パラメータエラー
 -3: ロックタイムアウト
 -4: 異常終了
 -7: CLOSE 済み

- [パラメータチェック]
 - 1) ^ZGLOG(day,handle,"cls-time")が存在する場合、エラーとして戻る。
- [セッション未解放 DB リスト作成]

過去に要求したクエリについて、タイマーによる解放がなされていないクエリ (リファレンス ID が存在する) についてしらべ対象ターゲット DB Gateway プロセスに CLOSE 要求を送信する。

 - 1) 過去の qid (^ZGLOG(DAY,HANDLE,"qid")以下) について、各 DB の GW プロセス id、とリファレンス id がともに NULL でない場合を対象として、以下の配列を作成する。

qidlis = 送信する要求数

qidlis(QID カウンタ) = 送信 DB 数 ^ QUERY-ID

qidlis(QID カウンタ, DB 番号) = GW プロセス ID ^ リファレンス ID

- 2) 各 DB には複数の GW プロセスが存在し、複数の GW プロセスに要求が発行されている場合がある。より少ない要求で多くの GW プロセスに CLOSE 要求を送信できるように GW プロセスの PID で重複チェックを行う。
- [^ZGQRY, ^ZGRD の初期化]

qidlis について要求数分以下の処理を行う。

 - 1) SEARCH 要求と同様な方法で、QUERY-ID を獲得し、 ^ZGQRY の初期化を行う。


```
^ZGQRY("M",QID,"day")=DAY
^ZGQRY("M",QID,"handle")=HANDLE
```
 - 2) qidlis(i)以下に登録されている DB 数分の Result-ID(RID)を SEARCH 要求と同様な方法で獲得する。
 - 3) 各 DB 番号について ^ZGQTY と ^ZGRD を初期化する。


```
^ZGQRY("M",QID,"db",DBn,"cmdno")=0
^("refid")=リファレンス id
^("rid")=RID
^("sts",0)=0
^ZGRD("M",RID,"qid")=QID
^("sts")=0
^("time")=獲得時刻
^("cat-p")=Z39.50/CAT-P 識別子
```

- [フリープロセス確保]
qidlis について要求数分以下の処理を行う。各 QID 毎の各 DB について、担当 Gateway プロセスを確保する。
 - 1) ^ZGJOB("F",DBn)、^ZGQRY("Q",DBn)をロックする。
 - 2) qidlis(カウンタ,DBn)より、前回実行 Gateway PID を得る。
 - 3) そのプロセス ID が Gateway プロセスとして存在しなければ、次の DB に移る。
 - 4) ^ZGQRY("Q",DBn,0,QID)=PID をセットする。
 - 5) Gateway PID が ZGJOB("Q",DBn)のフリープロセスキューに存在すれば、%WAKE し、
^ZGJOB("F",DBn,PID)エントリを削除する。
 - 6) ^ZGJOB(" F ",DBn)、^ZGQRY("Q",DBn)をアンロックする。

- [ログ更新]
 - 1) ^ZGLOG(+ \$h,ハンドル,"cls-time")にシステム時刻(\$H)をセットする。
 - 3) ステータスを 0 にセットし戻る。

4.6 Gateway 検索システム

4.6.1 メイン

- [初期処理]
 - 1) 以下の変数を初期化する。


```
CRLF=SC(13,10)
%ZGSWAIT=SEARCH 監視時間 ^ZGSYS(CONFIG)より獲得
%ZGSTRY=SEARCH 監視回数 ^ZGSYS(CONFIG)より獲得
%ZGPWAIT=PRESENT 監視時間 ^ZGSYS(CONFIG)より獲得
%ZGPTRY=PRESENT 監視回数 ^ZGSYS(CONFIG)より獲得
%ZGAPP=%KEY(MGWAPP)
%ZGNREC=HOLD 出力件数 ^ZGSYS(CONFIG)より獲得
%ZGDLL=URL_"?MGWLPN=...&MGWAPP=..."
%ZQID=$P(%KEY("ZQIDB"),",",1)
%ZDB=$P(%KEY("ZQIDB"),",",2)
%ZGQID=%KEY("ZQID") 前回使用 QUERY-ID
%ZGDAY=$P(%KEY("ZHN"),",",1); システム日
%ZGHN=$P(%KEY("ZHN"),",",2); クライアント・ハンドル
%ZGFLG("CLOSE")=^ZGSYS("CONFIG","close")の値
%ZGFLG("HISTORY")=^ZGSYS("CONFIG","app","gateway","HP5")の値
%ZGCODE=表示文字コード18
      SP(^ZGSYS("MGWAPP",%ZGAPP,"encode"),"^",2) より獲得(EUC etc)
%ZGCHRS=HTML charset 値
      SP(^ZGSYS("MGWAPP",%ZGAPP,"encode"),"^",1) より獲得(euc-jp etc)
```
 - 2) グローバル拡張シンタックスの UCI,VOL を得る。[gUCIVOL^ZGLIBN]


```
^ZGQRY %ZGUQRY,%ZGVQRY
^ZGRD %ZGURD,%ZGVRD
^ZGLOG %ZGULOG,%ZGVLOG
^ZGJOB %ZGUJOB,%ZGVJOB
```
 - 3) %KEY("ZHN")が存在しない場合、クライアントハンドルを得る。[INIT^ZGSND]
 クライアントハンドルが得られない場合、エラーメッセージを出力する。
 - 4) %KEY("ZNO")が存在しない場合に以下の条件で%KEY("ZNO")を作成する。


```
%KEY("ZHP2")が存在する %KEY("ZNO")=2
%KEY("ZHP3")が存在する %KEY("ZNO")=3
%KEY("ZHP0")が存在する %KEY("ZNO")=0
%KEY("ZHP5")が存在する %KEY("ZNO")=5
```
 - 5) %KEY(ZT1,ZT2,ZT3)が存在する場合、%ZGCODE により、\$\$DeCode^ZGCCNV をコールし、内部コード(EUC)へ変換する。
- [各画面呼び出し]
 - 1) %KEY("ZNO")の値により、[Search Menu],[Result list],[Data list],[Detail Data] [SearchHistory][Close]処理をコールする。

画面	条件
[Search Menu]	%KEY("ZNO")が存在しない、又は%KEY("ZNO")=1 の場合
[Result list]	%KEY("ZNO")=2 の場合
[Data list]	%KEY("ZNO")=3 の場合
[Detail Data]	%KEY("ZNO")=4 の場合
[Search History]	%KEY("ZNO")=5 の場合
[Close]	%KEY("ZNO")=0 の場合

¹⁸ Gateway 検索システム、各処理で検索結果や、テキスト、ボタンなどを HTML 出力する際、%ZGCODE により EUC->出力文字コードに EnCode して書き出す。(内部コードの EUC 以外に SJIS,UTF8 での出力が可能)

4.6.2 ターゲット選択・検索式入力[Search Menu]

- 1) %ZGAPP(表示単位 ID)をパラメータに、検索対象ターゲット DB リストを獲得(gDB^ZGLIBN)する。
- 2) 検索対象 DB と検索条件を入力するフォームを作成し、獲得した DB リスト順に検索対象ターゲット DB 名を一覧出力し、検索用 INDEX 名・検索語・演算方法を入力するフィールドを出力する。
- 3) リストに存在する各 DB 番号がアクセス可能か (\$\$isADB^ZGLIBN)、クライアント検索禁止か判断し、可能な場合は、チェックボックスを出力する。

フィールド名称	型	値の個数	値
ZDB	チェックボックス	複数	DB 番号
ZF	リストボックス	3	^ZGSYS("KEY")のキー番号
ZO	リストボックス	1	演算方法・組み合わせ
ZT	テキスト	3	キー番号に対応する検索語
ZHN	HIDDEN	1	ログイン・ハンドル(INIT^ZGSND で獲得した値)
ZHP2	SUBMIT	1	検索開始 / [Result list]へ移動を示す。
	RESET	1	入力消去
ZHP0	SUBMIT	1	終了 %ZGFLG("CLOSE")=1 の場合のみ出力。
ZHP5	SUBMIT	1	検索履歴 %ZGFLG("HISTORY")>0 の場合で、既に検索実行を行っている場合のみ出力。

4.6.3 検索処理の実行・検索結果一覧出力[Result list]

- [検索処理の実行]

%KEY("ZQID")が存在しない場合は、[Search]画面で入力値による検索処理を実行する。

 - 1) %KEY("ZDB")が存在しない場合は、エラーメッセージを出力し戻る。
 - 2) 入力された検索語と演算方法・組み合わせが整合していない場合エラーメッセージを出力し戻る。
 - 3) 検索語(%KEY("ZT"))と演算子(%KEY("ZO"))とキー番号(%KEY("ZF"))より DB 毎に ^ZGSYS("QUERY")を参照し、DB 毎の Z39.50(BIB-1 属性)/CAT-P(フィールド名)を得て検索式(逆ポーランド記法による)を作成する。
 - 4) SEARCH 要求送信。[SRCQ^ZGSND]
 - 5) ステータスがエラーならば、エラーメッセージを出力し戻る。
 - 6) SEARCH 応答が完了するか、エラー終了なるか、または繰り返し回数%ZGSTRY になるまで%ZGSWAIT 値 HANG しながら応答を待つ。[SRCR^ZGSND]
 - 7) ステータスがエラーならば、エラーメッセージを出力し戻る。

- [検索結果一覧出力]
 - 1) 検索応答で得た DB 毎のステータスリストよりステータス、ヒット件数、メッセージを得る。
 - 2) [Search Menu]へ戻るための送信データとして ZHN を付けたアンカーを出力する。¹⁹

アンカー	フィールド名	値
[Search Menu]	ZHN	ログイン・ハンドル
		%ZGFLG("HISTORY")>0 の場合のみ
[Search History]	ZHN	ログイン・ハンドル
	ZHP5	検索履歴への移動を示すフィールド名
	ZQID	QUERY-ID %ZGQID 値

- 3) 一覧出力対象 DB の選択と出力範囲を入力するためのフォームを作成する。
- 4) ステータスが完了でヒット件数が 1 以上なら PRESENT 要求するための DB を選択できるようにラジオボタンを出力する。
- 5) ステータスが完了なら DB 名、ヒット件数を出力する。
- 6) ステータスが処理中なら DB 名、"検索中"メッセージ(ヒット件数の代り)を出力する。
- 7) ステータスがエラー完了なら、DB 名、エラーメッセージ(ヒット件数の代り)を出力する。

¹⁹ これ以外の送信データとして MGWLPN と MGWAPP がある。

- 8) ヒット件数 1 以上の DB が存在すれば [Result list] へ移る「一覧作成」ボタンを出力する。
 9) 処理中の DB が存在すれば再表示するための「結果更新」ボタンを出力する。
 10) ヒット件数 1 以上の DB が存在せず、かつ処理中の DB も存在しなければ「戻る」ボタンを出力する。

フィールド名	型	値の個数	値
ZDB	ラジオボタン	1	データ出力対象の DB 番号
ZRANGE	リストボックス	1	レコード出力範囲(10,20,30)
ZHP2	SUBMIT	1	結果更新 / [Result list] の再表示を示す。
ZHP3	SUBMIT	1	一覧作成 / [Data list] への移動を示す。
	SUBMIT	1	戻る / [Search
ZHN	HIDDEN	1	ログイン・ハンドル(INIT^ZGSND で獲得した値)
ZNO	HIDDEN	1	1 [Search Menu] への移動を示す。戻るボタンが出力された場合。
ZQID	HIDDEN	1	QUERY-ID %ZGQID 値

4.6.4 データ一覧出力 [Data list]

- ZDB フィールドが存在しない場合、エラーメッセージを出力し戻る。
- [Result list からの初回呼び出し]
%KEY("ZDIR") が存在しない場合の処理
 - 選択された DB 番号のヒット件数を得て、ZRANGE 値より PRESENT 要求時のレコード獲得範囲 (%KEY("FROMTO")) を作成する。
 開始レコード 1
 終了レコード ヒット件数と RANGE 値の小さい数値
 - RID に対応する 1 レコード目が存在しない場合は、PRESENT 要求を送信する。[PRSQ^ZGSND] ステータスエラーならば、エラーメッセージを出力し戻る。
 - PRESENT 応答が完了するか、エラー終了するか、または繰り返し回数 %ZGPTRY になるまで %ZGPWAIT 値 HANG しながら応答を待つ。[PRSR^ZGSND]
 - ステータスがエラーならば、エラーメッセージを出力し戻る。
 処理中なら再度応答獲得するための [レコード取得中] 処理後戻る。
 完了なら [データ一覧出力] へ移る。
- [Data list/Detail Data による呼び出し]
%KEY("ZDIR") が存在する場合の処理
 - %KEY("ZDIR") の値によりレコード出力範囲 (%KEY("FROMTO")) を計算する。

値		
C		%KEY("FROMTO") の値をそのまま利用。(Detail Data からの呼び出し)
F	開始レコード	前回出力済レコードの次の値
	終了レコード	開始レコード + レコード出力範囲とヒット件数の小さい値
B	開始レコード	前回出力済レコード - レコード出力範囲 * 2 または 1 の大きい値
	終了レコード	開始レコード - 1 + レコード出力範囲

- 開始レコードが獲得済みかどうか確認し、獲得済みでない場合は、再度 PRESENT 要求を送信する。
[PRSQ^ZGSND]
- PRESENT 応答が完了するか、エラー終了するか、または繰り返し回数 %ZGPTRY になるまで %ZGPWAIT 値 HANG しながら応答を待つ。[PRSR^ZGSND]
- ステータスがエラーならば、エラーメッセージを出力し戻る。
 処理中なら再度応答獲得するための [レコード取得中] 処理後戻る。
 完了なら [データ一覧出力] へ移る。

- [レコード取得中]
 - 1) 次の送信データ付きのアンカーを出力する。

アンカー	フィールド名	値
[Search Menu]	ZHN	ログイン・ハンドル
		%ZGFLG("HISTORY")>0 の場合のみ
[Search History]	ZHN	ログイン・ハンドル
	ZHP5	検索履歴への移動を示すフィールド名
	ZQID	QUERY-ID %ZGQID 値
[Result List]	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	2/[Result list]を示す
	ZQID	QUERY-ID

- 2) レコード取得中メッセージと HIDDEN フィールドを送信するためのフォーム作成する。
- 3) 再度[Data List]へ移るための「結果更新」ボタンを出力する。

フィールド名	型	値の個数	値
ZDB	HIDDEN	1	データ出力対象のDB番号
ZRANGE	HIDDEN	1	レコード出力範囲(10,20,30)
ZHP3	HIDDEN	1	一覧作成/[Data list]への移動を示す。
ZHN	HIDDEN	1	ログイン・ハンドル(INIT^ZGSND で獲得した値)
ZQID	HIDDEN	1	QUERY-ID %ZGQID 値
ZDIR	HIDDEN	1	C
	SUBMIT	1	結果更新

- [データ一覧出力]
 - 1) DB名称、レコード出力範囲、ヒット件を出力する。
 - 2) 次の送信データ付きのアンカーを出力する。

アンカー	フィールド名	値
[Search Menu]	ZHN	ログイン・ハンドル
		%ZGFLG("HISTORY")>0 の場合のみ
[Search History]	ZHN	ログイン・ハンドル
	ZHP5	検索履歴への移動を示すフィールド名
	ZQID	QUERY-ID %ZGQID 値
[Result List]	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	2/[Result list]を示す
	ZQID	QUERY-ID
[Next]		次ページ[Data list]。終了レコード<ヒット件数の場合に出力
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	3/[Data list]を示す
	ZQID	QUERY-ID
	ZDIR	F
[Prev]		前ページ[Data list]。開始レコード>1
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	3/[Data list]を示す
	ZQID	QUERY-ID
	ZDIR	B

- 3) 開始レコードから終了レコードまで、以下の項目の出力を行う。

レコード番号
[Detail Data]用アンカー

アンカー	フィールド名	値
[TEXT]		各レコードの TEXT 詳細データの出力
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	4/[Detail Data]を示す
	ZQID	QUERY-ID
	ZMODE	T/TEXT の意味
	ZRNO	各レコードのレコード番号
[MARC]		各レコードの MARC 詳細データの出力
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	4/[Detail Data]を示す
	ZQID	QUERY-ID
	ZMODE	M/MARC の意味
	ZRNO	各レコードのレコード番号

データ(テキスト形式)

1レコード毎データを獲得(gRnoDt^ZGLIBN)し Prefrecsyn および CAT-P 識別を利用して項目データを得る。

Z39.50 MARC21(USMARC)の場合 MARC^ZGCOM
 Z39.50 SUTRS の場合 SUTRS^ZGCOM
 CAT-P の場合 MARC^ZGCCOM

MARC21 の場合、データの文字コードが返される。この値は以下の条件で定義されているターゲットのエンコード用文字コードを上書きする。

条件	変換用文字コード
MARC21 の Leader の CharCode="a"	UTF8
CharCode=" "で、ターゲットのエンコード定義 UTF8	EACC
CharCode=" "で、ターゲットのエンコード定義 UTF8 以外	ターゲットのエンコード定義 (EUC,SJIS,JIS7)

各DB毎の一覧フォーム^ZGSYS("FORM",DBn,"B")に定義されている情報を得る。(区切り文字および項目ID (MARC21:タグ番号、SUTRS:フィールド名、CAT-P:フィールド名))
 各項目データをテキスト形式に変換し、変換用文字コードと表示用文字コード(%ZGCODE)により変換し、区切り文字とともに出力します。

Z39.50 MARC21(USMARC)の場合 Cful^ZGCOM
 Z39.50 SUTRS の場合 Cfuls^ZGCOM
 CAT-P の場合 Cful^ZGCCOM

4.6.5 データ詳細出力[Detail Data]

- ZRNO フィールド>既獲得レコード数の場合、[データ獲得]処理を行う。獲得済みの場合は[データ詳細]出力へ移る。
- [データ獲得]
 - 1) QUERY-ID により参照できる^ZGQRY の各DB毎の次レコード ("next") とレコード出力範囲 ("range") とヒット件数 ("hits") よりレコード獲得範囲を決定する。[gQRY^ZGLIBN]
 - 2) PRESENT 要求を送信する。[PRSR^ZGSND]
 - 3) PRESENT 応答が完了するか、エラー終了なるか、または繰り返し回数%ZGPTRY になるまで%ZGPWAIT 値 HANG しながら応答を待つ。[PRSR^ZGSND]
 - 3) ステータスがエラーならば、エラーメッセージを出力し戻る。
 ステータスが処理中ならば、再度応答確認するための[レコード取得中]処理後を戻る。
 完了なら[データ詳細出力]へ移る。

- [レコード取得中]
 - 1) 次の送信データ付きのアンカーを出力する。

アンカー	フィールド名	値
[Search Menu]	ZHN	ログイン・ハンドル
		%ZGFLG("HISTORY")>0 の場合のみ
[Search History]	ZHN	ログイン・ハンドル
	ZHP5	検索履歴への移動を示すフィールド名
	ZQID	QUERY-ID %ZGQID 値
[Result List]	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	2/[Result list]を示す
	ZQID	QUERY-ID
[Data List]		現在ページ
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	3/[Data list]を示す
	ZQID	QUERY-ID
	ZDIR	C

2) レコード取得中メッセージと HIDDEN フィールドを送信するためのフォーム作成する。

3) 再度[Detail Data]へ移るための「結果更新」ボタンを出力する。

フィールド名	型	値の個数	値
ZDB	HIDDEN	1	データ出力対象のDB番号
ZFROMTO	HIDDEN	1	レコード出力範囲
ZNO	HIDDEN	1	4/[Detail Data]を示す。
ZHN	HIDDEN	1	ログイン・ハンドル
ZQID	HIDDEN	1	QUERY-ID %ZGQID 値
ZMODE	HIDDEN	1	T or M
ZRNO	HIDDEN	1	現レコードのレコード番号
	SUBMIT	1	結果更新

- [データ詳細出力]
 - 1) %KEY("ZQID")と%KEY("ZDB")より、Result-id (%ZGRID)を得る。
 - 2) %KEY("ZRNO")と%ZGRIDより、データを獲得する。(gRnoDt^ZGLIBN) Prefresyn および CAT-P 識別を利用して項目データを得る。

Z39.50 MARC21(USMARC)の場合 MARC^ZGCOM
 Z39.50 SUTRS の場合 SUTRS^ZGCOM
 CAT-P の場合 MARC^ZGCCOM

MARC21 の場合、データの文字コードが返される。この値は以下の条件で定義されているターゲットのエンコード用文字コードを上書きする。

条件	変換用文字コード
MARC21 の Leader の CharCode="a"	UTF8
CharCode=" "で、ターゲットのエンコード定義 UTF8	EACC
CharCode=" "で、ターゲットのエンコード定義 UTF8 以外	ターゲットのエンコード定義 (EUC,SJIS,JIS7)

- 3) DB 名称、レコード出力範囲、ヒット件を出力する。
- 4) 次の送信データ付きのアンカーを出力する。

アンカー	フィールド名	値
[Search Menu]	ZHN	ログイン・ハンドル
		%ZGFLG("HISTORY")>0 の場合のみ
[Search History]	ZHN	ログイン・ハンドル
	ZHP5	検索履歴への移動を示すフィールド名
	ZQID	QUERY-ID %ZGQID 値
[Result List]	ZHN	ログイン・ハンドル
	ZDB	データ出力対象のDB
	ZNO	2/[Result list]を示す
	ZQID	QUERY-ID
[Data List]		現ページ
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	3/[Data list]を示す
	ZQID	QUERY-ID
	ZDIR	C
[Next]		次データ[Detail Data]。レコード番号<ヒット件数
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	4/[Data list]を示す
	ZQID	QUERY-ID
	ZRNO	現レコード番号 + 1
	ZMODE	T or M

アンカー	フィールド名	値
[Prev]	前データ[Detail Data]。レコード番号 > 1	
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	4/[Data list]を示す
	ZQID	QUERY-ID
	ZRNO	現レコード番号 1
	ZMODE	T or M
[TEXT]	ZMODE = "M"の場合	
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	4/[Detail Data]を示す
	ZQID	QUERY-ID
	ZMODE	T
	ZRNO	現レコードのレコード番号
[MARC]	ZMODE="T"の場合	
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	4/[Detail Data]を示す
	ZQID	QUERY-ID
	ZMODE	M
	ZRNO	現レコードのレコード番号

5) レコード番号、データを出力する。

ZMODE=T の場合は、各DB毎の詳細フォーム^ZGSYS("FORM",DBn,"F")に定義されている情報を得る。(区切り文字および項目ID (MARC21: タグ番号、SUTRS: フィールド名、CAT-P: フィールド名)) 各項目データをテキスト形式に変換し、変換用文字コードと表示用文字コード(%ZGCODE)により変換し、区切り文字とともに出力します。

Z39.50 MARC21(USMARC)の場合 Cful^ZGCOM

Z39.50 SUTRS の場合 Cfuls^ZGCOM

CAT-P の場合 Cful^ZGCCOM

ZMODE=M の場合は、データを MARC 形式に変換しターゲットのエンコードによりコード変換し、出力します。

Z39.50 MARC21(USMARC)の場合 Crow^ZGCOM

Z39.50 SUTRS の場合 Crows^ZGCOM

CAT-P の場合 Crow^ZGCCOM

6) CAT-P で、DB名が BOOK または SERIAL の場合以下の処理も行う。

- [BHOLD/SHOLD データ検索]

BOOK の場合は、BHOLD を、SERIAL の場合、SHOLD のDBにたいして SEARCH 要求を発行し、結果を受信する。

1) BOOK/SERIAL のIDフィールドの値より、検索式 (BID="IDフィールドの値") を作成する。

2) SEARCH 要求送信。[SRCQ^ZGSND]

3) ステータスがエラーならば、エラーメッセージを出力し戻る。

4) SEARCH 応答が完了するか、エラー終了なるか、または繰り返し回数%ZGSTRY になるまで%ZGSWAIT 値 HANG しながら応答を待つ。[SRCR^ZGSND]

5) HOLD のQID を元のQID からたどれるように、^ZGQRY に登録する。

^ZGQRY("M",QID,"db",DBn,"hqid")=HOLD QID

%KEY("ZNQID")=HOLD QID

6) ステータスがエラーならば、エラーメッセージを出力し戻る。処理中なら[BHOLD/SHOLD メッセージ出力]へ移る。

- [BHOLD/SHOLD データ獲得]
 - 1) SEARCH ステータスが PRESENT 要求のレコード獲得範囲を決定する。
開始：%KEY("ZNRNO")が存在しなければ、1、存在すればその値
終了：開始 + %ZGNREC(HOLD 出力件数)-1 とヒット件数の少ない方。
 - 2) PRESENT 要求を送信する。[PRSR^ZGSND]
 - 3) PRESENT 応答が完了するか、エラー終了なるか、または繰り返し回数%ZGPTRY になるまで%ZGPWAIT 値 HANG しながら応答を待つ。[PRSR^ZGSND]
 - 4) ステータスがエラーならば、エラーメッセージを出力し戻る。ステータスが完了なら[BHOLD/SHOLD データ出力]へ移る。
- [BHOLD/SHOLD メッセージ出力]
 - 1) SEARCH 時にステータスが処理中ならば、[所蔵データベース検索中]のメッセージを出力する。
 - 2) PRESENT 時にステータスが処理中ならば、所蔵件数を出力し、[所蔵レコード取得中]メッセージを出力する。
 - 3) HIDDEN フィールドを送信するためのフォーム作成する。
 - 4) 再度[BHOLD/SHOLD データ検索・データ獲得]を行えるように「結果更新」ボタンを出力する。

フィールド名	型	値の個数	値
ZDB	HIDDEN	1	データ出力対象のDB番号
ZFROMTO	HIDDEN	1	レコード出力範囲
ZNO	HIDDEN	1	4/[Detail Data]を示す。
ZHN	HIDDEN	1	ログイン・ハンドル
ZQID	HIDDEN	1	QUERY-ID %ZGQID 値
ZMODE	HIDDEN	1	T or M
ZRNO	HIDDEN	1	現レコードのレコード番号
ZNQID	HIDDEN	1	HOLD 用 QUERY-ID %ZNQID
ZNRNO	HIDDEN	1	HOLD 用出力開始レコード番号 %ZGNRNO
	SUBMIT	1	結果更新

- [BHOLD/SHOLD データ出力]
 - 1) %KEY("ZNRNO")からデータ出力開始レコード番号、%ZGNREC とヒット件数よりデータ出力終了レコード番号を得る。
 - 2) [PREV/NEXT]のアンカーを出力する。

アンカー	フィールド名	値
[Next]	次の所蔵データ。出力終了レコード番号 < ヒット件数	
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	4/[Data list]を示す
	ZQID	QUERY-ID
	ZRNO	現レコード番号 + %ZGNREC(出力件数)
	ZMODE	T or M
	ZNQID	HOLD 用 QUERY-ID %ZNQID
	ZNRNO	HOLD 用出力開始レコード番号
[Prev]	前の所蔵データ。出力開始レコード番号 > 1	
	ZHN	ログイン・ハンドル
	ZFROMTO	レコード出力範囲
	ZDB	データ出力対象のDB
	ZNO	4/[Data list]を示す
	ZQID	QUERY-ID
	ZRNO	現レコード番号 %ZGNREC(出力件数) 又は 1
	ZMODE	T or M
	ZNQID	HOLD 用 QUERY-ID %ZNQID
	ZNRNO	HOLD 用出力開始レコード番号

- 3) HOLD の RID より、データを獲得する。(gRnoDt^ZGLIBN) 項目データを得る。(MARC^ZGCCOM)
- 4) 各 DB 毎の詳細フォーム^ZGSYS("FORM",DBn,"F")に定義されている情報を得る。(区切り文字および項目 ID (CAT-P:フィールド名))
 ZMODE=T の場合は、フォームに定義されている項目データをテキスト形式に変換し、ターゲットのエンコードによりコード変換し、区切り文字とともに出力します。(Cful^ZGCCOM)
 ZMODE=M の場合は、受信全データを MARC 形式に変換しターゲットのエンコードによりコード変換し、出力します。(Craw^ZGCCOM)

4.6.6 検索履歴[Search History]

- [出力開始 QID 選択]
 ZHQID フィールドが存在し、NULL でなければその値を出力開始(SO 開始)QID とする。
 存在しない場合で、ZQID が存在すれば、その値の 1 つ前の QID を^ZGLOG(DAY,HANDLE,"qid")より求め、その値を出力開始 QID とする。
 この時^ZGQRY("M",QID,"queryflag")により HOLD 情報がチェックし、1 の場合は出力対象としない。それ以外は、NULL をその値とする。
- [NEXT/PREV 用 QID 選択]
 出力開始 QID が NULL でなければ、^ZGLOG(DAY,HANDLE,"qid")に登録されている QID リストより、出力開始 QID を起点に検索履歴出力行数分だけ戻った値とする。最初まで戻った場合は、NULL を PREV 用 QID と決定する。
 ^ZGLOG(DAY,HANDLE,"qid")に登録されている QID リストより、出力開始 QID を起点に検索履歴出力行数分だけ進めた値とする。最後まで進んだ場合は、NULL を NEXT 用 QID と決定する。
 この時^ZGQRY("M",QID,"queryflag")により HOLD 情報がチェックし、1 の場合は出力対象としない。
- [アンカー出力]

アンカー	フィールド名	値
[Search Menu]	ZHN	ログイン・ハンドル
[PREV]	出力開始 QID が NULL 以外の場合	
	ZHN	ログイン・ハンドル
	ZHP5	検索履歴への移動を示すフィールド名
	ZHQID	PREV QID 値
[NEXT]	NEXT QID 値が NULL 以外の場合	
	ZHN	ログイン・ハンドル
	ZHP5	検索履歴への移動を示すフィールド名
	ZHQID	NEXT QID 値

- [フォーム出力]
 - 1) 検索履歴と HIDDEN フィールドを送信するためのフォーム作成する。
 - 2) 出力開始 QID より ^ZGLOG(DAY,HANDLE,"qid") から処理対象 qid を得て、最大検索履歴出力行数だけ処理する。この時 ^ZGQRY("M",QID,"queryflag") により HOLD 情報かチェックし、1 の場合は出力対象としない。
 - 3) 各 QID について検索結果応答 [SRCR^ZGSND] を呼び、DB 毎ステータスを得る。
 - 4) 検索応答で得た DB 毎のステータスリストよりステータス、ヒット件数、メッセージを得る。
 - 5) QID に対応する検索式 (INDEX, キーワード, 演算子) を出力する。
 - 6) ステータスが完了でヒット件数が 1 以上なら PRESENT 要求するための DB を選択できるようにラジオボタンを出力する。
 - 7) ステータスが完了なら DB 名、ヒット件数を出力する。
 - 8) ステータスが処理中なら DB 名、"検索中" メッセージ (ヒット件数の代り) を出力する。
 - 9) ステータスがエラー完了なら、DB 名、エラーメッセージ (ヒット件数の代り) を出力する。
 - 10) ヒット件数 1 以上の DB が存在すれば [Result list] へ移る「一覧作成」ボタンを出力する。

フィールド名	型	値の個数	値
ZQIDB	ラジオボタン	1	データ出力対象の QID_,"_DB 番号
ZRANGE	リストボックス	1	レコード出力範囲
ZHP3	HIDDEN	1	3/[Data list]を示す。
ZHN	HIDDEN	1	ログイン・ハンドル

4.6.7 終了[Close]

- ログアウト要求を呼び出す。[CLOSE^ZGSND]
- ログアウトメッセージを出力する。

5 グローバル仕様

サンプルアプリケーションで参照、更新しているグローバルについて以下順に仕様を示します。

5.1 ZGSYS

ターゲットDB情報と、ゲートウェイ設定情報が格納されます。

グローバル名称	ターゲット・ゲートウェイ設定情報			グローバルID	ZGSYS
	キー				データ
"TARGET"	DB 番号 n(1~)	"count" "dispname" "dbname" "target" "port" "cat-p" "user" "pwd" "nrs" "autotart" "prefrecsyn" "clientmax" "encode" "queryflag" "pfile" "optrecsyn" "lchar"			ターゲット毎の Gateway プロセスの数 ターゲットの DB 名称(表示用) ターゲットの DB ID ターゲットの IP アドレス ターゲットのポート番号(省略値:210) cat-p/Z3950 識別フラグ ターゲットのユーザ id ターゲットのパスワード named result set サポート有無 Gateway process 起動フラグ Preferred Record Syntax クライアント数/1 プロセス データのコード体系 ユーザ検索対象有無 Z39.50 Client パラメータファイル 特別レコード編集 MARC21 Leader Character Code 有効有無
"CONFIG"	"checker" "log" "login" "gateway" "global" "lock" "close" "search" "present"	"interval" "sleep" "directory" "preserve" "starttime" "z3950clientlog" "retry" "auto" "option" "createjob" "ZGQRY" "ZGRD" "ZGLOG" "ZGJOB" "client" "gateway" "flag" "timeout" "retry" "num" "timeout" "retry" "hold"	"max" "stop"	1 2	アソシエーション解放時間(秒) checker process sleep time(秒) Gateway/Checker プロセス用ログ dir ログ保存期間(日) ログクリア起動時刻 (時間 秒) z3950 ecall logging flag ログイン handle 獲得リトライ回数 -- 自動起動 Gateway プロセス-- 自動起動 gateway プロセスの最大数 停止までの checker Wakeup 回数 Gateway プロセスの job option 値 ジョブ再起動フラグ UCI VOL (ZGSYSはローカルUCI/VOLに配置) client lock timeout (秒) gateway/checker lock timeout close サポートフラグ search 応答監視間隔(秒) 応答監視リトライ回数 SEARCH FIELD 数フラグ present 応答監視間隔(秒) 応答監視リトライ回数 BHOLD/SHOLD 表示件数

グローバル名称	ターゲット・ゲートウェイ設定情報(その2)				グローバルID	ZGSYS
キー						データ
"CONFIG"	"app"	"gateway"	"HP5"			gateway ap プログラム id 1 検索履歴行数 2 出力DB数 管理用 ap プログラム id クライアント接続情報表示開始時刻
		"sysman"	"HP50"	"from"		
"KEY"	検索キー 番号 (1~)					1 表示名
"QUERY"	DB 番号	検索キー 番号				1 USE 又はフィールド名 2 Relation 3 Position 4 Structure 5 Truncate 6 Completeness
FORM"	DB 番号	フォーム名 ("F" OR "B")	定義行 no			1 行数 1 要素数 2 定義行データ 1 終了区切り文字位置 1 直前属性部位置 1 種別(区切文字=0/属性名=1/SP=2) 2 要素値
				-2 -1 n	^°A区切 n 単独区切 n	
"BIB-1"	"optrecsyn"	"COLON"				1 MARC 出力有無(0=無/1=有) 2 表示名称
	"Use"	数値				1 検索可フラグ(0=否/1=可) 2 Use
	"Relation"	数値				1 Relation
	"Position"	数値				1 Position
	"Structure"	数値				1 Structure
	"Trucation"	数値				1 Trucation
	"Completeness"	数値				1 Completeness
"NII-CAT"	"BOOK" or "SERIAL"	フィールド名				1 検索可フラグ(0=否/1=可) 2 フィールド説明

グローバル名称	ターゲット・ゲートウェイ設定情報(その3)				グローバルID	ZGSYS
	キー					データ
"MARC"	タグ番号					1 表示用名称 2 名称
"MGWAPP"	DB 表示単位 ID	"dbs" "description" "encode"				1 DB 番号リスト(カマ区切) 1 説明 1 HTML CharSet (euc-jp etc) 2 表示文字コード(EUC etc)

項 目	説 明
cat-p/Z3950 識別フラグ	0: Z39.50 1: CAT-P
DB 番号	ターゲットを識別するための通番
named result set サポート	1: タグレットが named result set を提供している場合 0: していない場合
Gateway process 起動フラグ	0: スタート時 1: 検索要求時
アソシエーション解放時間	Gateway プロセスが最後に起動されてから現在までの時間 > アソシエート解放時間である場合、Z39.50/Cat-P のセッションタイムアウトを指示する。
ログクリア起動時刻	0-86399 までの値 : 開始時刻 999999 : ログクリア禁止
clientmax	0: 無制限 Z39.50 / named result set non support の場合 0 でも 32 となる。
prefrecsyn	Z39.50 の場合の Record syntax MARC21 (USMARC) SUTRS
encode	SEARCH/PRESENT など送受信データのコード体系 (漢字) EUC、EACC、SJIS、JIS7、UTF8
lchar	MARC21 Leader Character Code 有効有無 0: 無効、encode の値を利用する。 1: 有効、空白なら MARC8 (EACC)、a なら UTF8
queryflag	ユーザによる直接の検索対象とするかを指定するフラグ 0: 直接検索対象 1: 直接検索禁止 (間接検索有り)
ジョブ再起動フラグ	1: Gateway/Checker プロセスが異常終了した時に再起動する。 0: 再起動しない。
SEARCH FIELD 数フラグ	0: V1.1 2 個のフィールドと演算関係ラジオボタン 1: V2.0 3 個のフィールドと演算関係リストボックス
close サポートフラグ	0: 終了ボタンを出力しない。 1: 終了ボタンを出力する。
Gateway ap プログラム id 管理用 ap プログラム id	^MGWAPP(apid)に登録する検索システムの id "Z39V2" ^MGWAPP(apid)に登録する管理システムの id "Z39MAN"
検索履歴行数	0: [検索履歴]機能をサポートしない。 1 以上: Search Menu]などに検索履歴ボタンや、[Search History]アンカーを出力するかを示す値で、かつ何件分の検索履歴を 1 ページに出力するかを示す値。 1 検索当たり最大何件 DB 検索結果を出力するのかを示す値。
出力 DB 数 HTML CharSet 示文字コード	shift_jis,euc-jp,utf-8 SJIS,EUC,UTF8

5.2 ZGJOB

各 Gateway プロセス、Checker プロセスについて情報や Gateway プロセスフリーキューの情報が格納されます。

グローバル名称	Gateway プロセス情報					グローバルID	ZGJOB
キー						データ	
"J"	DB 番号	Gateway プロセス \$J	"prn" "jobno" "sts" "chk" "off" "lst" "start" "cat-p" "nrs" "req" "qid"	アソシエーション 確立中 qid			ターゲット毎に起動されている Gateway 現在起動済みプロセスの数 Gateway プロセス名 Gateway プロセスの Job no 処理ステータス ターゲットタイムアウト checker フラグ プロセス毎停止フラグ 最終処理時刻(\$H) 自動起動フラグ cat-p/Z3950 識別フラグ named result set 有/無 現在処理中要求 QID アソシエーション確立中 QID 数
			"pro"			1	ターゲット IP アドレス^
				handle		2	ターゲットポート番号^
			"clients"			3	データベース名称^
				handle		4	ユーザ id^
			"refid"	refid	handle	5	パスワード
			"default rsn"	handle			handle 数 (処理担当クライアントの handle) Reference-id (アソシエーション済み reference-id)
							default rsn 利用 query-id
グローバル名称	Gateway プロセスキュー					グローバルID	ZGJOB
キー						データ	
"F"	DB 番号	Free プロセス \$J					
グローバル名称	CHECK プロセス情報					グローバルID	ZGJOB
キー						データ	
"CHECKER"	"pid" "log"	+\$H					checker process の pid ログクリア実行 \$H

グローバル名称	システム情報					グローバルID	ZGJOB
キー							データ
"SYSTEM"	"off"	個別停止用 DB 番号					全 Gateway,Checker 停止フラグ 個別 DB Gateway 停止フラグ
	"start"						開始時刻 (\$H)
	"stop"						終了時刻 (\$H)

項 目	説 明
Gateway プロセス名	'ZGDB'_n_99 n:ターゲットのDB番号 99:ターゲット毎のGatewayプロセスの通番(右詰め、0埋め)
処理ステータス	0:アイドル 1:処理中
ターゲットタイムアウト checker フラグ	Checker プロセスにより SET される。(300 秒)
停止フラグ	Gateway プロセスを停止させるためのフラグ 1:停止指示 0:未指示
最終処理時刻	最終処理時刻 (\$H) * Gateway プロセスが CGI からの要求に応じて処理完了したら更新される。24 時を越えた場合のことを考慮して、\$H そのままをセットする。 比較方法 絶対時間 \$P(\$H,"",1)*86400+\$P(\$H,"",2) で比較する。
自動起動フラグ	1:Gateway プロセスが固定起動ではなく、クライアント要求により自動起動されたプロセス 0:固定起動
cat-p/Z3950 識別フラグ	0:Z39.50 1:CAT-P ^ZGSYS の値がコピーされる
named result set サポート	1: ターゲットが named result set を提供している場合 0: していない場合 ^ZGSYS の値がコピーされる
Free プロセスPID	処理ステータスがアイドルのプロセス
client handle	Gateway プロセスが担当しているクライアント handle
default rsn 利用 query-id	named result set をサポートしていない DB にたいしてクライアントが最後に出力した QUERY-ID を記録する。同一 handle(クライアント)が他の QUERY-ID で PRESENT 要求が発生した場合に再度 SEARCH から実行する必要がある。

5.3 ZGQRY

QUERY-ID で管理される検索情報や、DB 毎・ステータス毎に管理される要求キューなどが格納されます。

グローバル名称		クライアント要求			グローバルID	ZGQRY
キー					データ	
"C"	システム日付 \$(H, ", ", 1) QID	"handle" "day" "query"	n			最大 QID 値
"M"		"queryflag" "db"	ターゲットの DB 番号	"cmdno" "pro"	1 2 3	システム日付毎の QID の最大値 セッション識別子 ログインシステム日(+H) 行数(入力されたデータ) INDEX ID 検索語 演算子 ID クライアント検索禁止フラグ 全ターゲット数(SEARCH DB 数)
			PRESENT ターゲットの DB 番号	"query" "refid" "rid" "pid" "hits" "mess" "mess2" "get" "next" "range" "fromto" "sts" "hqid"	n	現在のコマンド番号 1 ターゲット IP アドレス 2 ターゲットポート番号 3 テーブル名 4 ユーザ ID 5 パスワード 検索式の行数 検索式 逆ポーランド技法 Z39.50/CAT-P reference-id result-id (RID)
		"prsdb"		コマンド番号 レコード番号		アソシエート Gateway プロセス PID HIT 件数 (Z39.50/CAT-P 共通) AP レベル エラーメッセージ プロトコルレベル エラーメッセージ 取得したレコード数(0->) 次レコード番号(1->) PRESENT のレコード範囲 PRESENT 開始-終了レコード番号 qid ステータス(DB 毎) qid ステータス(DB 毎/コマンド毎) BHOLD/SHOLD 検索 QID
グローバル名称		クライアント要求キュー			グローバルID	ZGQRY
キー					データ	
"Q"	DB 番号	QID ステータス	QID			^ZGQRY("M", QID, "db", DBn, "pid") の値(前回処理 Gateway プロセスの\$J)

グローバル名称 項 目	Client 要求 説 明
QID	QUERY-ID/クエリ要求の通番(1以上)
QID ステータス	0:未処理 1:処理中 2:完了 3:クライアント情報獲得完了 -1:異常終了
cmdno コマンド番号	0:CLOSE 1:SEARCH 2:PRESENT 3:SCAN
handle セッション識別子	実際は、Client CGI 接続時の秒 (\$P(\$H, ", ", 2)) から作成される値 %KEY("ZHN") = 秒 x100+\$R(100) 単純な秒にしないのは、同時接続クライアントによるハドル獲得時の LOCK 競合を防ぐため。
result-id(RID)	結果識別子(1~連番)
rid ステータス	0:未処理 1:処理中 2:完了
BHOLD/SHOLD 検索用 QID	該当 QID が Cat-p でデータベース BOOK/SERIAL の場合のみ存在し、BHOLD/SHOLD データベースの IDフィールド から検索したときの qid をレコード毎に保存する。
クライアント検索禁止フラグ	該当 QID が、BOOK/SERIAL からの自動検索で発生したことを示すフラグ 0:許可 クライアント検索により発生 1:禁止 自動発生

5.4 ZGRD

RESULT-ID で管理される DB 毎・コマンド毎の情報や、獲得レコードなどが格納されます。

グローバル名称		結果				グローバルID		ZGRD
		キー						データ
"C"	システム日付 \$(H, ", ", 1)							RID の最大値 システム日付毎の RID の最大値
"M"	RID	"sts" "db" "query" "qid" "time" "cat-p" "rsn" コマ ンド番 号 (1,2,3)	"a" "c" "z" "r" "s" "w" "time"	"m" "resultCount" "errorMessage" "Number...ed" "resultCount" "errorMessage" "NumberOfRecordR eturned"	レコード番号 レコード番号	連番 連番 1	連 2	rid ステータス DB 番号 検索式 (z39.59 用又は/cat-p 用) 対応 QID RID 獲得日時 \$H cat-p/Z39.50 識別フラグ result set name AP ステータスコード AP レベルエラーメッセージ -- CAT-P -- ヒット件数 プロトコルレベルメッセージ レコード件数 -- Z39.50 -- ヒット件数 プロトコルレベルメッセージ レコード件数 獲得レコード数 データ (Z39.50) データ (CAT-P) Socket エラーコード Socket エラーメッセージ データ 属性 値 1 GW 要求処理開始時刻 (\$H) 2 GW 要求処理終了時刻 (\$H)
				語通番		"GLOLOC" "ATRTYPE" " "ATRVALU E"	通番 通番	

グローバル名称	結果
項目	説明
RID	ターゲット毎の結果を管理するための通番
rid 状態	0:未処理 1:処理中 2:完了
AP ステータスコード	0:成功 <0:エラーコード
サブ番号	0 : 分割なし 1 以上 : 分割あり、1...
result set name	Z39.50 の場合 named result-set 有 : query-id 無 : 'default' CAT-P の場合 : query-id
"z",サブスクリプト	<pre> -- init -- "protocol" "options" "prefMsgSize" "maxRecSize" "result" "implementedId" "implementName" "implementVersion" -- search -- "resultCount" "searchStatus" "resultSetStatus" "errorCode" "errorMessage" -- present -- "numberOfRecordsReturned" "nextResultSetPosition" "presentStatus" "errorCode" "errorMessage" "recordSyntax" -- scan -- "scanStatus" "numberOfentriesreturned" "errorCode" "errorMessage" </pre>

グローバル名称 項目	結果 説明
"c", サブスクリプト	<pre>-- open -- "errorCode" "errorMessage" "supportMethod" -- search -- "errorCode" "errorMessage" "resultCount" -- present -- "errorCode" "errorMessage" "Next-result-set-position" "Number-of-records-returned" -- scan -- "errorCode" "errorMessage" "resultCount"</pre>

5.5 ZGLOG

クライアント接続時、初期化される接続情報、要求した Query-id リストなどが格納されます。

グローバル名称	オンラインログ キー				グローバルID	ZGLOG データ
システム日付 \$(H,"",1)	セッション識別子 (handle) \$(H,"",2))	"in-time" "out-time" "cls-time" "client"				初回接続時刻 \$H 最終処理時刻 \$H close 処理時刻 \$H 1 接続 client address 2 host 3 ident 4 username
		"qid"	QID			1 処理時刻 2 CLOSE 要求送信済みフラグ(1/0)
		"cur"				処理中の QID
		"mgwapp"				1 %KEY("MGWAPP")接続アプリケーション ID

5.6 ZGDBG

トレース情報や、デバッグ用のフラグなどが格納されます。

グローバル名称	デバッグ情報					グローバルID	ZGDBG
	キー						データ
"APP"							%ZMGW2 デフォルト設定フラグ
"E"	"ルーチン, ラベル"						プログラムエラー時の\$ZE の値
"NOACCESS"	"INIT" "SEARCH" "PRESENT" "CLOSE"						外部DBアクセス不可フラグ INIT アクセス不可フラグ SEARCH アクセス不可フラグ PRESENT アクセス不可フラグ CLOSE アクセス不可フラグ
"P"	"ルーチン, ラベル"	通番		変数サブ スクリプト			通番 データ
"T"	"ルーチン, ラベル"						トレースフラグ データ
"WAIT"	DB 番号						ダミデータセット待ち時間(秒) DB 毎の待ち時間

グローバル名称	デバッグ情報
項目	説明
%ZMGW2 デフォルト設定フラグ	MGWAPP 変数が、設定されていない場合に、%ZMGW2 の TEST ルーチン内から呼び出される Gateway アプリケーション識別のための値。(仮) 0: ST^ZGWAPI Gateway V1.1 1: ST^ZGAPI Gateway V2.0
アクセス不可フラグ	\$\$isACC^ZGLIBN(コメント)コール時に参照される。各コマンド毎のアクセス不可フラグがあればその値を利用し、存在しない場合外部DBアクセス不可フラグを用いる。 0:アクセス可能 この場合 Gateway プロセスは、z39.50/cat-p client API を呼ぶ。 1:アクセス不可 この場合 Gateway プロセスは、ダミデータ作成ルーチンを呼ぶ。
"P"	pDBGpinf^ZGLIBN で指定された変数の値。 V2.0 で存在するのは、トレースONの際、存在するルーチン,ラベルは、 "ZGAPI,LOG" : ZGAPI が WEBLINK の%KEY 変数を叩いている。 "ZGMAN,LOG" : ZGMAN が WEBLINK の%KEY 変数を叩いている。
"T"	トレースフラグ 0:トレースOFF 1:トレースON V2.0 でルーチン,ラベルにデータを pDBGtinf^ZGLIBN でセットしているプログラムは存在しない。
ダミデータセット待ち時間	NOACCESS=1 の場合で、Gateway プロセスがダミデータをセットする前に待つ時間。 検索中や取得中を実現するための値。各DB毎の値 > 0 ならそのDBの時のみその値を利用する。0の場合は、"WAIT"の値を利用する。

5.7 EUC2EACC

EUC コードを EACC コードに変換するためのテーブル。

グローバル名称	システム情報					グローバルID	EUC2EACC
キー							データ
EUC 文字 (1文字)						1	{h1h2h3} h1:1 バイト目の HEX 値 h2:2 バイト目の HEX 値 h3:3 バイト目の HEX 値

5.8 EACC2EUC

EACC コードを EUC コードに変換するためのテーブル。

グローバル名称	システム情報					グローバルID	EACC2EUC
キー							データ
EACC 1 バイト 目の 10 進値	EACC 2 バイト 目の 10 進 値	EACC 3 バイト 目の 10 進 値				1	EUC 文字 (対応する 1 文字)

5.9 MARC8

UTF8 と EACC 間の相互コード変換を行うためのテーブル。

グローバル名称	システム情報					グローバルID	MARC8
キー							データ
"MARC8"	CharSetSym	MARC8 1 バイト 目の 10 進 値	MARC8 2 バイト 目の 10 進 値	MARC8 3 バイト目の 10 進値		1 2	h1h2h3 (UTF8 1,2,3 バイトの HEX 値) EUC 対応文字
"UTF8"	UTF8 1 バイト 目の 10 進 値	UTF8 2 バイト 目の 10 進 値	UTF8 3 バイト 目の 10 進 値			1 2 3	CharSetSym h1h2h3 (MARC8 1,2,3 バイトの HEX 値) EUC 対応文字

グローバル名称	デバッグ情報
項目	説明
CharSetSym { AN 又は AA }	A:G0,G1 領域に呼び出される文字セットを N:0=G0,1=G1 を示す L0、L1:Latin A0、A1:Cyrillic G0:Greek H0:Hebrew GS:Greek Symbol SB:Subscript SP:Superscript EA:EACC (CJK)

5.10MGWAPP

WEBLINK アプリケーション起動用。

グローバル名称	システム情報					グローバルID	MGWAPP
	キー						データ
Z39MAN						1	ST^ZGMAN
Z39V2						1	ST^ZGAPI
Z39V2U						1	ST^ZGAPI
"DB 表示単位						1	ST^ZGAPI

6 画面一覧

サンプルアプリケーションで提供される WEB 画面を示します。

6.1 システム管理

管理メニュー	
1	DB新規登録 DB登録確認 エラーメッセージ出力
2	DB修正・削除登録[DB選択] DB修正・削除登録 DB修正・削除登録確認 エラーメッセージ出力
3	ゲートウェイ設定変更 ゲートウェイ設定確認 エラーメッセージ出力
4	ゲートウェイ開始/停止
5	クライアント接続情報 検索情報
6	検索キー登録・修正 検索キー登録・修正確認 エラーメッセージ出力
7	DB毎 BIB-1 属性登録[DB選択] DB毎 BIB-1 属性登録(Z39.50用/NII-CAT用) DB毎 BIB-1 属性登録確認 エラーメッセージ出力
8	DB毎 一覧表示項目登録[DB選択] DB毎 一覧表示項目登録(MARC21/SUTRS/NII-CAT) DB毎 一覧表示項目登録確認 エラーメッセージ出力
9	DB毎 詳細表示(TEXT)項目登録[DB選択] DB毎 詳細表示(TEXT)項目登録(MARC21/SUTRS/NII-CAT) DB毎 詳細表示(TEXT)項目登録確認 エラーメッセージ出力
11	DB表示単位新規登録 DB表示単位登録確認 エラーメッセージ出力
12	DB表示単位修正・削除登録[DB表示単位選択] DB表示単位修正・削除登録 DB表示単位修正・削除登録確認 エラーメッセージ出力
1	Z39V2 検索システム(EUC)
2	Z39V2U 検索システム(UTF8)

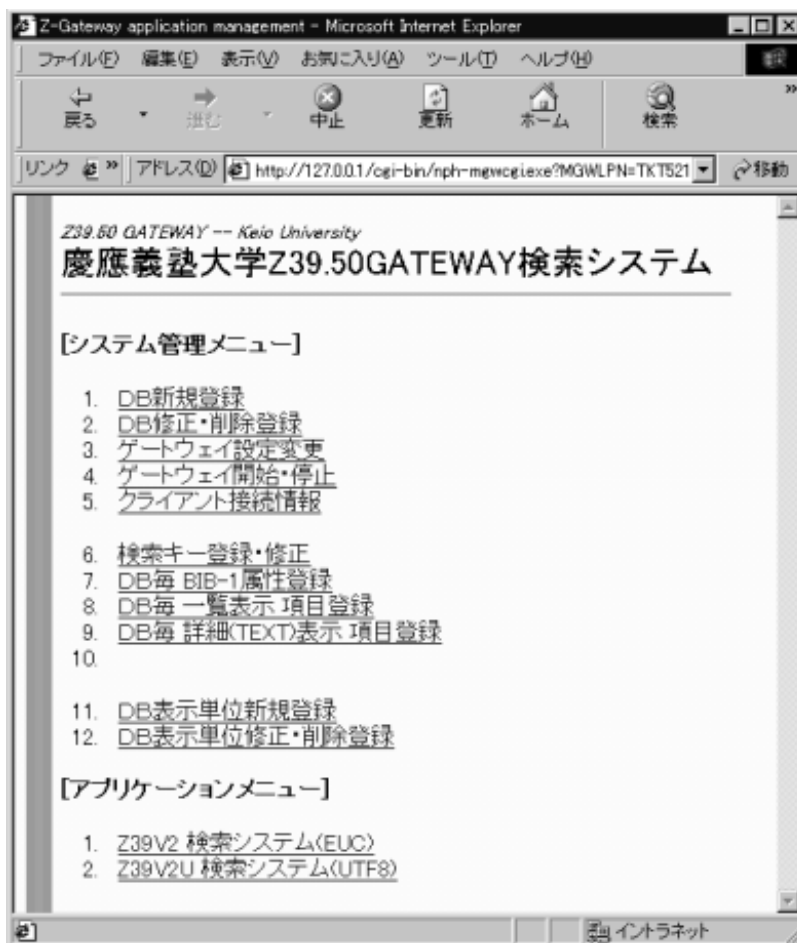
6.1.1 管理メニュー²⁰

図 6-1 管理メニュー

²⁰ この管理メニューへは MGWAPP="Z39MAN" 。検索システムを直接呼び出す場合は、MGWAPP="Z39V2"。

6.1.2 DB情報新規登録

Z39.50 GATEWAY -- Keio University
 慶應義塾大学Z39.50GATEWAY検索システム

[システム管理メニュー]

[DB新規登録]

DB番号	13	(指定又は自動採番)
DB名称	TESTDB	
DB ID名	TESTDB	
IPアドレス	127.0.0.1	
ポート番号	210	
ユーザ名		
パスワード		
Z39.50/CAT-P識別	<input checked="" type="radio"/> Z39.50 <input type="radio"/> CAT-P	
NamedResultSetサポート	<input checked="" type="radio"/> 有 <input type="radio"/> 無	
レコードタイプ	<input checked="" type="radio"/> MARC21 <input type="radio"/> SUTRS <input type="radio"/> OTHER	
特別レコード編集	none	
起動方法	<input type="radio"/> 自動起動 <input checked="" type="radio"/> 固定起動	
最大起動数	1	
クライアント数/プロセス	0	
※ NRS サポート有りの場合、0は無制限を意味し、無い場合は、最大32となる。		
コード	<input type="radio"/> EUC <input type="radio"/> EACC <input type="radio"/> JIS7 <input type="radio"/> UTF8 <input type="radio"/> SJIS <input checked="" type="radio"/> 無変換	
MARC21 Leader Character Code 対応	<input type="radio"/> 有 <input checked="" type="radio"/> 無	
※ レコードタイプ = MARC21の場合 CHARACTER CODE の値により EACC <-> UTF8 を切り替えるかを指定。		
ユーザ検索対象	<input checked="" type="radio"/> 許可 <input type="radio"/> 禁止	

登録 消去

ページが表示されました

イントラネット

図 6-2 DB新規登録



図 6-3 DB登録確認

6.1.3 DB情報修正・削除登録

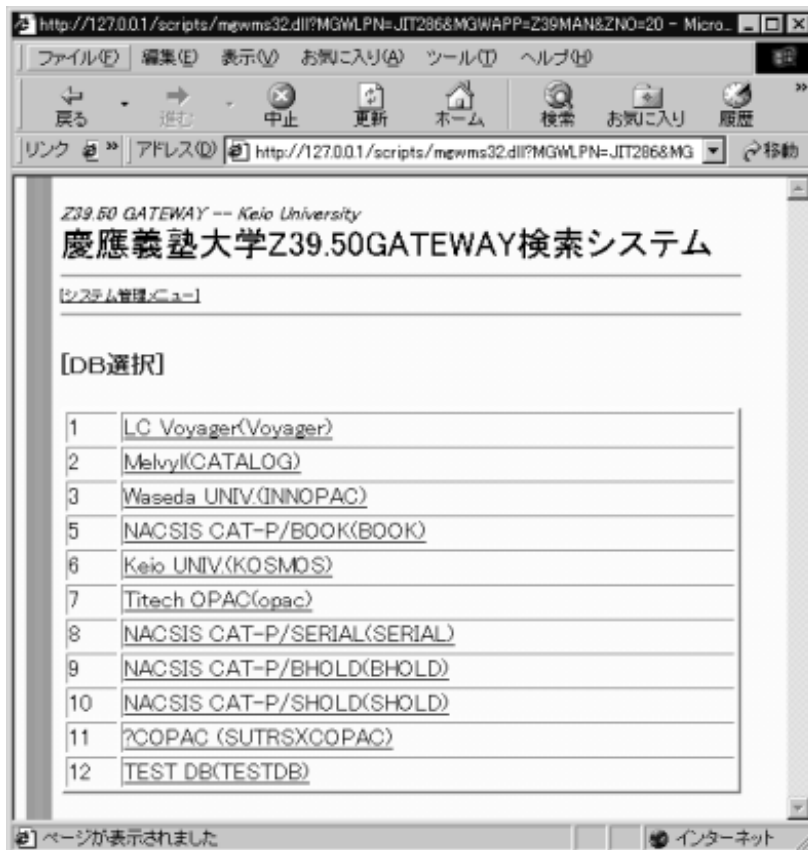


図 6-4 DB 修正・削除登録 [DB 選択]

Z-Gateway application management - Microsoft Internet Explorer

ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)

Z39.50 GATEWAY -- Keio University
慶應義塾大学Z39.50GATEWAY検索システム

システム管理メニュー(C) (詳細図)

[DB修正・削除登録]

DB番号	3
DB名称	Waseda UNIV.
DB ID名	INNOPAC
IPアドレス	wine.wu.waseda.ac.jp
ポート番号	210
ユーザ名	
パスワード	
Z39.50/CAT-P識別	<input checked="" type="radio"/> Z39.50 <input type="radio"/> CAT-P
NamedResultSetサポート	<input type="radio"/> 有 <input checked="" type="radio"/> 無
レコードタイプ	<input checked="" type="radio"/> MARC21 <input type="radio"/> SUTRS <input type="radio"/> OTHER
特別レコード編集	none
起動方法	<input type="radio"/> 自動起動 <input checked="" type="radio"/> 固定起動
最大起動数	1
クライアント数/プロセス	0
※ NRS サポート有りの場合、0は無制限を意味し、無い場合は、最大32となる。	
コード	<input type="radio"/> EUC <input checked="" type="radio"/> EACC <input type="radio"/> JIS7 <input type="radio"/> UTF8 <input type="radio"/> S.JIS <input type="radio"/> 無変換
MARC21 Leader Character Code 対応	<input checked="" type="radio"/> 有 <input type="radio"/> 無
※ レコードタイプ = MARC21の場合 CHARACTER CODE の値により EACC <-> UTF8 を切り替えるかを指定。	
ユーザ検索対象	<input checked="" type="radio"/> 許可 <input type="radio"/> 禁止

修正 削除 消去

ページが表示されました

イントラネット

図 6-5 DB 修正・削除登録

6.1.4 ゲートウェイ設定変更

[ゲートウェイ設定変更]		
解放間隔(アノンシェーション)	<input type="text" value="300"/>	(秒)
解放・自動起動監視間隔	<input type="text" value="3"/>	(秒)
ログ・ディレクトリ	<input type="text"/>	
ログ・保存期間	<input type="text" value="7"/>	(日)
ログ・クリア開始時刻	<input type="text" value="06:00:00"/>	(24時間)
ログイン・リトライ	<input type="text" value="10"/>	(回)
SEARCH・監視間隔	<input type="text" value="5"/>	(秒)
SEARCH・監視リトライ	<input type="text" value="15"/>	(回)
PRESENT・監視間隔	<input type="text" value="5"/>	(秒)
PRESENT・監視リトライ	<input type="text" value="15"/>	(回)
自動GWプロセス最大数	<input type="text" value="5"/>	/システム
自動GWプロセス停止リトライ	<input type="text" value="5"/>	(ア解放回数)
クライアント・ロックタイムアウト	<input type="text" value="3"/>	(秒)
GW・ロックタイムアウト	<input type="text" value="5"/>	(秒)
GW・CHECKER 異常終了時再起動	<input checked="" type="radio"/> 有 <input type="radio"/> 無	
検索履歴出力行数	<input type="text" value="10"/>	
検索履歴出力DB数	<input type="text" value="10"/>	
所蔵情報出力件数	<input type="text" value="10"/>	

登録 消去

ページが表示されました

図 6-6 ゲートウェイ設定変更

6.1.5 ゲートウェイ開始/停止/監視

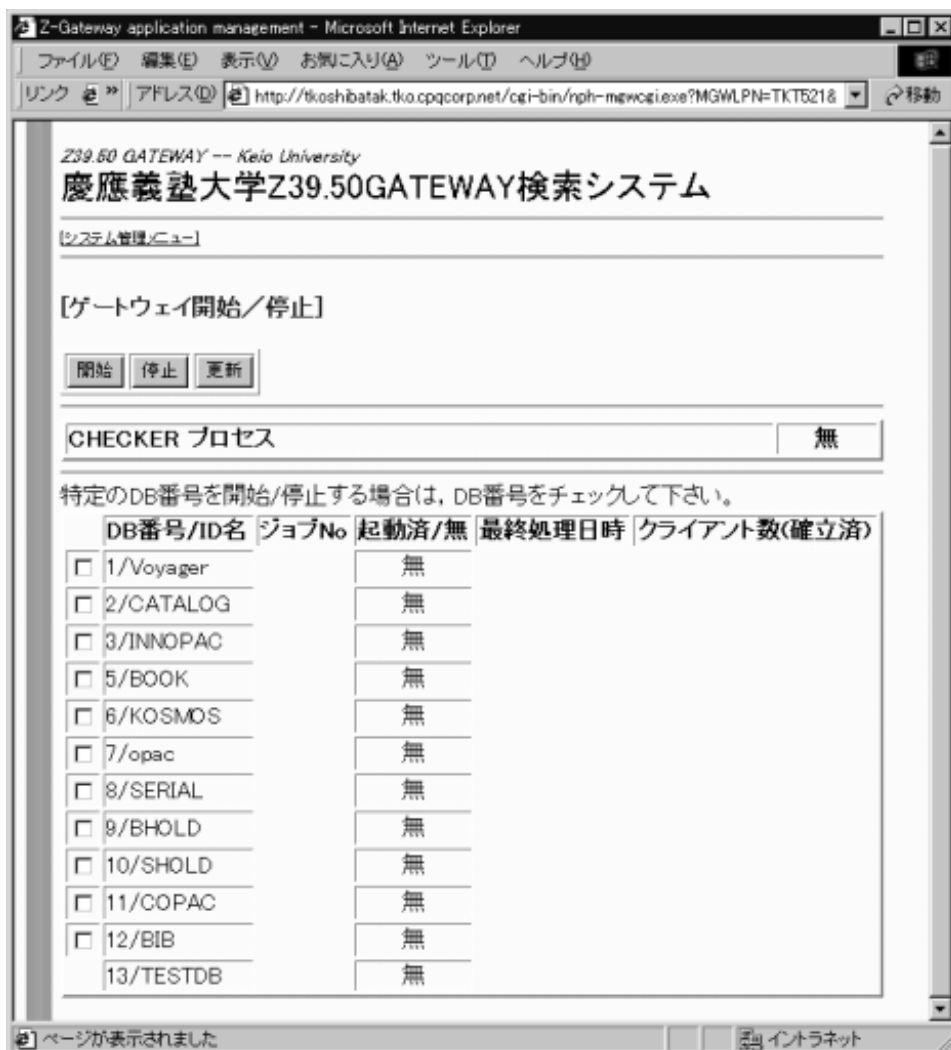


図 6-7 ゲートウェイ開始/停止（停止状態）



図 6-8 ゲートウェイ開始/停止 (開始済)

6.1.6 クライアント接続情報

Z39.50 GATEWAY -- Keio University
慶應義塾大学Z39.50GATEWAY検索システム

[システム管理メニュー]

[クライアント接続情報]

システム日付:

開始時刻~終了時刻:

接続元	接続日時	最終アクセス日時	表示単位	QUERY-ID リスト
	08/22/01 15:46:50		Z39V2E	
	08/22/01 15:47:16	08/22/01 15:48:45	Z39V2E	4 3 2
	08/22/01 15:49:08	08/22/01 15:51:24	Z39V2E	5

図 6-9 クライアント接続情報



図 6-10 検索情報

6.1.7 検索キー登録・修正

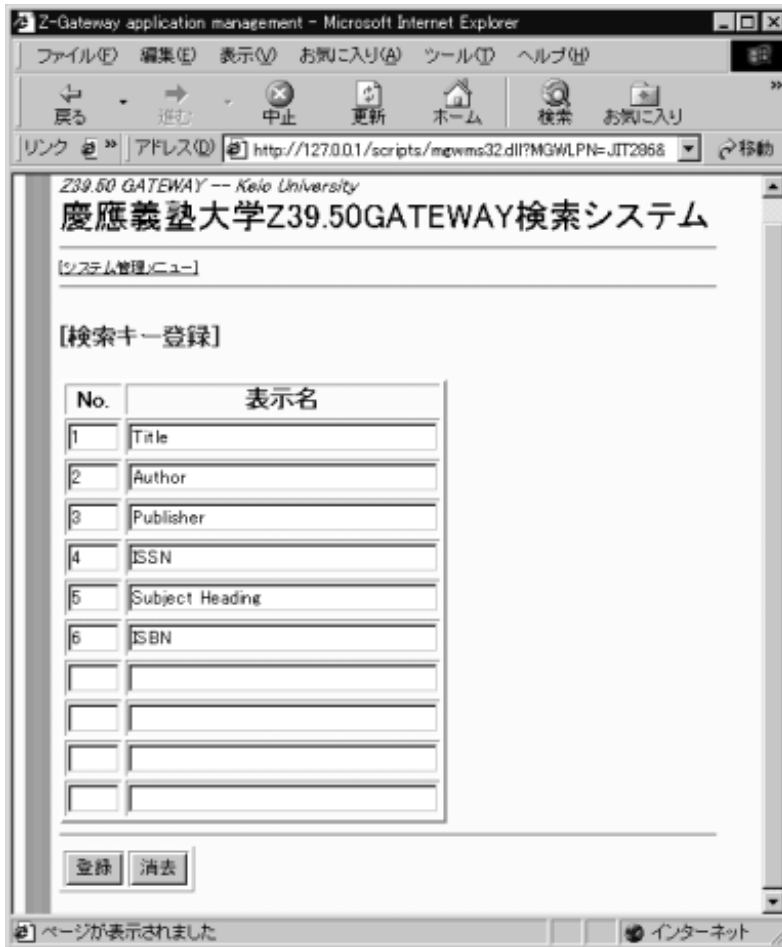


図 6-11 検索キー登録・修正

6.1.8 DB毎 BIB-1 属性登録



図 6-12 DB毎 BIB-1 属性登録(Z39.50)

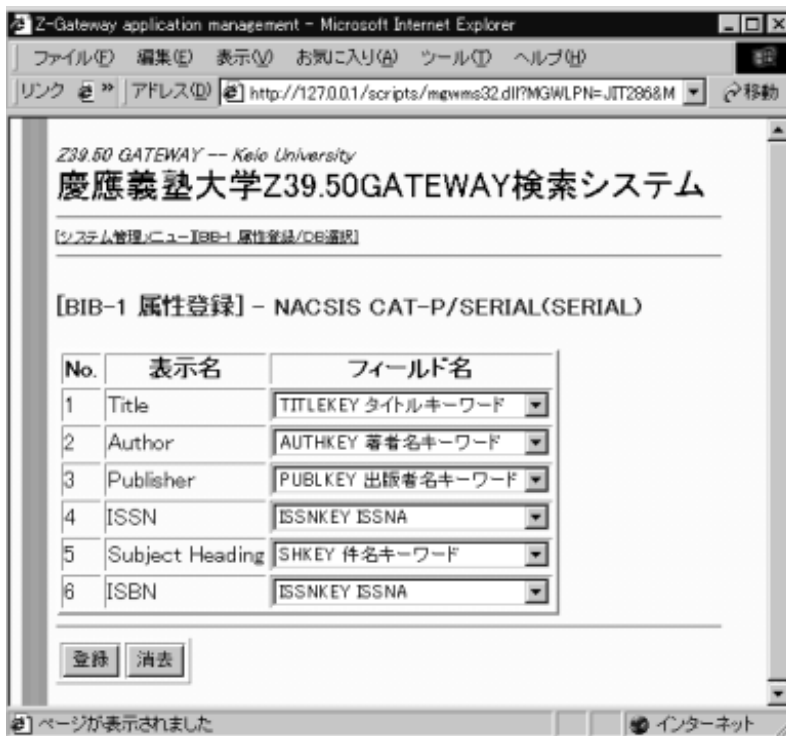


図 6-13 DB毎 BIB-1 属性登録(NII-CATP)

6.1.9 DB毎 一覧表示項目登録

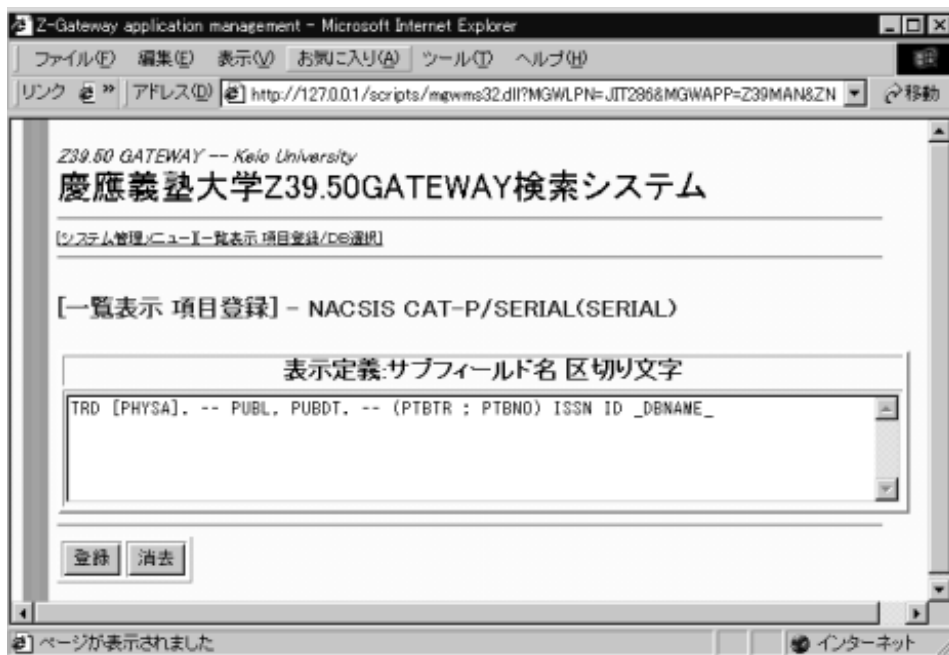


図 6-14 DB毎 一覧表示項目登録

6.1.10 DB毎 詳細表示 (TEXT) 項目登録

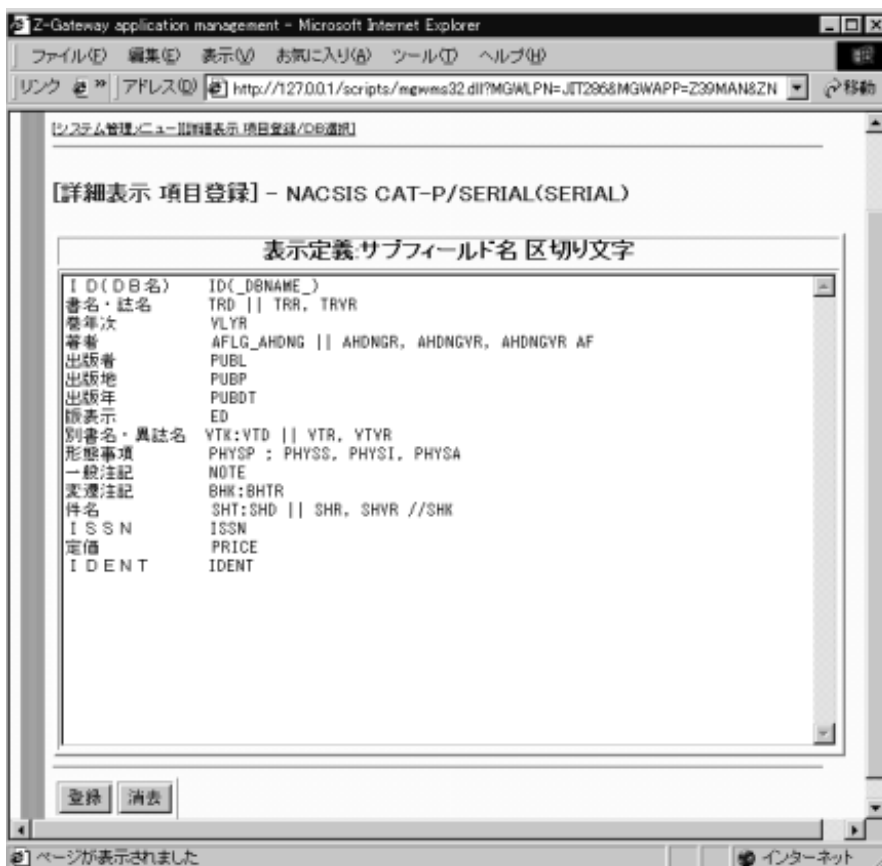


図 6-15 DB毎 詳細表示 (TEXT) 項目登録

6.1.11 DB表示単位新規登録

Z39.50 GATEWAY -- Keio University
慶應義塾大学Z39.50GATEWAY検索システム

[システム管理メニュー]

[DB表示単位新規登録]

アプリケーション(表示単位)ID

説明

エンコード EUC SJIS UTF-8

DB番号リスト

DB番号を表示順にカンマ(,)で区切り指定して下さい。

1	LC Voyager(Voyager)
2	Melvyl(CATALOG)
3	Waseda UNIV.(INNOPAC)
5	NII CAT-P/BOOK(BOOK)
6	Keio UNIV.(KOSMOS)
7	Titech OPAC(opac)
8	NII CAT-P/SERIAL(SERIAL)
11	?COPAC (SUTRS)(COPAC)
12	NII Z39.50(BIB)
13	TEST DB(TESTDB)

ページが表示されました

図 6-16 DB表示単位新規登録

6.1.12 DB表示単位登録確認

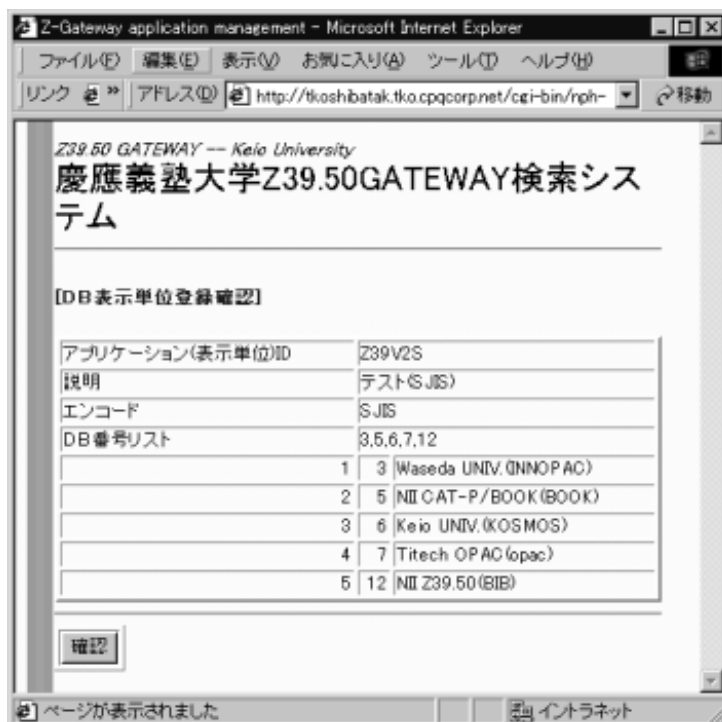


図 6-17 DB表示単位登録確認

6.1.13 DB表示単位選択



図 6-18 DB表示単位選択

6.1.14 DB表示単位修正・削除登録

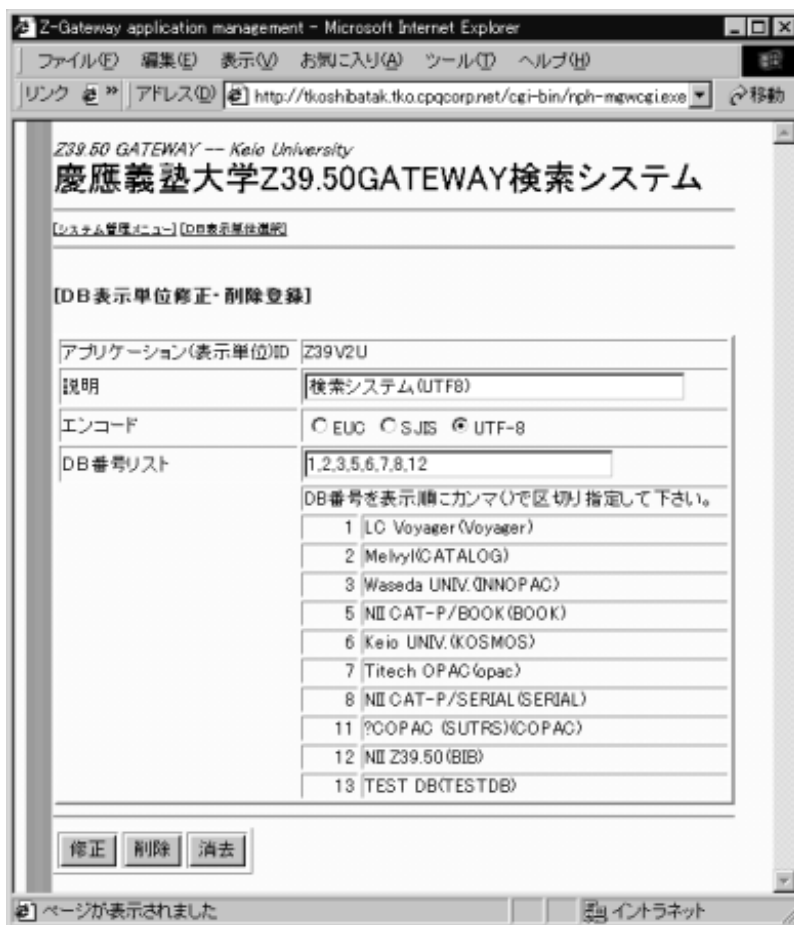


図 6-19 DB表示単位修正・削除登録

6.1.15 その他

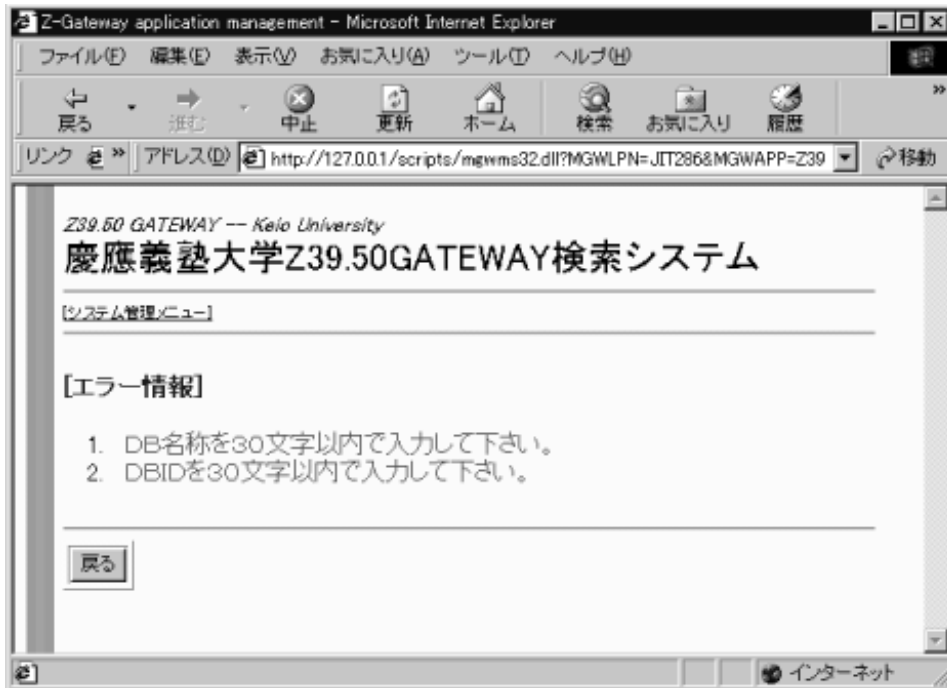
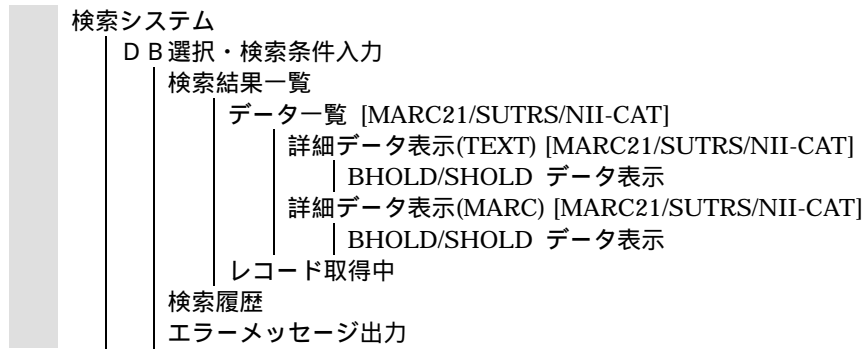


図 6-20 エラーメッセージ出力

6.2 検索システム



6.2.1 DB選択・検索条件入力

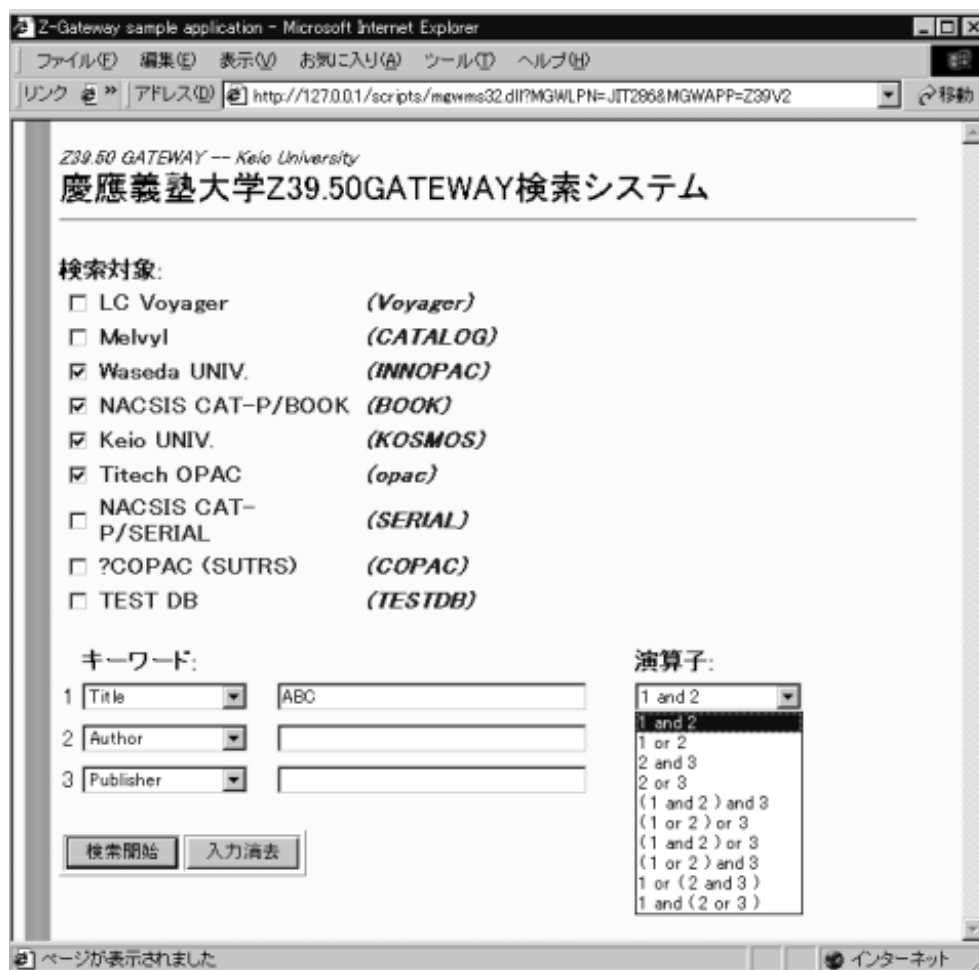


図 6-21 DB選択・検索条件入力

6.2.2 検索結果一覧



図 6-22 検索結果一覧（検索中）

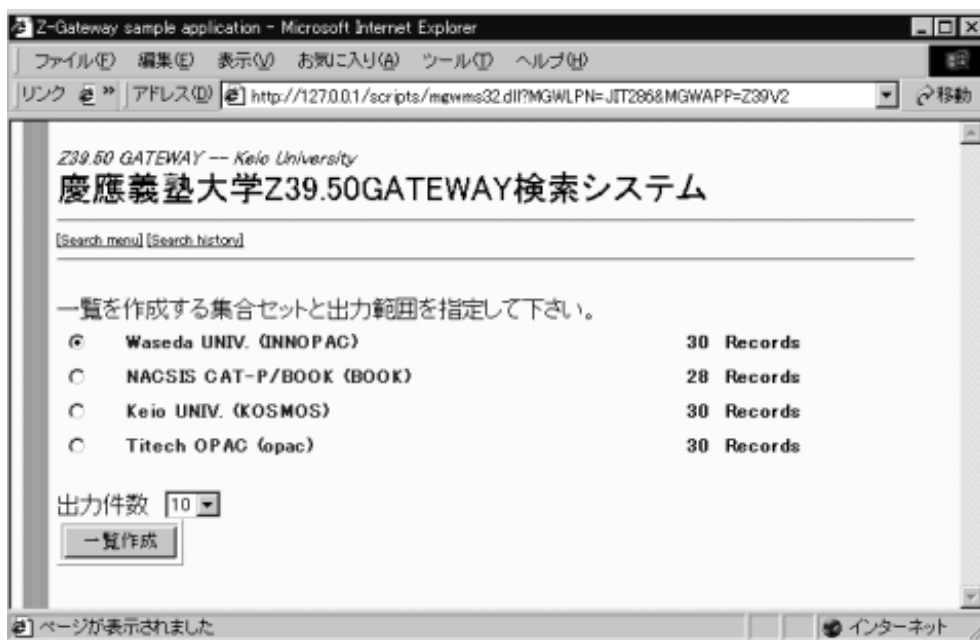


図 6-23 検索結果一覧（検索完了）

6.2.3 データ一覧

6.2.3.1 データ一覧 MARC21(USMARC)

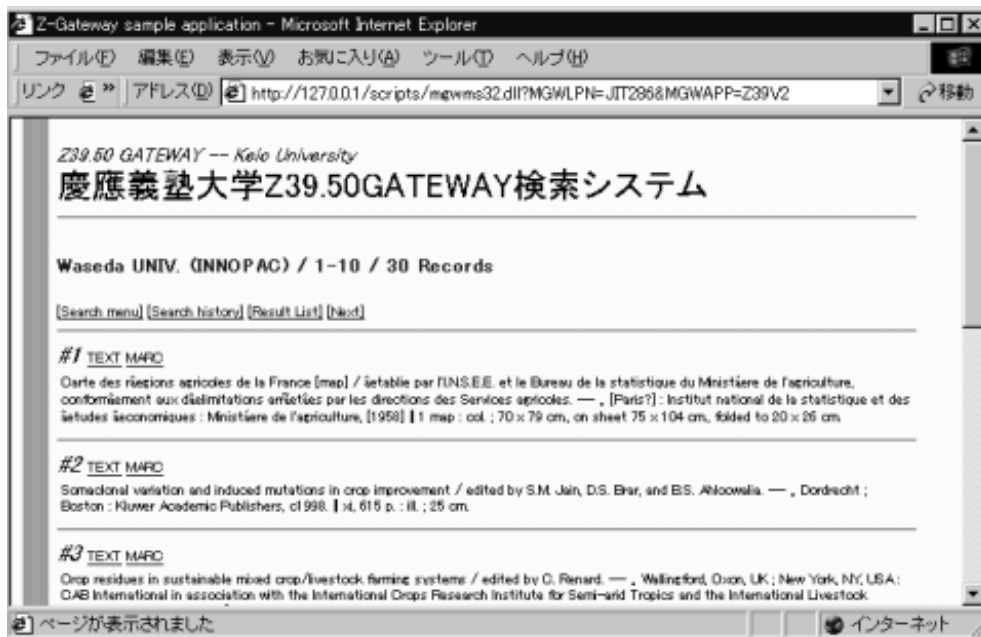


図 6-24 データ一覧 MARC21(USMARC)

6.2.3.2 データ一覧 NII-CAT

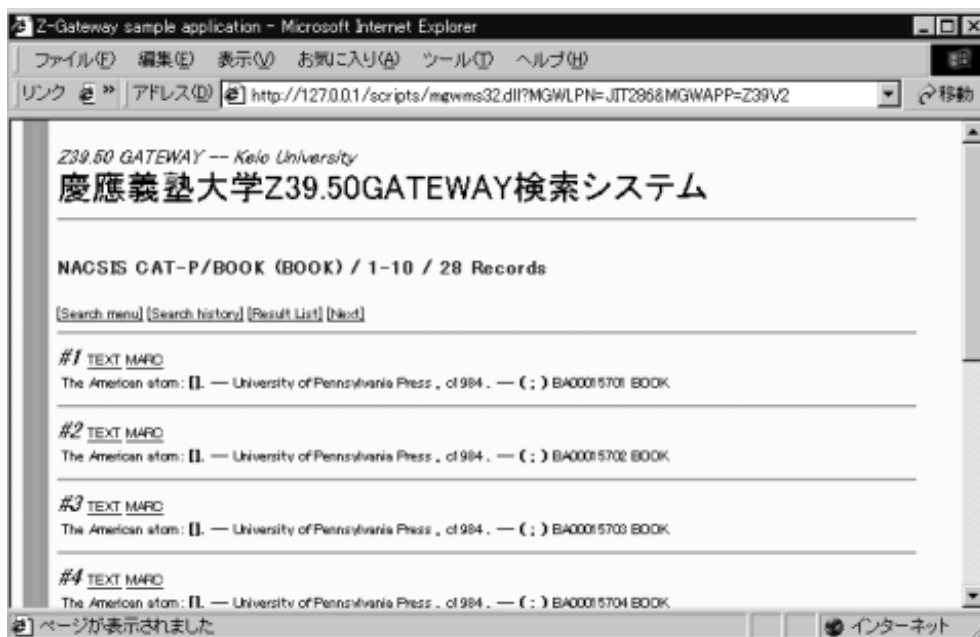


図 6-25 データ一覧 NII-CAT



図 6-26 レコード取得中

6.2.4 詳細データ表示

6.2.4.1 詳細データ表示 MARC21(USMARC)



図 6-27 詳細データ表示 MARC21(USMARC) (TEXT)

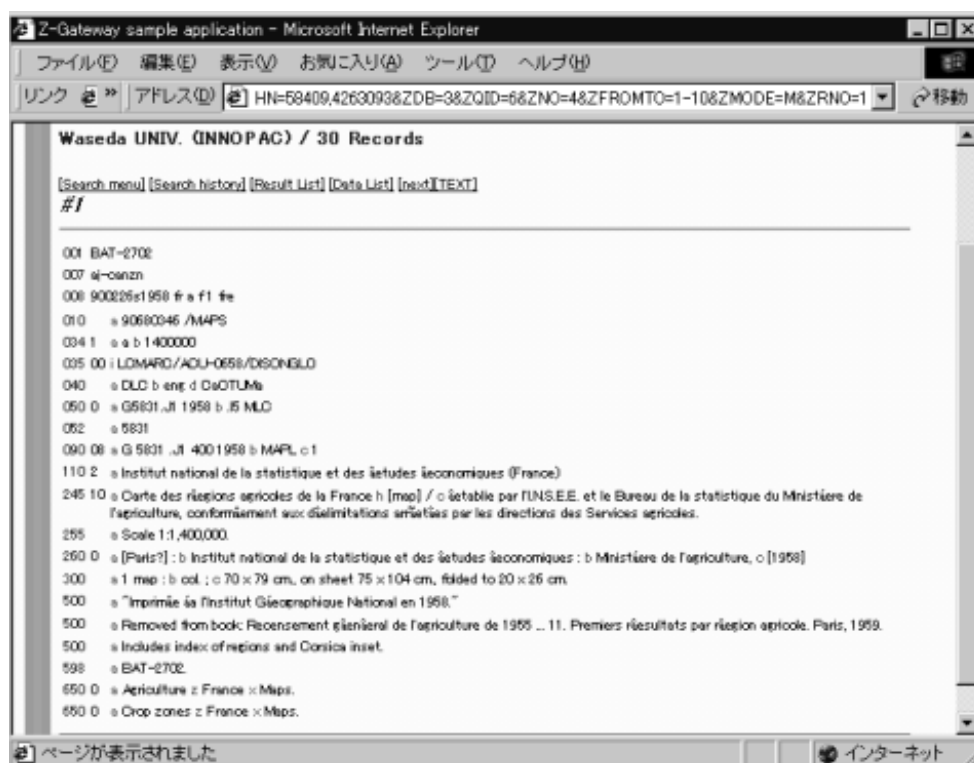


図 6-28 詳細データ表示 MARC21(USMARC) (MARC)

6.2.4.2 詳細データ表示 SUTRS

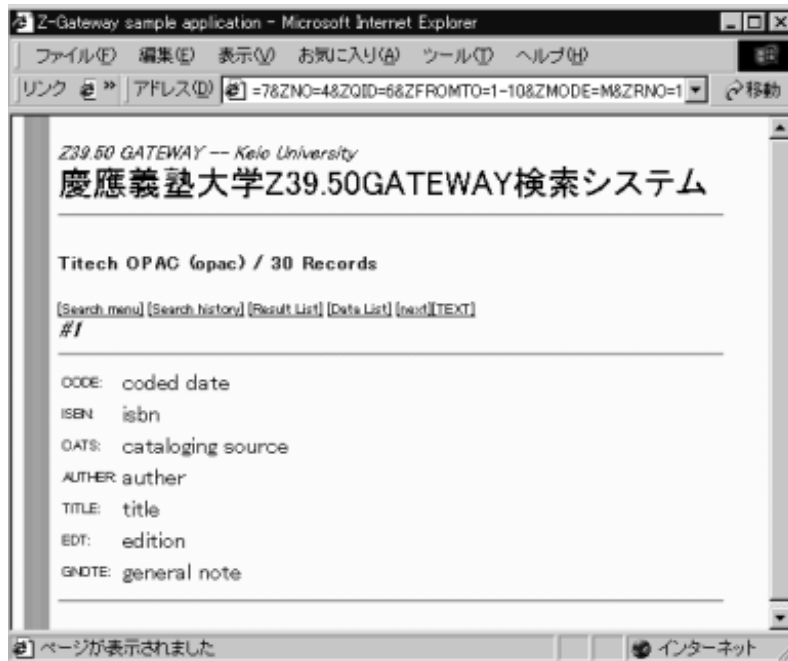


図 6-29 詳細データ表示 SUTRS (MARC)

6.2.4.3 詳細データ表示 NII-CAT

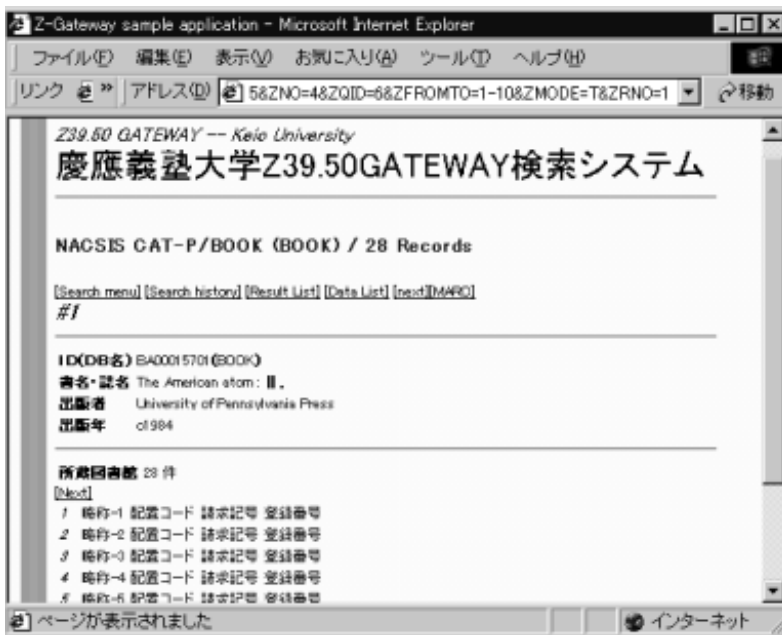


図 6-30 詳細データ表示 NII-CATP (TEXT)

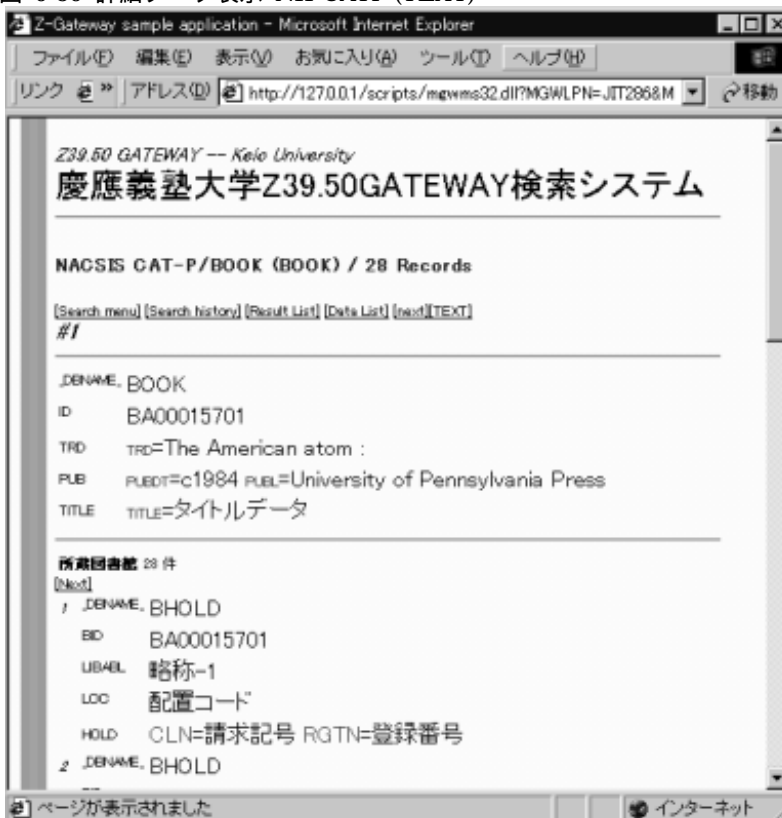


図 6-31 詳細データ表示 NII-CATP (MARC)

6.2.5 検索履歴



図 6-32 検索履歴

6.2.6 その他



図 6-33 エラーメッセージ出力