

**ExtGas/IsisGAS - revisione 3.0 :  
manuale di installazione e d'uso**

giugno 2008

Giuseppe A. Romano  
(e.mail [giuseppe.romano@isti.cnr.it](mailto:giuseppe.romano@isti.cnr.it))

**CNR: Istituto di Scienza e Tecnologie dell'Informazione**



## Indice

0.Premessa	pag	5
1.Il software	pag	7
2.L'accesso al servizio	pag	10
3.La distribuzione	pag	17
3.1.Prerequisiti di installazione	pag	17
3.2.L'installazione	pag	17
4.Definizione dei parametri di configurazione del sistema	pag	19
4.1.Parametri generali	pag	19
4.1.0.Parametri generali per la personalizzazione della interfaccia utente.	pag	27
4.2.Prametri della Query Area	pag	29
4.3.Parametri della Header Area	pag	32
4.4.Parametri della Display Area	pag	33
4.5.Parametri della Dummy Area	pag	34
4.6.Parametri della Message Area	pag	34
4.7.Parametri della Command Area	pag	34
5.Definizione dei parametri di presentazione dei risultati	pag	36
5.1 Utilizzo dell'OpenURL protocol	pag	44
6.Il file di distribuzione del software	pag	49
7.Funzionamento del sistema	pag	51
8.Installazione di un generico database	pag	55
9.Bibliografia	pag	56



## 0.Premessa

Questo manuale è il naturale aggiornamento descrittivo agli aggiornamenti e upgrade del software **ExtGAS**. Fondamentalmente sono stati aggiunti:

1. Un OPEN URL resolver
2. Una nuova API CDS/ISIS (wxis) che dovrebbe essere free dalla fine dell'anno 2008 ed è compatibile con le nuove versioni di Linux RedHat
3. Una nuova gestione del wait-time per l'attivazione dei thread
4. Un generatore di applicazioni ExtGAS cui verrà fatto cenno e riferimento al manuale specifico.

L'ultimo aggiornamento del manuale risale al giugno 2006 ed è descritto in "ExtGas/IsisGAS revisione 2.0: manuale di installazione e d'uso - ISTI-B4-16, 2006" .

**ExtGAS** è il nuovo acronimo del sistema **IsisGAS** descritto in "Isis/GAS : un sistema multi piattaforma per l'accesso a basi di dati testuali via Internet - ISTI-B4-28, 2003". Nel tempo a **IsisGAS** è stata aggiunta la possibilità di accedere a sistemi catalografici memorizzati su database system diversi da CDS/ISIS; vedi : " ExtGAS un sistema multipiattaforma per l'accesso ai cataloghi di sistemi biblioteconomici eterogenei – ISTI-B4-15,2005 ".

Per tale motivo, si è deciso di passare a una nuova versione di **IsisGAS** denominata **ExtGAS/IsisGAS** per indicare quella parte del sistema che si appoggia come sistema di memorizzazione e indicizzazione su CDS/ISIS.

Quello che segue è un manuale tecnico di installazione e utilizzo del sistema. Esso è stato completamente revisionato e in parte riscritto aggiungendo quelle modifiche che nel tempo erano state aggiunte a **IsisGAS**.

Quasi tutti gli esempi fanno riferimento alla directory **<distrib>/unix**, per ambiente Unix, ma sono validi anche per ambiente Windows. Non esiste un ambiente preferenziale.

Si presuppone che il lettore abbia conoscenza del sistema CDS/ISIS, delle sue procedure di generazione, indicizzazione e ordinamento, e conosca, inoltre, il linguaggio di stampa/formattazione dei dati (vedi Bibliografia).

Nel seguito si parlerà indifferentemente di:

1. Isis System File
2. Database System Table
3. Database System File
4. Configuration File

Intendendo il file di configurazione che definisce l'applicazione **ExtGAS** verso un particolare database CDS/ISIS. Lo stesso software può gestire più di un database per volta. Da ciò consegue che lo Isis System File definisce il database da utilizzare, le modalità di selezione e accesso ai dati e le modalità di presentazione dell'interfaccia utente e dei dati stessi.

Si tenga conto che il file di configurazione Database System File può risiedere su un

host remoto e essere accessibile via http protocol.

- `http://<host><ipport>/<ExtGAServlet>?Conf=<configurationfile>`
- `http://<host><ipport>/<ExtGAServlet?Conf=http://<host><port>/<configurationfile>`

## 1. Il software

Il software ExtGAS è un sistema per l'accesso all'informazione memorizzata su database CDS/ISIS distribuiti su rete IP. Il software è costituito da un insieme di moduli java che costituiscono il servlet ExtGAS/IsisGAS per l'accesso a database locali CDS/ISIS, altri moduli java per l'accesso ai database remoti CDS/ISIS (database omogenei) e database remoti e/o locali gestiti da differenti Database System (database eterogenei), via protocollo http, e da un modulo eseguibile, public domain, per l'accesso alle banche dati Isis (wwwi32.exe : MS Windows, e wwwisis : Unix Linux, IBM AIX, HP UX e sun solaris), distribuito da [www.bireme.org](http://www.bireme.org)) importate sui sistemi Microsoft Windows e/o Unix.

A questo deve essere aggiunto il software wxis che è la versione 5 dell'API wwwisis. In realtà, wxis è a tutti gli effetti un software per l'accesso a database CDS/ISIS usando un linguaggio XML.

Il seguente manuale si riferisce alla installazione e gestione di database omogenei (memorizzati sul sistema CDS/ISIS).

Per quello che riguarda i database eterogenei (cioè Database System non CDS/ISIS) si invia al manuale "ExtGAS – versione 3.0: Manuale di Installazione e uso : ISTI-B4-XX, 2008".

Il sistema permette l'accesso a database locali, cioè installati sullo stesso host su cui è installato il servlet ExtGAS e a insiemi di basi di dati (Logic Database) costituiti da più database fisici collocati o sulla stesso host o indifferentemente su altri host remoti che implementano il servlet ExtGAS.

Ogni database sia logico che fisico viene identificato da un file di configurazione, Database System File o Database System Table, che deve essere installato sulla stesso host su cui opera il servlet ExtGAS .

Il file di configurazione che definisce un database logico, visto come collezione di database fisici distribuiti, contiene la Logic Database Table che è un file, noto al servlet, che, a sua volta, contiene le informazioni necessarie per accedere a tutti i database fisici che costituiscono l'insieme "database logico". La Logic Database Table contiene tante righe quanti sono i database fisici che la compongono.

Ogni riga contiene sei (6) parametri obbligatori.

1. Protocol service, host name, IP port e servlet location del database fisico descritto, per esempio:

<http://leonardo.isti.cnr.it/metaopac/servlet/ExtGAS>

2. Locazione del file di configurazione del database fisico descritto, relativamente al server che lo possiede, per esempio:

/export/home/metaopac/mpisa/cnrConf/cnr.sys.file

(come da file distribuito in **<distrib>/unix/cnrConf**)

3. Tempo di wait in millisecondi per ricevere la risposta dal server remoto.
4. Numero di massimo di oggetti da inviare dal server remoto al server locale.  
Corrisponde al parametro remoto memorizzato in  
**NumberOfObjectToBeDisplayed**
5. Stringa da utilizzare nella restituzione della ricerca come primo valore costante.  
Si tenga presente che non sono ammessi spazi. Lo spazio può essere sostituito dal simbolo "+".

<hr>From: +<http://leonardo.isti.cnr.it/>

6. Stringa da utilizzare nella restituzione della ricerca come secondo valore costante.

```
/<a href="http://leonardo.isti.cnr.it/metaopac/servlet/ExtGAS?
Conf=/export/home/metaopac/mpisa/cnrConf/cnr.sys.file">CNR Pisa
Metaopac</a>
```

Gli ultimi due parametri possono assumere il valore dot '.' il che significa stringa vuota

Come esempio vedi il file **<istrib>/unix/mopConf.tgz/metaopacunipi.LogicDbFile**

L'accesso alle informazioni distribuite viene prodotto dal servlet GAS che esegue un broadcast della richiesta verso i database fisici descritti. L'accesso viene parallelizzato mediante l'attivazione dei Thread necessari a gestire le connessioni. Viene gestito l'Expire Timeout, l' Error Connection e il Malformed URL Error. E' possibile definire un wait time tra l'attivazione di un Thread e il successivo (parametro **WaitTimeForThreads**, **FwaitTimeForThreads** e **HMTFWaitTimeForThreads**) e il massimo numero di Thread da attivare (parametro **MaxThreadAllowed**).

Le informazioni accedute sui server remoti sono presentate nell'ordine in cui la lista degli accessi remoti è stata descritta. Inoltre gli http-pointer prodotti puntano ai reali server distributori dei dati.

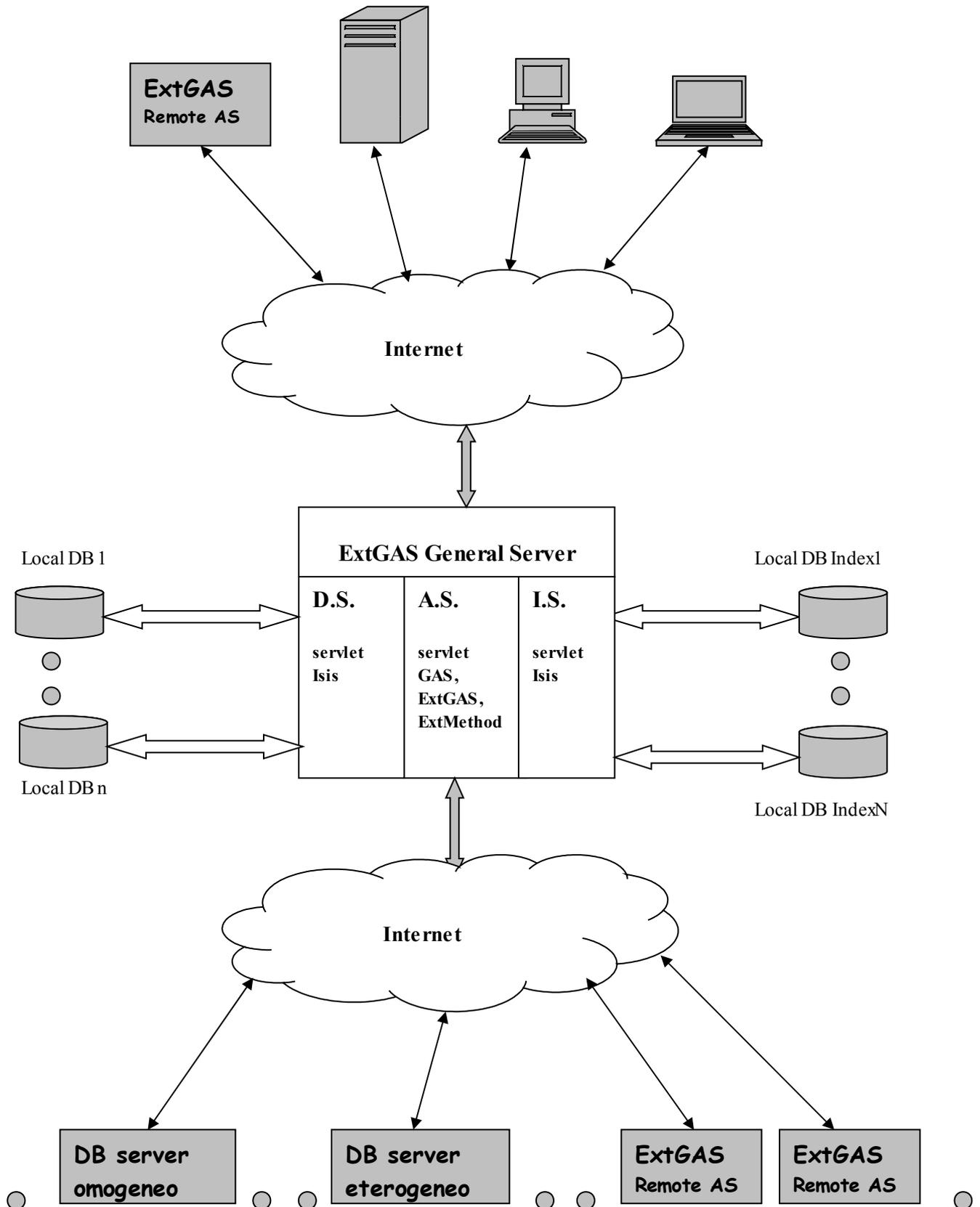
## Funzionamento del sistema ExtGAS

Nella pagina seguente nello schema logico/fisico del sistema ExtGAS le sigle DS, IS, AS hanno il seguente significato:

- DS è Database Server e rappresenta la funzione di accesso al catalogo locale.
- IS è Index Server e rappresenta la funzione di accesso agli indici estesi ottenuti sia localmente che cumulati come somma di tutti i cataloghi e utilizzabili per accedere la base di dati locale o le basi di dati remote
- AS è Access Server e rappresenta la funzione di accesso a insiemi logici e/o fisici di database remoti e/o locali.

Una volta che il servlet ExtGAS è stato attivato su un host remoto, se esiste, per lo specifico servizio, nel file di configurazione, un insieme logico di database vengono attivati i thread paralleli che gestiscono l'accesso ai database locali/remoti.

## Il Sistema ExtGAS : organizzazione logico/fisica



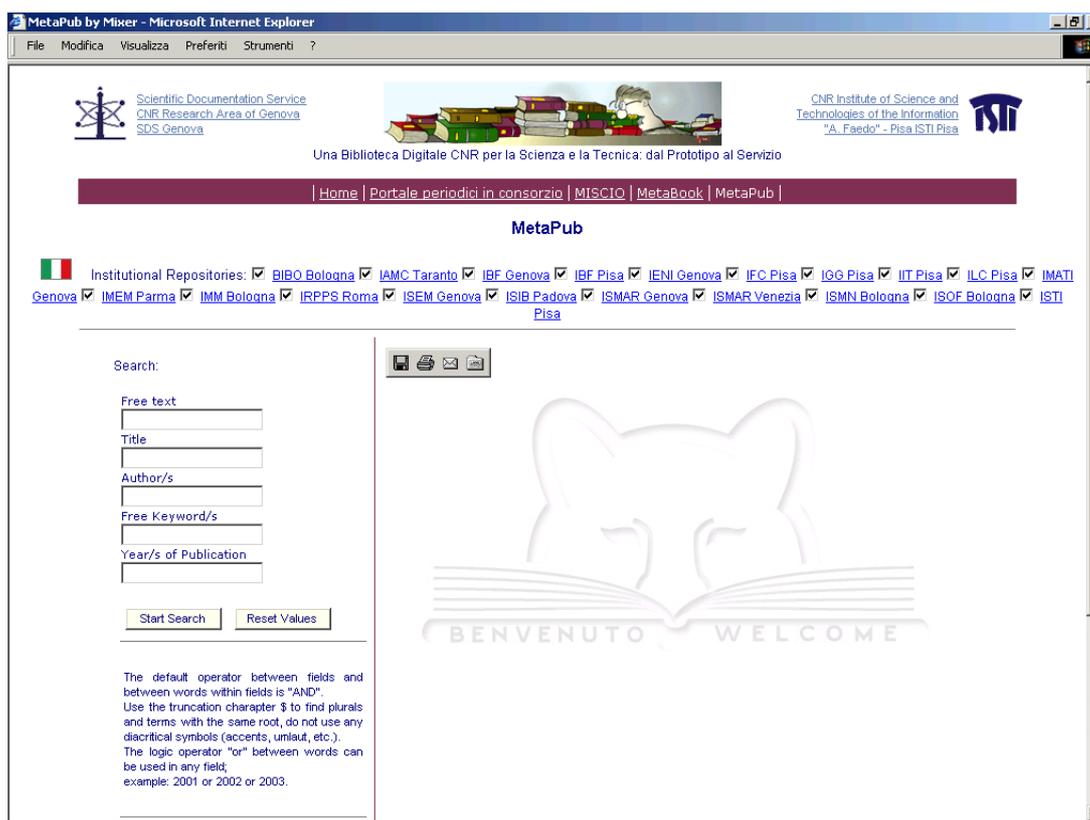
## 2.L'accesso al Servizio

L'accesso al servizio viene effettuato tramite browser WWW.

Genericamente il servlet ExtGAS viene attivato verso l'http-server di competenza con metodo **GET**. Vengono generate dinamicamente pagine HTML che richiamano ricorsivamente lo stesso servlet utilizzando sempre il metodo **GET**. Il sistema mantiene aperti contemporaneamente 2 (due) panel principali più un **HelpPanel** :

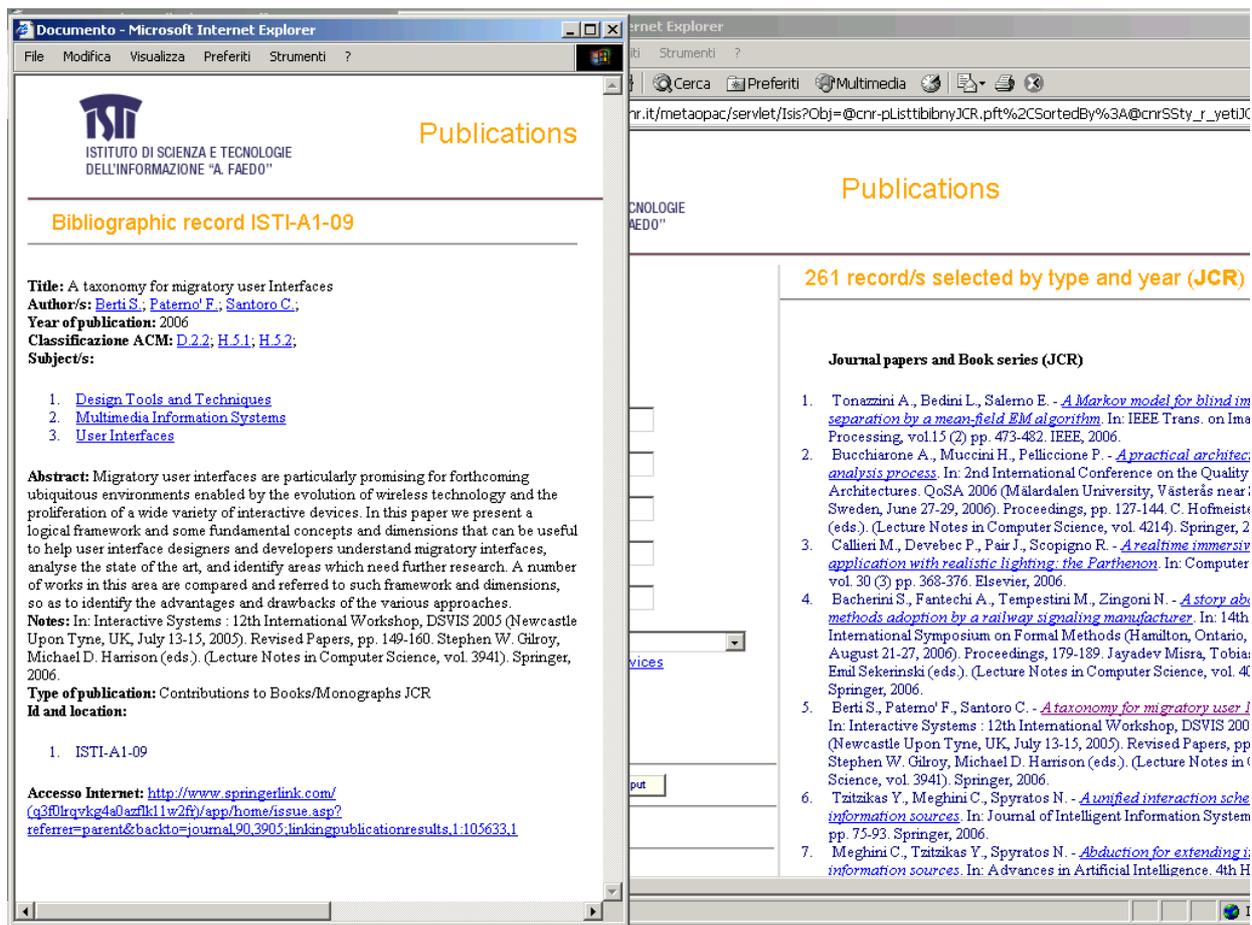
- a) il **WelcomePanel** che è il pannello principale sui cui si possono:
  - a. ottenere liste qualificate e non, dall'inverted file e dai database utilizzati come indici di supporto alla ricerca (operazione browse).
  - b. eseguire ricerche (operazione search)
  - c. visualizzare i risultati delle liste di cui sopra al punto a. e b. (operazione get)

Il sottostante è un esempio di **WelcomePanel** prodotto dalla URL:  
<http://leonardo.isti.cnr.it/metapub>:



- b) il **RecordPanel** su cui vengono presentati i risultati selezionati prevalentemente dal **WelcomePanel** e da altre istanze del **RecordPanel**.

Nell'esempio sottostante è descritto un **RecordPanel** che contiene una descrizione bibliografica ricavato aprendo il link posto sull'elemento numero 5 della lista degli oggetti selezionati :



Possono essere attivati altri particolari pannelli utilizzando il linguaggio di presentazione descritto più avanti.

A seconda dell'operazione richiesta il servlet ExtGAS attiva la classe relativa:

1. IsisBrowse
2. IsisSearch
3. IsisGet

Ognuna di queste classi comunica con il database tramite una unica API Isis (wwwi32.exe in ambiente Microsoft Windows e wwwisis pe ambiente Unix) che viene attivata tramite una serie di parametri (vedi: **<istrib>/BiremePrograms.tgz/wwwisis.rtf** nel distribution file). Al termine dell'esecuzione delle API Isis il sistema ritorna al servlet un XML file che contiene i dati identificati da tag univoci e strutturati. A questo punto, il servlet ExtGAS, utilizzando le specifiche di presentazione contenute nella Field Presentation Table costruisce dinamicamente il relativo panel di competenza utilizzando il linguaggio HTML.

Nel caso di accesso a un Logic Database (insieme di database fisici dislocati su server locali/remoti) le classi IsisSearch e IsisBrowse vengono attivate, tramite la classe GASbroadcast, sui server remoti. La presentazione delle informazioni ricevute dai server remoti viene fatta utilizzando uno stesso metodo comune (la Field Presentation Table del server che descrive il database Logico).

La classe IsisGet è, invece, sempre attivata sul server che possiede il database fisico.

Per una migliore comprensione del funzionamento delle API in relazione alle richieste del servlet vedi i **<istrib>/BiremePrograms.tgz/wwwisis.rtf** nel file di distribuzione.

Ognuno dei due Panel principali è costituito da:

- a) una **HeaderArea**
- b) una **MessageArea**
- c) una **DisplayArea**

Inoltre, il **WelcomePanel** contiene a sinistra della **DisplayArea** una **QueryArea** e una **DummyArea** (la DummyArea è riservata a uso futuro).

Di seguito vengono utilizzate le seguenti notazioni grafiche:

- I parametri racchiusi tra parentesi quadre sono opzionali: [parm=parmvalue]
- I parametri racchiusi tra parentesi angolate rappresentano una istanza generica del valore parametro: parm=<parmvalue>
- I parametri liberi (non racchiusi tra [] o <>) sono obbligatori.

Il servlet ExtGAS può essere attivato in 4 (quattro) modi distinti, sempre utilizzando il metodo GET, e per ognuno di essi sono definiti alcuni parametri:

1) Modalità welcome

- a. Conf=<ConfigurationFile>

2) Modalità browse

a. MasterFileRecord by Number

- i. Conf=<ConfigurationFile>
- ii. [SrchWin=1]
- iii. Opt=browse
- iv. Obj=<PrintFormatName>
- v. [Dsfr=<StartToDisplayFromRecordNumber>]
- vi. [Dsfor=<EndToDisplayForNumberOfRecord>]
- vii. [RReq=1] (Is a remote request from a ExtGAS/Isis server)
- viii. [Gizmo=<GizmoNameFile>/no]

b. Inverted file e Indici Alternativi by starting string

- i. Conf=<ConfigurationFile>
- ii. [SrchWin=1]
- iii. Type=Inv
- iv. Obj=<PrintFormatName>
- v. kwf=<StartInvertedFileAccessFromString>
- vi. [kwt=<EndInvertedFileAccessToString>]





```

mhl,'<Record>',/,
mhl,if p(v30) then '<font face=TimesNewRoman size=2><strong>','Titolo: ',
,'</strong>',v30,| / |v40,if p(v20) then '<a href=javascript:top.WindowOpen("<$OurServer$>?
<$OurConf$>&Obj=@unipiTiL0.pft,SortedBy:@Sti.pft&Opt=search&Field0=id=',v20^a,'"Doc");>',v20^b,'</a>'
fi,'</font>','<br>' fi,/,
mhl,if p(v80) then '<font face=TimesNewRoman size=2><strong>','Altri Titoli: ',
,'</strong>',(v80,'<br>',/),</font>' fi,/,
mhl,if p(v10) then '<font face=TimesNewRoman size=2><strong>','Opera: ',
,'</strong>','<a href=javascript:top.WindowOpen("<$OurServer$>?
<$OurConf$>&Obj=@unipiTiL0.pft,SortedBy:@Sti.pft&Opt=search&Field0=id=',v10^a,'"Doc");>',v10^b,'</a>',</font
>','<br>' fi,/,
mhl,if p(v70) then '<font face=TimesNewRoman size=2><strong>','Autore/i: ',
,'</strong>',('<a href=javascript:top.WindowOpen("<$OurServer$>?
<$OurConf$>&Obj=@unipiTiL0.pft,SortedBy:@Sti.pft&Opt=search&Field0=&Field1=&Field2=',v70,'"Doc");>',v70,'<
/a>',;','/),</font>','<br>' fi,/,
mhl,if p(v90) then '<font face=TimesNewRoman size=2><strong>',if p(v70) then 'Altri Autori: ' else 'Autore/i: ' fi,
,'</strong>',('<a href=javascript:top.WindowOpen("<$OurServer$>?
<$OurConf$>&Obj=@unipiTiL0.pft,SortedBy:@Sti.pft&Opt=search&Field0=&Field1=&Field2=',v90^a,'"Doc");>',v90
^a,'</a>',;','/),</font>','<br>' fi,/,
.....
.....
mhl,if p(v250) then '<font face=TimesNewRoman size=2><strong>',Accesso Internet: ',</strong>',v250^a,if
p(v250^b) then '<a href="',v250^b,'">',v250^b,'</a>' fi,</font>',<br>' fi,/,
/,
'</Record>',/,

```

## Formato unipiRTFbox.prolog.pft

```
mhl,if a(v1) then v1 fi,
```

## Formato unipiRTFbox.epilog.pft

```
mhl,if a(v1) then v1 fi,
```

## Formato unipiRTFbox.epilog.cnr

```
mhl,'</Start>',/,
```

## Formato unipiRTFbox.prolog.cnr

```

'<Prolog>',/,
'<CNR/Pisa>',/,
'</Prolog>',/,'<Start>',/,

```

Con l'operazione 1) viene scritto sul file di Output:

```

<Prolog>
CNR – Pisa
</Prolog>
<Start>

```

che è contenuto nel prolog unipiRTFbox.prolog.cnr, mentre viene utilizzato il formato fittizio unipiRTFbox.prolog.pft che, per ovvi motivi non scriverà mai nulla (a(v1)=non esiste il campo 1).

Con la l'operazione 2),3),4):

```
<Record>  
<contenuto di record 1/2/3>  
</Record>
```

Con l'operazione 5) viene scritto:

```
</Start>
```

che è contenuto nell'epilog unipiRTFbox.epilog.cnr mentre viene utilizzato il formato unipiRTFbox.epilog.pft che si comporta come nel caso 1).

### 3.La Distribuzione

Il file di distribuzione è disponibile in formato .tgz (compatibile con WinZip) alla seguente URL

<http://leonardo.isti.cnr.it/metaopac/ExtGasSystem.tgz>

Scaricare il file, decomprimerlo e procedere all'installazione.

#### 3.1.Prerequisiti di installazione:

##### Sistemi Operativi:

MS/Windows 95/98/NT/2000,2003,XP, Unix

ATTENZIONE: per le piattaforme Unix le API Isis sono

1. **wwwisis.operatingsystemname** per Unix , e **wwwi32.exe** per Windows.
2. **wxis** per Unix/Linux, e **wxis.exe** per Windows

##### Software public domain:

Java jdk & jsdk o jre vedi <http://java.sun.com/>

Tomcat Jserver <http://tomcat.apache.org/> o altri Jserver

apache server <http://www.apache.org/>

#### 3.2.l'installazione:

Copiare i java class file dalla directory **<distrib>/unix/ExtGAS.src** nella **servlet zone** scelta. Configurare il Jserver ospite (per il Jserver Tomcat confrontare il file **<distrib>/unix/web.xml** che contiene le modalità di definizione dei servlet del sistema ExtGAS):

**ExtGAS** o **ISIS** è il servlet che gestisce il dialogo con l'utente e l'accesso ai database CD/ISIS

**ExtGASgcrun** è il servlet che attiva il Garbage Collector di java:

G=<y/n>: attiva il Garbage Collector (calcola e si prepara a rilasciare lo spazio non utilizzato.

F=<y/n>: esegue il Run Finalization (chiude i pending object).

L=<logfile>: definisce il file di log da utilizzare

T=<y/n>: attiva la registrazione delle operazioni sul <logfile>

**ExtMethod** è il servlet che ottiene i dati da Database System non CDS/ISIS.

Per il suo utilizzo si veda; “ExtGAS – versione 3.0 : manuale di installazione e d'uso – ISTI-B4-XX, 2008”

**ExtGASsfxLCP** è il servlet per gestire ExtGAS SFX LIST Cookie Pusher: per visualizzare il cookie del browser con l'eventuale OPEN URL caricata

**ExtGASsfxCP** è il servlet per gestire ExtGAS SFX Cookie Pusher: per caricare il cookie del browser con il default cookie, o selezionato da una lista o immesso direttamente da un utente

Una volta che i servlet sono stati installati verificare il funzionamento della servlet zone attivando la servlet **ItIsWorking** () e in seguito:

- 1) Copiare il file **<distrib>/unix/Doc/MetaOpacHelp.html** nella directory **htdocs** dell'apache tomcat server
- 2) Configurare la SystemDefinitionTable (Database System Table) utilizzando come esempio uno dei file di configurazione in:  
**<distrib>/unix/cnrConf/cnr.sys.file**
- 3) Configurare, se si ritiene di utilizzare database logici, i file che ne descrivono la composizione fisica (vedi:  
**<distrib>/unix/mopConf.tgz/metaopacunipi.LogicDbFile**)
- 4) Configurare la System PresentationTable utilizzando come esempio uno dei file **<distrib>/unix/cnrConf/Isis\*FieldFile**.
- 5) Generare il database CDS/ISIS locale (utilizzare come riferimento le procedure di generazione contenute in **<distrib>/unix/IsisDbdir.cnr.tgz**)
  - a. loadiso.cnr: per caricare il database
  - b. sortmst.cnr: per ordinare il database in ordine crescente di **titolo**
  - c. genproc.cnr: per generare gli indici b-tree per la selezione degli oggetti
  - d. genindexes: per generare, se si ritiene necessario, estraendo dal database i dati, gli indici di supporto all'utente per una ricerca guidata.
  - e. Fare ulteriormente riferimento al contenuto della directory **<distrib>/unix/AIX**
- 6) Attivare l'accesso al server e eventualmente limitarlo con le access list.

## 4. Definizione dei parametri di configurazione del sistema

Di seguito sono descritti uno per uno i parametri che possono essere definiti nel **IsisSystemFile**. I valori default sono racchiusi tra parentesi. Tenere presente che tutti i parametri sono opzionali (cioè possono non essere definiti, nel qual caso valgono i valori default), escluso il nome dell'archivio che, in caso di non definizione è **CDS**.

I parametri possono essere ripetuti all'interno dello stesso **SystemFile**. I valori contenuti nelle istanze dello stesso parametro sono concatenati nell'ordine in cui sono caricati.

I parametri sono caricati nell'ordine in cui sono definiti nel **SystemFile**.

Un parametro preceduto dal carattere # viene ignorato.

### 4.1. Parametri Generali

**ServletsDirName**(servlets) : directory che contiene le java class del servlet ExtGAS

**ProgramName**(Isis) : nome del servlet che attiva l'applicazione

**IsisProgramName**(C:\wwwi32.exe) : l'applicativo che accede al sistema database CDS/ISIS ospite: il nome della API CDS/ISIS

**EnabledwxisProgram**(false/true) autorizza il sistema ad accedere ai database CDS/ISIS utilizzando la API wxis attivata dal programma wxisProgramName.

**wxisProgramName**(C:\wxis.exe) è il programma, distribuito dall'UNESCO, che implementa la API wxis

**wxisSyntaxErrorMessage**('<error-msg>',/, 'Error ',v1091,', syntax error: ',v1092,', '</error-msg>') che definisce la modalità di presentazione degli eventuali "Syntax Error" che dovessero verificarsi.

Si notino i virtual tag v1091 e v1092 che contengono rispettivamente il codice di errore e l'errore medesimo. Il messaggio può essere personalizzato in qualunque modo si ritenga opportuno.

Nel caso particolare del sistema **MetaPub** è stato definito, per le Pubblicazioni dell'ISTI, come:

```
'<error-msg>',/, 'Pubblicazioni ISTI: Error ',v1091,', syntax error: ',v1092,', '</error-msg>'
```

**wxisEpilogMessage**(if v3333:'Shproof.pft' OR v3333:'Aix.pft' OR v3333:'v1002' then else '</Start>' fi) definisce il messaggio finale da inviare al sistema per concludere le operazioni di stampa. Il default tiene conto del virtual tag v3333 che viene caricato dal sistema con il nome del Print Format utilizzato e a seconda del relativo nome invia o meno la chiusura della lista dei record prodotti (</Start>)

Anche questo messaggio può essere personalizzato a proprio piacimento.

Nel caso particolare del sistema **MetaPub** è stato definito, per le Pubblicazioni dell'ISTI, come:

```
if v3333:'Shproof.pff' OR v3333:'Aix.pff' OR v3333:'v1002' then else '</Start>' fi
```

si noti che la costante letterale 'v1002' rappresenta il contenuto di un PrintFormat particolare utilizzato per stampare il massimo numero di documenti/chiavi selezionate

**wxisZeroRecordMessage**('<error-msg>',/,,'0 record retrieved',/,,'</error-msg>')  
rappresenta il messaggio di errore da inviare al client nel caso di nessun risultato selezionato.

Anche esso può essere personalizzato così come nel file di configurazione dell'OPAC dell'Area di Ricerca del CNR di Pisa relativo al metaopac MAI/AZALAI:

```
'<error-msg>',/,,'Your request ',v1021,' : 0 record retrieved',/,,'</error-msg>'
```

Si noti il virtual tag v1021 che contiene la query eseguita.

Per un maggiori informazioni sull'utilizzo dei parametri: **EnabledwxisProgram**, **wxisProgramName**, **wxisEpilogMessage**, **wxisZeroRecordMessage** e **wxisSyntaxErrorMessage** vedi "Da wwwisis 3.0 a wxis (wwwisis 5.0): una nuova API per ExtGAS - - ISTI-B4-XX, 2008

**WorkDir**(/tmp/,C:/TEMP/) : directory di lavoro del sistema. In questa directory vengono memorizzati i file che contengono le query e i file di risultato in formato xml. La gestione del directory viene fatta dal sistema.

**GASLogicDbFile**("") : è il nome del file, che contiene la lista dei database fisici e la loro locazione, che costituisce il database logico. Se non definito il sistema ritiene che la IsisSysTable si riferisca a un unico database locale.

**IsisArchiveName**(CDS) : contiene il nome del database cui accedere.

**BrowseLogic**(true) : attiva/disattiva, per la funzione browse, l'accesso ai database fisici che costituiscono il database logico definito in GASLogicDbFile. Il valore default è true. Valori possibili(true/false)

**SearchLogic**(true) : attiva/disattiva, per la funzione search, l'accesso ai database fisici che costituiscono il database logico definito in GASLogicDbFile. Il valore default è true. Valori possibili(true/false)

**WriteLogicSpecialResult**(false) : attiva/disattiva, per la funzione search, il display sulla Search Windows dei risultati ottenuti accedendo un insieme di database fisici. Il display, nel caso di valore true, avviene presentando solo il numero totale di documenti selezionati per ogni database. Se tale valore è maggiore di 0 esso verrà reso attivabile e selezionabile. Valori possibili(true/false)

**sfxAllowed (false/true)** : definisce l'attivazione di un OpenUrl resolver cui passare una richiesta secondo il protocollo OpenUrl: NISO standard Z39.88 (vedi alla seguente URL <http://www.exlibrisgroup.com/category/sfxopenurl> )

**sfxAllowedOnSearch (false/true)** : definisce se l'attivazione dell'OpenUrl resolver http-pointer deve essere eseguita a livello di ogni singolo risultato della lista prodotta dall'opzione **search** o semplicemente durante la presentazione di ogni record (opzione **get**)

Una trattazione più approfondita delle due precedenti opzioni e delle loro modalità di utilizzo è presente in un paragrafo a parte.

**PftCountSuffix(.countpft)** : è il suffisso del PFT file da utilizzare come counter, nel caso di **WriteLogicSpecialResult(true)**, per produrre l'output che contiene il numero di record selezionati da un database fisico che costituisce una parte di un database logico. Questa tecnica viene utilizzata per inviare al server logico solo il numero di documenti selezionati (per ottimizzare l'utilizzo della rete), mentre successivamente il suo suffisso viene cambiato con il valore contenuto in PftPrintSuffix in modo da rendere possibile l'accesso al database fisico per la lista dei documenti che soddisfano la richiesta.

**PftPrintSuffix(.printpft)** : è il suffisso del PFT file da utilizzare per produrre la stampa dei risultati che provengono da un database fisico facente parte di un database logico. Vedi PftCountSuffix.

**GeneralQuerySuffix(null)** : è il suffisso, se esiste, da appendere alla query già tradotta nel query language del sistema ospite (CDS/ISIS). Definita QUERY, la query in ISIS query language, la traduzione avviene secondo il seguente schema:

```
if (GeneralQuerySuffix!=null) QUERY="( "+QUERY+" ) "+GeneralQuerySuffix
```

**GeneralQueryPrefix(null)** : è il prefisso , se esiste, da anteporre alla query già tradotta nel query language del sistema ospite (ISIS). Definita QUERY la query in ISIS query language, la traduzione avviene secondo il seguente schema:

```
if (GeneralQueryPrefix!=null) QUERY=GeneralQueryPrefix+"( "+QUERY+" )"
```

```
if (GeneralQueryPrefix!=null AND GeneralQuerySuffix!=null)
    QUERY=GeneralQueryPrefix+"( "+QUERY+" ) "+GeneralQuerySuffix
```

**DbServerViaProxy(null)** : è la REQUEST\_URI definita per il particolare database fisico, appartenente al database logico descritto dal Database System File.

**DbLocalServer(null)** : è la REQUEST\_URI da utilizzare per il particolare database fisico in sostituzione del valore del parametro **DbServerViaProxy**.

Si tenga presente che DbServerViaProxy e DbLocalServer sono liste di REQUEST\_URI separate, ove siano più di una, da virgole (","). L'associazione viene fatta uno ad uno dalla prima all'ennesima assegnazione (max=MaxThreadAllowed).

Nel caso le due liste non siano bilanciate la conversione di ferma al valore più basso.

I due precedenti parametri sono utilizzati uno rispetto all'altro in modo da forzare l'accesso al database su un particolare server invece che su quello specifico. Si considerino le due seguenti definizioni

DbServerViaProxy <http://leonardo.isti.cnr.it/metaopac/servlet/Isis>

DbLocalServer <http://146.48.84.30:8080/metaopac/servlet/Isis>

Supposto che la richiesta di accesso al database selezioni un proxy server che in seguito indirizzerà l'accesso reale: ove si conosca realmente il nome del server che eseguirà fisicamente la richiesta, è possibile forzare l'indirizzamento.

Nel caso particolare la richiesta verso il database logico arriva al computer 146.48.84.24 tramite il proxy server leonardo.isti.cnr.it. Alcuni database fisici sono acceduti tramite il proxy server sullo stesso computer 146.48.84.24 che gestisce il database logico. E' evidente che utilizzando le due definizioni si evita il rimpallo end-user->proxy-server->server->proxy-server->server->proxy-server->end-user che diviene: end-user->proxy-server->server->proxy-server->end-user.

**XMLPftSuffix(.xmlpft)** : è il suffisso del PFT file da utilizzare per produrre la stampa dei risultati in formato XML. Naturalmente il formato che produce l'out XML deve essere tale da produrre un XML well formed a meno di non accettare errori.

ATTENZIONE:

Se non si richiede ordinamento dei risultati il formato di stampa è  
@[formatname].xmlpft

Se si richiede l'ordinamento dei risultati allora il formato di stampa e di ordinamento deve essere

@[formatname].xmlpft,SortedBy:@[sortedformatname].xmlpft

cioè ObjectToGet e ObjectToSearch devono terminare con **XMLPftSuffix**

**XMLNOXMLPftSuffix(.noxmlxmlpft)** : è il suffisso del PFT file da utilizzare per produrre la stampa dei risultati in formato XML ma non viene richiesto di elaborare direttamente l'xml (viene forzato un ContentType html/txt) Naturalmente l'xml prodotto è bene che sia well formed. Tale modo di interagire viene usato per mandare in input il file XML a qualche applicazione diversa e tale da produrre in OUTPUT dati compatibili con html/txt. Si prenda and esempio la shell che viene attivata dal sistema a fronte della definizione:

IsisProgramName /export/home/metaopac/mpisa/cnrConf/THESH

```
#!/bin/bash
filein=${1:3}
echo $filein
fileout=`grep out= $filein`
fileo=${fileout:4}.sh
```

```
fileout=${fileout:4}
query=$1
#execute ISIS access
/export/home/metaopac/mpisa/BiremePrograms/bin/wwwisis $query
#now we modify the ResultFile
executeMyPGM in=fileout out=fileo
mv $fileo $fileout
```

**WriteHTMLCharCode(true)** : definisce per default che l'output del sistema deve essere prodotto in formato XML compatibile. Nel caso non lo si desideri il parametro va impostato a "false". Può essere utilizzato nel contesto degli **XMLPftSuffix** e **XMLNOXMLPftSuffix** .

**MaxThreadAllowed(10)** : è massimo numero di Thread che possono essere attivati contemporaneamente per eseguire un accesso parallelo ai database fisici.

**WaitTimeForThreads(0)** : è il wait time (in millisecondi) da applicare tra l'attivazione di un Thread e il successivo.

**FwaitTimeForThreads(0)** : è il wait time iniziale (in millisecondi) da applicare ai primi **HMTFWaitTimeForThreads** thread attivabili.

**HMTFWaitTimeForThreads(0)** : è il numero iniziale di thread cui applicare il **FwaitTimeForThreads**.

ATTENZIONE: i due parametri **FwaitTimeForThreads** e **HMTFWaitTimeForThreads** sono presi in considerazione solo se il parametro **WaitTimeForThreads** è maggiore di 0. La definizione di tali parametri è utile quando è necessario inviare un numero elevato di richieste di accesso verso lo stesso server e quest'ultimo ha tempi di attivazione iniziali relativamente brevi perché, causa meccanismi di ottimizzazione della query inviata, richiede un numero di millisecondi elevato rispetto alle query successive che utilizzano le query già ottimizzate. Nella applicazione **MetaBook**, allo stato attuale, quattordici (14) accessi su sedici (16) sono indirizzati allo stesso server: <http://polarcnr.area.ge.cnr.it/cataloghi/> . Impostando i parametri come segue:

```
MaxThreadAllowed 20
WaitTimeForThreads 75
FWaitTimeForThreads 500
HMTFWaitTimeForThreads 2
```

Si ottiene che dei 16 accessi il primo è inviato immediatamente, il secondo dopo 500 millisecondi e il terzo, anch'esso, dopo 500 millisecondi. I rimanenti sono inviati a distanza di 75 millisecondi uno dall'altro. In questo modo si da tempo (1 secondo) al server di ottimizzare la query e accedere ai primi 2 database, successivamente in 1050 millisecondi vengono eseguiti i rimanenti accessi.

**TimeoutMessage(null)** Definisce il contenuto del **TimeoutMessage** durante l'accesso a database fisici collocati su computer remoti. Il default è **null**, nel qual caso viene inviato il messaggio standard:

Timeout expired by xxxxxx milliseconds accessing <hostname> + <ipPort > + <URI> +

QueryString

**ErrorConnectingServerMessage**(null) Definisce il contenuto del **ErrorConnectingServerMessage** durante l'accesso a database fisici collocati su computer remoti. Il default è **null**, nel qual caso viene inviato il messaggio standard:

Error Connecting Server <hostname> + <ipPort > + <URI> + QueryString

**SystemJavaScript**(on) : se impostato a **on**, come per il suo default, carica i javascript di sistema. Per escludere il caricamento dei javascript di sistema assegnare al parametro il valore off.

```
function WindowOpen(address,name) {
    thisWin=window.open(
        address,name,
        'toolbar=no,
        scrollbars=yes,
        location=no,
        status=no,
        menubar=yes,
        resizable=yes,
        width=500,
        height=600');
    thisWin.focus();
    void(0);
}
function WindowHelp(address,name)
    thisWin=window.open(
        address,name,
        'toolbar=yes,
        scrollbars=yes,
        location=no,
        status=no,
        menubar=yes,
        resizable=yes,
        width=500,
        height=600');
    thisWin.focus();
    void(0);
}
```

**LocalJavascriptFunctions** ("") : identifica il file che contiene le function javascript locali (se esistono). Viene utilizzata dalla classe IsisWriteHeader. Vedi esempio in **<distrib>/unix/isti\_piConf/cnr-pJavascript.file** per il suo utilizzo in concomitanza con definizioni dei CSS.

**DisplayAreaContentFile** ("") : identifica il file che si vuole venga presentato nella DisplayArea del MainPanel al welcome time. Viene incluso in un **<table><tr><td></td></tr></table>** **<DisplayAreaContentFile>** **</tr></table>**

**DefaultIsisFormat**(@<dbname>.pft) : è l'Isis Print Format di default per la funzione

search se non specificato diversamente.

**BrowseIsisFormat**(@<dbname>.pft) : è l'Isis Print Format di default per la funzione browse dei Master File Record Number.

**BrowseInvertedIsisFormat**(@Inv.pft) : è l'Isis Print Format di default per la funzione browse delle entry (parole) caricate nell'Inverted File a partire dalla stringa <kwf> fino alla stringa <kwt>, se non specificato altro formato.

**IsisCiparFileName**(<WorkDir>/<dbname>.txt) : è il CDS/ISIS file che contiene le definizioni (ddname) e le locazioni dei file del CDS/ISIS Database, dei PrintFormat utilizzati, dei Translate file utilizzati (gizmo files). Per maggiori informazioni vedi:  
**<istrib>/BiremePrograms/wwwisis.rtf**

**IsisSearchTranslateTable** (àèùòì<sup>°a</sup>:aeeuoi00) : è l'Isis translate table da utilizzare durante la fase di search. Ai valori dei campi viene applicata la translate table. Tale opzione viene utilizzata per normalizzare i caratteri diacritici all'inverted file costruito. Il default è nessuna translate table. La translate table è costituita da due stringhe di caratteri di lunghezza uguale separate da un carattere jolly come, per esempio, “.” (duepunti) . La stringa di sinistra rappresenta i caratteri di partenza, quella di destra quelli di arrivo. Il carattere “0” (zero) rappresenta il vuoto. Il seguente è un breve esempio:

àèùòì<sup>°a</sup>ç:aeeuoi00c

**IsisGizmoFile**(html) : è l'Isis database (\*.mst, \*.xrf) che contiene l'associazione tra i caratteri in input (del database) e i caratteri in output (da inviare al client). Viene spesso usato per convertire i caratteri diacritici. Per maggiori informazioni vedi:  
**<istrib>/BiremePrograms/wwwisis.rtf**. Se non si vuole utilizzare nessun gizmo file è necessario specificare l'opzione **Gizmo=no** nella richiesta: vedi i parametri generici di chiamata del servlet Isis.

**TRACE**(off) : è il valore del trace flag. Se settato a “on” non cancella i workfile temporanei e invia al client alcune informazioni sulla richiesta effettuata. Viene utilizzato per la messa a punto del sistema.

**IsisLogFile**(/tmp/IsisLogFile,C:/TEMP/IsisLogFile) : nome del file di log su cui vengono registrate le operazioni eseguite. Nel caso non si voglia registrare nessuna informazione il parametro va definito:

1. Ambiente Unix: **IsisLogFile /dev/null**
2. Ambiente Windows: **IsisLogFile NUL**

**HelpFileUrl** () : la locazione del file di Help del sistema

**IsisFileTable**(/tmp/IsisFieldFile,C:/TEMP/IsisFieldFile) : il file che contiene le modalità di presentazione dei vari TAG xml usati dal sistema

**IsisFilePrefix**(Isis) : prefisso da usare nella costruzione dei file temporanei di lavoro in WorkDir

**IsisQueryFileSuffix**(query.xml) : suffisso da usare nella costruzione dei file temporanei di lavoro contenenti le richieste in WorkDir

**IsisResultFileSuffix**(out.xml) : suffisso da usare nella costruzione dei file temporanei di lavoro contenenti i risultati delle richieste in WorkDir

**TableAreaWidth**(680) : Ampiezza (larghezza) della WelcomePage

**WindowOpenAreaWidth**(300) : Ampiezza (larghezza) della RecordPage

**WindowOpenAreaHeight**(400) : Lunghezza (altezza) della RecordPage

**MaxRecordsToBeDisplayed**(1) : numero massimo di record di cui viene fatta completa presentazione. ATTENZIONE: al momento il valore non è modificabile

**StartObjectToBeDisplayed**(1) : numero del record da cui iniziare a presentare gli elementi di una lista di record.

**NumberOfObjectToBeDisplayed**(50) : numero di record da presentare in una lista a partire da StartObjectToBeDisplayed.

**AllObjectToBeDisplayed**(0) : numero di record da presentare in una lista a partire da StartObjectToBeDisplayed da utilizzare con la definizione di LocalAllCommand. Se > di 0 viene attivato un comando con il valore di LocalAllCommand. Attenzione, il limite massimo non è definito....!!!

**OldSortedSearchStyle**(false) : poiché è stato definito un nuovo modo di eseguire il search ordinato (la ricerca che produca risultati ordinati secondo un certo metodo) se l'opzione non viene abilitata (true) non viene applicato il nuovo metodo. Valori possibili: true/false

Il nuovo metodo di ricerca per ottenere risultati ordinati è il seguente:

Viene selezionato il numero totale di record che soddisfano la richiesta

Se il risultato è maggiore di zero tale risultato viene utilizzato nella restituzione dei risultati stessi (Viene salvato in un'area del Sistema chiamata **TotalRecordToBeSorted**). Se per esempio sono stati trovati 150 documenti da visionare 50 per volta si ha:

50(1,50) di 150 record selezionati .....**Avanti, Tutto**

Se invece fosse stato impostato a true **OldSortedSearchStyle** il risultato sarebbe stato:

50(1,50) di 50 record selezionati .....**Avanti, Tutto**

Naturalmente, in entrambi i casi l'ordinamento sarebbe sempre stato eseguito sui primi 50 e successivi record, ma il primo caso offre direttamente la misura della grandezza

della lista dei risultati.

**PoweredSortedSearch**(false) : se impostato a true permette di ordinare **tutti** i record che soddisfano la richiesta e presentarli così all'utente finale. Il limite di record da ordinare è dato dal valore di **AllObjectToBeDisplayed** che sostituisce il valore di **TotalRecordToBeSorted** se è minore di quest'ultimo.

Se **PoweredSortedSearch** è impostato a **true** sono selezionati e ordinati tutti record per un massimo di **TotalRecordToBeSorted** se inferiore a **AllObjectToBeDisplayed**.

In fase di presentazione vengono inviati all'interfaccia utente i record a partire dal valore del parametro **DsFrom** per **DsFor** record.

Le successive richieste di avanzamento verranno sempre esaudite in modo da avere l'ordinamento sul totale dei record selezionati. L'ordinamento viene, quindi, eseguito volta per volta sul totale degli oggetti. Il metodo è molto utile, ma è time-consuming ; per cui ne andrebbe limitato l'utilizzo a risultati di poche centinaia di record.

Ove si volesse navigare, invece, attraverso migliaia o decine di migliaia di record selezionati è possibile prevedere un ordinamento a priori, secondo un certo metodo, da eseguirsi al momento del caricamento del database. Vedi la procedura:

sortmst.cnr: per ordinare il database del **CNR** in ordine crescente di **titolo** in **<distrib>/unix/IsisDbdir.cnr.tgz**

#### 4.1.0. Parametri Generali per la personalizzazione dell'interfaccia utente

L'interfaccia utente è genericamente costituita dalla seguente schematica struttura:

```
<HeaderMainDefinition (html-string | NOPRINT)>
  <HeaderMainDefinitionContent>
    <SystemJavaScriptFunction>
    <LocalJavascriptFunction>
    <Title Area>
  </HeaderMainDefinitionContent>
  <MessageArea>
  <CommandArea>
  < HeaderMainTable(html- string | NOPRINT)>
    <HeaderQueryTable(html-string | NOPRINT)>
      <FormatText(html- string | NOPRINT)>
      <QueryText(html- string | NOPRINT)>
      <ListInverted (html-string | NOPRINT)>
    <TrailerQueryTable(html-string | NOPRINT)>
  <HeaderDisplaySearchBrowseTable(html- string | NOPRINT)>
    <DisplayedDataBySearchOrBrowse>
  <HeaderDisplayGetTable(html-string | NOPRINT)>
    <DisplayedDataBySearchOrBrowse>
  <TrailerDisplayTable(html-string | NOPRINT)>
  <TrailerMainTable>
```

<TrailerMainDefinition>

**HeaderMainDefinition** (default-string) : definisce la stringa da usare come definizione principale del documento. Il default è contenuto nella classe IsisWriteHeader.java (WriteHeader).

- a) Viene accettato il valore NOPRINT cui corrisponde nessun valore (null).
- b) <SystemJavaScriptFunction> corrisponde ai Javascript di sistema.
- c) <LocalJavascriptFunction> corrisponde al parametro LocalJavascriptFunctions
- d) <Title Area> corrisponde al parametro IsisHeaderArea

**TrailerMainDefinition** (default-string) : definisce la stringa da usare come chiusura della definizione principale del documento. Il default è contenuto nella classe IsisWriteTrailer.java (IsisWriteTrailer). Viene accettato il valore NOPRINT cui corrisponde nessun valore (null).

**HeaderMainTable** (default-string) : definisce la stringa da usare come definizione della tabella (TABLE) principale che organizza tutte le tabelle (TABLE) secondarie. Il default è contenuto nella classe IsisWriteHeader.java(WriteHeaderTable). Viene accettato il valore NOPRINT cui corrisponde nessun valore (null).

**TrailerMainTable** (default-string) : definisce la stringa da usare come chiusura della definizione della tabella (TABLE) principale che organizza tutte le tabelle (TABLE) secondarie. Il default è contenuto nella classe IsisWriteTrailer.java. Viene accettato il valore NOPRINT cui corrisponde nessun valore (null).

**HeaderQueryTable** (default-string) : definisce la stringa da usare come definizione della tabella (TABLE) che definisce la QueryArea. Il default è contenuto nella classe IsisWriteQueryArea.java (IsisWriteQueryArea). Viene accettato il valore NOPRINT cui corrisponde nessun valore (null).

**TrailerQueryTable** (default-string) : definisce la stringa da usare come chiusura della definizione della tabella (TABLE) che definisce la QueryArea. Il default è contenuto nella classe IsisWriteQueryArea.java (IsisWriteQueryArea). Viene accettato il valore NOPRINT cui corrisponde nessun valore (null).

**HeaderDisplaySearchBrowseTable** : (default-string) : definisce la stringa da usare come definizione della tabella (TABLE) che definisce la DisplayArea per l'ambiente browse e search. Il default è contenuto nella classe IsisUtil.java (PrintHeaderDisplayForSearchAndBrowse). Viene accettato il valore NOPRINT cui corrisponde nessun valore (null).

**HeaderDisplayGetTable** (default-string) : definisce la stringa da usare come definizione della tabella (TABLE) che definisce la DisplayArea per l'ambiente get. Il default è contenuto nella classe IsisUtil.java(PrintHeaderDisplayForGet). Viene accettato il valore NOPRINT cui corrisponde nessun valore (null).

**TrailerDisplayTable** (default-string) : definisce la stringa da usare come chiusura della definizione della tabella (TABLE) che definisce la DisplayArea per l'ambiente get, browse e search. Il default è contenuto nella classe IsisUtil.java(PrintTrailerDisplay). Viene accettato il valore NOPRINT cui corrisponde nessun valore (null).

**WindowOpenDefinition** (<a href="javascript:top.AutoWindowOpen(') : definisce la stringa iniziale da usare come comando "href" di apertura di finestra.

**CloseWindowOpenDefinition** (','Doc');\")>) : definisce la stringa finale da usare come comando "href" di apertura di finestra.

## **4.2. Parametri della QueryArea**

**SearchAreaColor**(bgColor=aqua) : definizione della presentazione della SearchArea per la WelcomePage

**SearchAreaWidth**(220) : Ampiezza della SearchArea

**RestoreQueryArea**(on) : attiva il display dei valori immessi nella Query Area: i valori ammessi sono: on/off. Attenzione può essere usato congiuntamente all'operatore Rqar=y/n, da utilizzare nella GET di tipo search, ove si voglia inibire il display dei valori immessi per quella specifica richiesta (transazione).

**QueryText**(Tutti i campi<br><input name=Field0><br>,NOPRINT) : Contiene la descrizione dei campi su cui è possibile eseguire la ricerca. Se specificato NOPRINT non viene generata la sezione search. I campi sono genericamente definiti come Field0,Field1,.....Field31. Il valore semantico del campo è definito dalla sua etichetta esterna(passiva) e interna(attiva).

Nell'esempio contenuto nel valore QueryText del file:

**<istrib>/windows/IsisConf/Ar.sys.file:**

"Tutti i campi<br><input name=Field0><br>Autore<br><input name=Field1><br>..."

L'etichetta esterna(passiva) di Field0 è "Tutti i campi"

L'etichetta esterna(passiva) di Field1 è "Autore"

E' possibile definire l'etichetta interna di ogni campo e le sue modalità di utilizzo. Facendo sempre riferimento a **<istrib>/windows/IsisConf/Ar.sys.file:**

Field1 /(9,10,12,13)

Field2 /(1,11) s (F)

Field3 /(40) s (F)

L'etichetta interna(attiva) di Field1 che semanticamente vale "Autore" è "/(9,10,12,13)".

L'etichetta interna(attiva) di Field2 che semanticamente vale "Titolo" è "/(1,11)".

Nel contesto l'etichetta attiva /(1,11) sta a indicare di cercare il valore associato nei campi (tag) CDS/ISIS 1 e 11.

Le etichette attive sono utilizzate per la costruzione della query da eseguire verso CDS/ISIS. Il formato di definizione della etichetta è:

[<fieldname> <etichetta> [SfxPfx] [operator][LocalTranslateTable]

dove:

- **<fieldname>** è il nome del campo da Field0 a Field31.
- **<etichetta>** è il qualificatore della stringa da ricercare
- **SfxPfx** definisce se l'etichetta va utilizzata come suffisso o prefisso per ogni parola del campo. Esso vale:
  - a. "s" per suffisso e "s\*" per unico suffisso per il campo (il valore default è s).
  - b. "p" per prefisso e "p\*" per unico prefisso per il campo
  - c. "b" sta per suffisso+prefisso per gli elementi e b\* come unico suffisso+prefisso per il campo: la presenza di \* in SfxPfx determina il valore del prefisso e del suffisso, altrimenti sono trattati entrambi come stringa unica sia per il suffisso che per il prefisso.
  - d. "<anychar>" sta per qualunque carattere che se presente in SfxPfx determina il valore del suffisso e del prefisso, altrimenti viene usato come valore unico sia di prefisso che di suffisso.
  - e. Il valore "n" sta per né prefisso né suffisso.
  - f. Il valore "." sta per né prefisso né suffisso

Esempi: data la stringa "f1 f2 f3" se:

- Field1 au= p and . Diviene au=f1 and au=f2 and au=f3
  - Field1 /au s and . Diviene f1/au and f2/au and f3/au
  - Field1 :au: b and . Diviene :au:f1:au: and :au:f2:au: and :au:f2:au:
  - Field1 au= p nop . Diviene au=f1 au=f2 au=f3
  - Field1 au= p\* nop . Diviene au=f1 f2 f3
  - Field1 /au s\* and . Diviene f1 and f2 and f3/au
  - Field1 :au: b nop . Diviene :au:f1:au: :au:f2:au: :au:f3:au:
  - Field1 au:\*/(1) b\* nop . Diviene au:f1 f2 f3/(1)
  - Field1 au:/(1) b\* nop . Diviene au:/(1)f1 f2 f3au:/(1)
  - Field1 au:|/(1) | and . Diviene au:f1/(1) and au:f2/(1) and au:f3/(1)
  - Field1 au:/(1) | and . Diviene au:/(1)f1/au:(1) and au:/(1)f2au:/(1) and au:/(1)f3au:/(1)
- **operator** è l'operatore logico da utilizzare per associare due parole (stringhe) appartenenti allo stesso campo. Il default è "and".
  - **LocalTranslateTable** è la translate table locale al singolo Field. Viene applicata dopo l'applicazione della **IsisSearchTranslateTable**. Il default è null.

Per LocalTranslateTable valgono le stesse regole di **IsisSearchTranslateTable**. Nel caso particolare

Field2 /(1,11) s (F) ;:\_+000

Traduce i caratteri ;:\_ in " "

**ATTENZIONE:** esiste un operatore fittizio “nop” che significa no-operator che se specificato lascia la stringa così come all’origine, cioè non modificata

Field2 /(1,11) s nop ;:\_+000

Ogni campo può contenere più parole(stringhe). Lo spazio contenuto tra ogni parola viene sostituito dal valore default “and” a meno di non specificare gli opportuni operatori accettati: and, \*, or, + ,andnot ,^ ,(G) ,(F) ,\$ ,.(punto).

Nel caso che si voglia trattare il campo come stringa unica con tutti gli spazi compresi il valore va racchiuso tra doppi apici, Es. (“Beppe Romano”).

**FormatText(“”)** : Contiene la definizione dei vari formati che si vogliono utilizzare. Nell’esempio contenuto nel valore FormatText del file **<istrib>/windows/IsisConf/Ar.sys.file:**

```
<br><input type=radio name=Obj value=@Ar.pft>Formato dati lungo<br><input type=radio name=Obj value=@Arsl.pft,SortedBy:@Aro.pft> Formato dati sorted<br>
```

Vengono definiti due formati

1. il formato Ar.pft cui è associato il nome “Formato dati lungo”
2. il formato Arsl.pft che deve essere ordinato secondo le specifiche del formato Aro.pft

Ogni formato di presentazione è costituito da:

```
<printformat>[,SortedBy:<sortformat>]
```

dove:

1. <printformat> è o un formato esplicito nel PrintFormatLanguage di Isis o il nome di un file prefissato da “@” che contiene i comandi del PrintFormatLanguage
2. ,SortedBy: è il delimiter che identifica il formato da utilizzare per l’ordinamento dei risultati della ricerca
3. <sortformat> è o un formato esplicito nel PrintFormatLanguage di Isis o il nome di un file prefissato da “@” che contiene i comandi del PrintFormatLanguage per definire le modalità di ordinamento.

Per maggiori informazioni vedi: **<istrib>/BiremePrograms/wwwisis.rtf** e verifica le modalità di definizione dei formati:

- **<istrib>/windows/IsisConf/Arsl.pft**
- **<istrib>/windows/IsisConf/Arso.pft**

**<istrib>/windows/IsisConf/Arso.pft** contiene:

```
'^t',v1,'^m',mfn
```

**<istrib>/windows/IsisConf/Arsl.pft** contiene:

```

.....ref(['Ar']val(v1^m),'&nbsp;</b><a
href=javascript:top.WindowOpen("/ISISGAS/Isis?
Conf=C:/TEMP/IsisConf/Ar.sys.file&Obj=@Ard.pft&Opt=get&Type=Doc&Id=',mfn,'"Do
c");><i>',v1^a,'</i></a>'),/
.....

```

Viene caricato un tag01 di un database temporaneo (in memoria) con il sottocampo **^t** che contiene il valore del tag01 del database in oggetto (**tag01** è il campo Titolo), e il sottocampo **^m** con il valore del relativo MasterFileNumber. A sort avvenuto viene stampato il sottocampo **^a** del tag01 del database in oggetto tramite una **ref** pilotata dal valore del **mfn** del sottocampo **^m** del **tag01** del database temporaneo

**ListInverted**( <h5>Accesso agli indici</h5><hr><a href= <ServletsDirName>  
<ProgramName>?  
<ConfigurationFile>&Opt=browse&Obj=@Inv.pft&SrcWin=1&Type=Inv&kwt=z&kwf=a>  
Accedere all'Inverted File</a>, NOPRINT) :

1. Contiene le definizioni per l'accesso all'inverted file.
2. Se specificato NOPRINT non viene generato nessun accesso.
3. L'operazione di accesso all'Inverted File è una operazione di browse.
4. Per maggiori dettagli e modalità di uso vedi la definizione di ListInverted in **<distrib>/windows/IsisConf/ArchisisSystemFile-0.0**

**ListaMfn**(<b>Lista degli Mf record</b>, NOPRINT):

1. Contiene la definizione per l'accesso alla lista dei record via Master File Record Number.
2. Se specificato NOPRINT non viene generato nessun accesso.
3. Anche ListaMfn viene eseguita come una operazione di browse.
4. Per maggiori dettagli e modalità di uso vedi l'HTML source generato.

**ButtonStart**(Start) : Valore del bottone ButtonStart per iniziare la ricerca

**ButtonReset**(Reset) : Valore del bottone ButtonReset per ripristinare a NULL (vuoto) il valore dei campi di ricerca

**BeforeSubmitButton**(<hr>) : Testo da prefissare al ButtonStart

**AfterResetButton**(<hr>) : Testo da suffissare al ButtonReset

### **4.3.Parametri della HeaderArea**

**IsisHeaderArea**(<CENTER><h3>Welcome at Isis/Gas system</h3></CENTER>) : Messaggio da presentare nella HeaderArea di tutte le pagine

**FontDefinitionStart**(<FONT size=1>) : definizione del font da utilizzare per tutto il documento

**FontDefinitionEnd**(</FONT>) : chiusura del font definito per tutto il documento

**Title**(Isis is working) : testo da usare come HeadingTitle per la WelcomePage

**GeneralCommonAreaBody**(<BODY aLink=#000000 bgColor=white link=#000000 vLink=#000000>) : definizione generale della presentazione per la WelcomePage

**RecordCommonAreaBody**(<BODY aLink=#000000 bgColor=aqua link=#000000 vLink=#000000>) : definizione generale della presentazione per la RecordPage

#### **4.4. Parametri della DisplayArea**

**DisplayAreaColor**(bgColor=yellow) : definizione della presentazione della DisplayArea per la WelcomePage

**DisplayAreaWidth**(450) : Ampiezza della DisplayArea

**DisplayAreaWidthReduction**(0) : Di quanti pixel ridurre l'area del display all'interno della DisplayArea.

**DisplayAreaWidthAlign**(""): Allineamento orizzontale della DisplayArea. Il default è ALING=left

**GetSearchSyntaxErrorMsg1**(<strong>search syntax error: Operator as first element</strong>) : Primo messaggio di errore di sintassi per il search: un Operatore (and, or, not ...) è il primo elemento di una query o di un componente di una query.

**GetSearchSyntaxErrorMsg2**(<strong>search syntax error: Operator/Operand sequence error</strong>) : Secondo messaggio di errore di sintassi per il search: esistono sequenze di stringhe diverse da Operando Operatore Operando che non sono accettate

**GetSearchSyntaxErrorMsg3**(<strong>search syntax error: No search fields entered</strong>) : Terzo messaggio di errore di sintassi per il search: è stato attivato il search senza inserire stringhe da selezionare. Attenzione: alcune definizioni del **DatabaseSystemFile** possono contenere query "hidden" che bypassano l'errore.

**GetSearchSyntaxErrorMsg0**(<br><br><i>Warning/Error Message:</i><br><br>) : Prefisso introduttivo dei messaggi di errore per il **SearchSyntaxError**

**GetRecordMsg1**( Records sent of ) : Primo messaggio per i record selezionati

**GetRecordMsg2**( selected) : Secondo messaggio per i record selezionati

**GetChildMsg1**( Children sent of ) : Primo messaggio per i record figli selezionati

**GetChildMsg2**( selected) : Secondo messaggio per i record figli selezionati

**LocalAllCommand**(ALL) : definizione del comando All per la DisplayArea

**LocalBackCommand**(BACK) : definizione del comando BACK per la DisplayArea

**LocalMoreCommand**(MORE) : definizione del comando MORE per la DisplayArea

**LineResultAtBottom**(false) : Stampa la linea che contiene la sintesi dei risultati unitamente ai comandi **LocalMoreCommand**, **LocalBackCommand** e **LocalAllCommand** al termine della lista dei risultati e precisamente prima della chiusura della DisplayArea.

**LineResultAtBottomPrefix**(</i>) : definizione del prefisso da utilizzare per la costruzione della linea **LineResultAtBottom**. Se specificato **null** si intende vuoto (blank/"").

**LineResultAtBottomSuffix**(</i>) : definizione del suffisso da utilizzare per la costruzione della linea **LineResultAtBottom**. Se specificato **null** si intende vuoto (blank/"").

**LineResultAtBottomInfix**(blank/"" ) : definizione dell'infixo da utilizzare per la costruzione della linea **LineResultAtBottom**. Se specificato **null** si intende vuoto (blank/""). L'infixo viene posto tra la sintesi dei risultati "100(1,100) record visualizzati di 128 selezionati" e i comandi **LocalMoreCommand**, **LocalBackCommand** e **LocalAllCommand** .

#### **4.5.Parametri della DummyArea**

**DummyAreaColor**(bgColor=yellow) : definizione della presentazione della DummyArea per la WelcomePage

**DummyAreaWidth**(10) : Ampiezza della DummyArea

#### **4.6.Parametri della MessageArea**

**MessageAreaActivated**(on): La Message Area viene utilizzata (vengono inviati i messaggi informativi e di errore). Per disabilitare l'Area messaggio impostare il valore a "off".

**MessageAreaColor**(bgColor="#CCCC99") : definizione della presentazione della MessageArea

#### **4.7.Parametri della CommandArea**

**CommandAreaActivated**(on): La Command Area viene utilizzata (vengono resi disponibili i comandi Help, Back e More). Per disabilitare l'Area dei comandi impostare il valore a "off".

**HelpCommandColor**(bgColor=red) : definizione del colore per il bottone di HELP per la MessageArea

**BackCommandColor**(bgColor=yellow) : definizione del colore per il bottone di BACK per la MessageArea

**MoreCommandColor**(bgColor=aqua) : definizione del colore per il bottone di MORE per la MessageArea

**HelpCommand**(HELP) : definizione del comando HELP per la MessageArea

**BackCommand**(BACK) : definizione del comando BACK per la MessageArea

**MoreCommand**(MORE) : definizione del comando MORE per la MessageArea

## 5. Definizione dei parametri di presentazione dei risultati

La presentazione dei risultati viene fatta utilizzando il file di output prodotto dalle API di accesso alla base di dati (nel caso specifico API Isis: wwwisis, wxis) in formato XML. Per comprendere la struttura dei file XML prodotti si suggerisce di:

1. Abilitare il TRACE con il parametro "TRACE on". In questo modo i workfile non verranno cancellati.
2. Verificare il contenuto del file di input **<WorkDir>/Isis.\*.query\***. Esso contiene vari parametri tra cui il parametro `prolog=`, `epilog=`, `pft=` che sono i Print Format da usare come prologo alla stampa e come epilogo alla stampa; verificarne il contenuto e cercare di capirlo confrontandolo con il contenuto del file **<WorkDir>/Isis.\*.out\***.

La classe **<istrib>/unix/ExtGas.src/IsisWriteResults** converte il contenuto dei file **<WorkDir>/Isis.\*.out\*** nell'insieme dei comandi HTML necessari utilizzando come mappa di conversione il file **IsisFieldTable**

Esempi di specifiche di presentazione dei risultati sono contenuti nella directory **windows/IsisConf** e **unix/cnrConf** nei file **Isis\*FieldFile**.

Genericamente, nei file **Isis\*FieldFile**, ogni field viene identificato dal suo start tag `<tagname>` e dal suo end tag `</tagname>`.

Per ogni valore di `<tagname>` incontrato è possibile definire la stringa associata che si vuole stampare prima del valore del field,.

Per ogni valore di `</tagname>` incontrato è possibile definire la stringa associata da stampare dopo il valore del field.

I field:

**<NumberOfSelectedIndexRecords>**  
**</NumberOfSelectedIndexRecords>**

e

**<NumberOfSelectedRecords>**  
**</NumberOfSelectedRecords>**

Hanno significato particolare poiché sono utilizzati dal sistema per selezionare informazioni riassuntive e necessarie per navigare all'interno delle liste di risultati.

Per comprendere il funzionamento di:

**<NumberOfSelectedIndexRecords>**  
**</NumberOfSelectedIndexRecords>**

si consideri il formato **<istrib>/windows/IsisConf/ArInv.pft:**

```
mhl,if val(v1001) = 1 then '<SubProlog>',/,'<span class=quattroBold>Indice del Database  
Arezzo</span>',/,'</SubProlog>',/,'<Start>',/,'fi,
```

```

if a(v1) then
'<NumberOfSelectedIndexRecords>',v1031,',v1032,',v1002,','</NumberOfSelectedIndexRecords>',else,,
mhl,'<Record>',/,
mhl,'&nbsp;</b><span class=quattro><a href=javascript:top.WindowOpen("<$OurServer$>?
<$OurConf$>&Obj=@ArRTF.pft&Opt=search&Field0=!',V1,!',", "Doc");>',V1,','(,V2,')</a></span>',/,
'</Record>',/,
fi,

```

esso produce il seguente file di risultati:

```

<Prolog>
</Prolog>
<SubProlog>
<span class=quattroBold>Database Index </span>
</SubProlog>
<Start>
<Record>
&nbsp;</b><span class=quattro><a href=javascript:top.WindowOpen("<$OurServer$>?
<$OurConf$>&Obj=@ArRTF.pft&Opt=search&Field0=!AC!', "Doc");>AC(5)</a></span>
</Record>
.....
.....
</Record>
<NumberOfSelectedIndexRecords>
AC :ACCOMPAGNARE :10
</NumberOfSelectedIndexRecords>
</Start>

```

I valori tra <NumberOfSelectedIndexRecords> e </NumberOfSelectedIndexRecords> sono interpretati dal software per produrre le informazioni riassuntive della navigazione sull'**Inverted File** del sistema ospite. Si tenga presente che senza la presenza di tali definizioni verrebbe completamente a mancare la stringa:

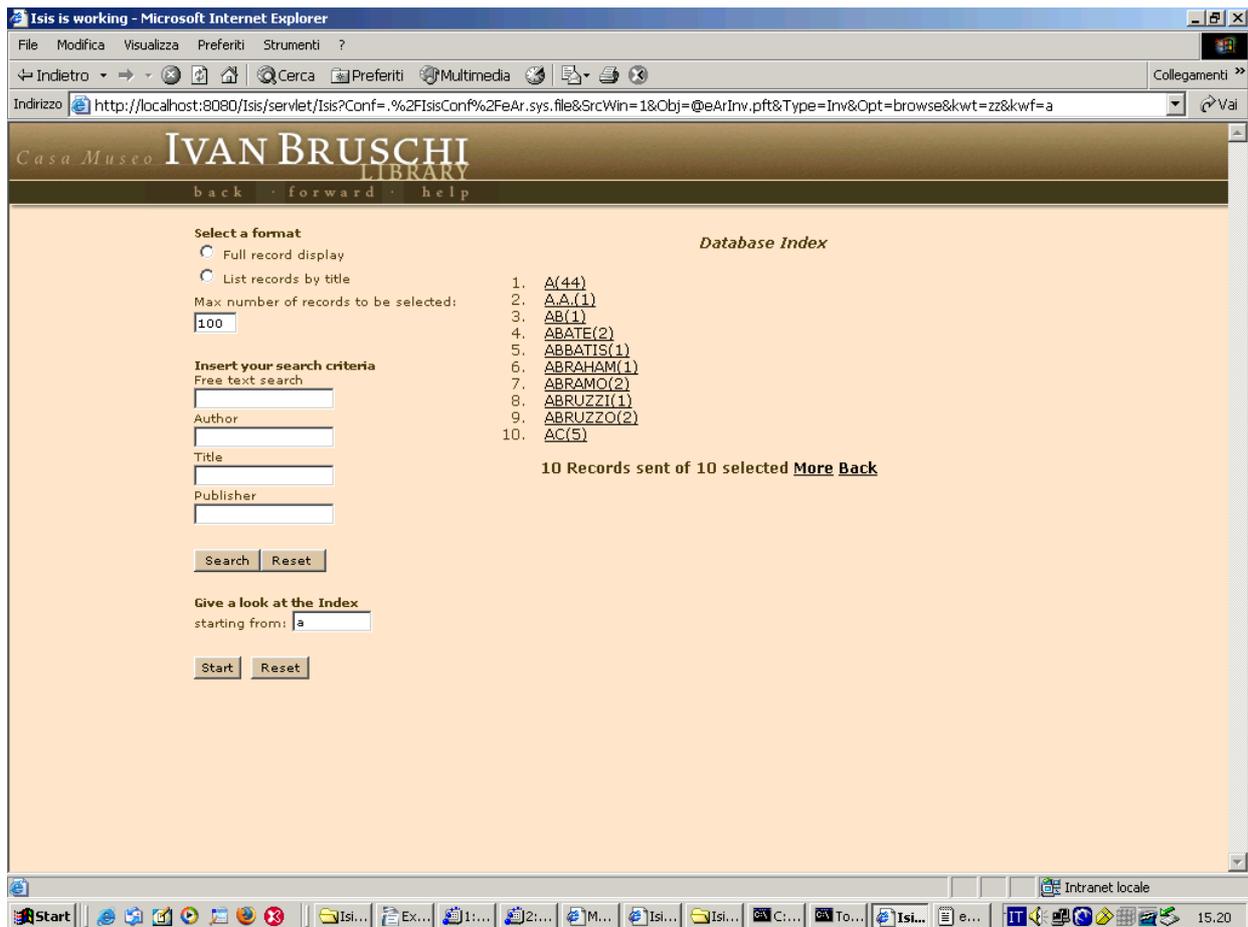
**10 Records sent of 10 selected More Back**

La stringa: "AC :ACCOMPAGNARE :10"

viene costruita, preceduta dal tag <NumberOfSelectedIndexRecords> e seguita dal tag </NumberOfSelectedIndexRecords>, a partire da un record fittizio generato dal sistema al momento in cui raggiunge la chiave più alta specificata per quella richiesta. Attenzione: la stringa di cui sopra è disponibile dopo la lista delle entry.

In questa maniera si ottengono risultati così come mostrato nella pagina successiva sul database del Catalogo dei libri della Collezione Bruschi di Arezzo accessibile alla seguente URL:

<http://museo.fondazionebruschi.it/Isis/servlet/Isis?Conf=C:/Isis/IsisConf/Ar.sys.file> .



Per i tag

**<NumberOfSelectedRecords>**  
**</NumberOfSelectedRecords>**

il meccanismo è simile. Le informazioni riassuntive sono trattate solamente in presenza dei tag in oggetto:

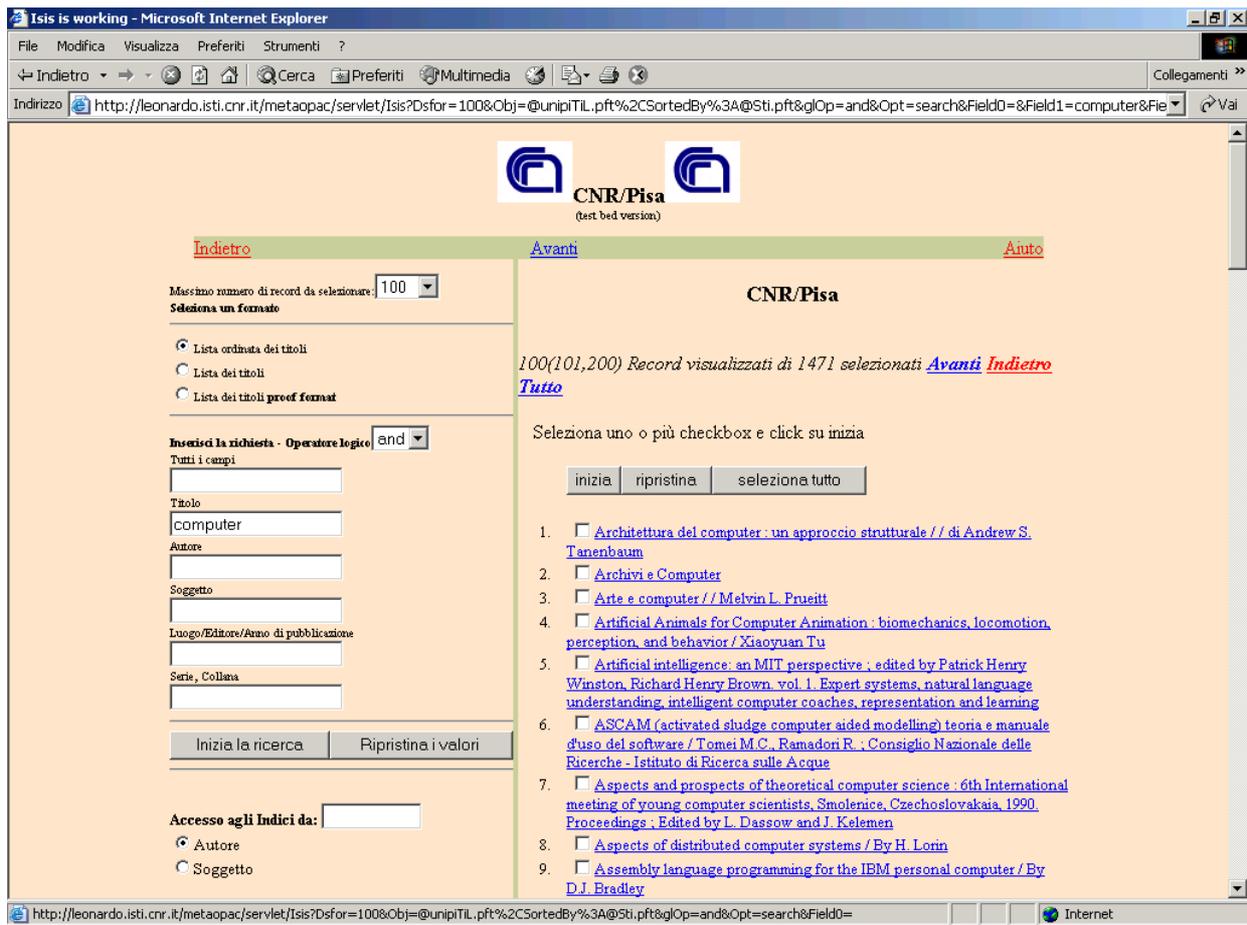
si consideri il formato **<distrib>/unix/cnrConf/unipiTiL.pft** di cui ne viene presentata la parte iniziale:

```
mhl,if val(v1001) = val(v1031) then'<NumberOfSelectedRecords>','/,F((val(v1032)-
val(v1031)+1),3,0),','/,v1002,/,','</NumberOfSelectedRecords>','/,
.....
.....
```

esso produce il seguente file di risultati:

```
<Prolog>
CNR/Pisa
</Prolog>
<NumberOfSelectedRecords>
100:100
</NumberOfSelectedRecords>
<StartForm>
```





Tutti i tag, inclusi i due descritti sopra, possono essere definiti con un prefisso e un suffisso al valore contenuto, oltre alla possibilità di eseguire su di essi alcune funzioni specifiche come per esempio è definito in **<istrib>/unix/cnrConf/Isiscnr-pubFieldFile:**

.....  
 .....

```

<TRTR>
</TRTR> OnlyOneValueHere
<NewCmd>
</NewCmd> <p>
<Start> <FONT size=2><ol>
<Record> <li>
</Record> &nbsp; < $ChangeHrefSpaceToPlus$ >
<DRecord> <br>
</DRecord> &nbsp; < $ChangeHrefSpaceToPlus$ >
</Start> </ol></FONT>
<error-msg> <br>&nbsp;&nbsp;&nbsp;<i>Warning/Error message:</i><strong>&nbsp;&nbsp;&nbsp;
</error-msg> &nbsp;&nbsp;&nbsp;</strong>
<NumberOfSelectedRecords> <i>
</NumberOfSelectedRecords> </i>
<NumberOfSelectedIndexRecords> <br><i>
</NumberOfSelectedIndexRecords> </i>

```



E verrà presentato sul browser come:

---

**valore dell'istanza di yourfield valore dell'istanza di myfield**

---

&nbsp; sta per spazio (cioè lo spazio va specificato con questa notazione).

E' possibile utilizzare alcune notazioni particolari per:

### **Non stampare il contenuto di un campo**

`</myfield> <$NOPRINT>`

### **Stampare il contenuto di un campo convertendone il valore secondo una hashtable caricata da file**

`<ObjectType> <br><FONT face="Comic sans MS" size=5><center>  
</ObjectType> </center></FONT><br><$changevalue C:/Sibar/conf/ObjectType.file>`

Per cui se è:

`<ObjectType>  
Aggregato  
</ObjectType>`

e ObjectType.file fosse caricato come:

.....  
get.Aggregato Complesso Archivistico  
.....  
.....  
search.Produuttore Soggetto Produttore dell'Archivio  
.....

Verrebbe prodotto:

`<br><FONT face="Comic sans MS" size=5><center> Complesso Archivistico  
</center></FONT><br>`

Se invece fosse :

`<ObjectType>  
Produttore  
</ObjectType>`

Sarebbe:

`<br><FONT face="Comic sans MS" size=5><center>Soggetto Produttore  
dell'Archivio</center></FONT><br>`



## 5.1 Utilizzo dell'OpenURL protocol

L'OpenURL è un protocollo che definisce l'interoperabilità tra un fornitore di informazione e un servizio di accesso all'informazione stessa. Il concetto di base prevede che i link all'Internet dovrebbero indirizzare l'utente verso le risorse utili a quest'ultimo. Un servizio (un server), nel momento in cui accetta una richiesta OpenURL in input, offre all'utente un insieme di servizi diversificati che sono stati definiti come uno specifico contesto dell'utente stesso.

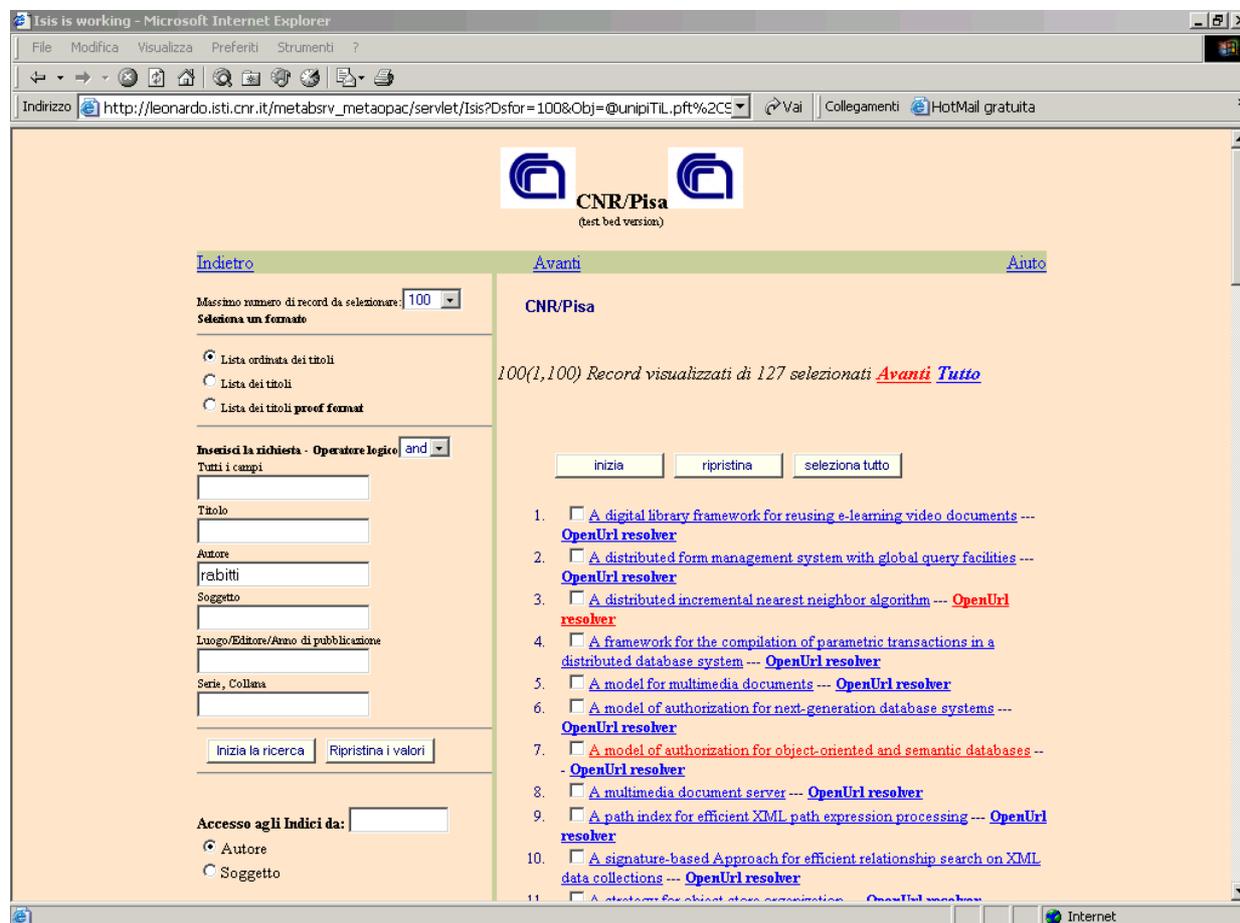
Il protocollo OpenURL permette ad un utente che ha identificato una risorsa informativa di accedere nel modo migliore la risorsa stessa utilizzando una o più modalità di accesso.

Nel contesto del sistema ExtGAS è possibile connettersi a un servizio (OpenURL resolver) inviando in input le informazioni che identificano la risorsa, secondo il protocollo OpenURL.

Gli OpenURL resolver server possono essere selezionati da una lista locale o definiti dinamicamente.

L'accesso all'OpenURL resolver può essere fatto direttamente dalla lista dei risultati selezionati da un database ExtGAS e/o dal display della referenza bibliografica stessa.

Vedi i successivi esempi, data una ricerca per Rabitti/(Autore) nel Catalogo dell'Area del CNR di Pisa, e successivi display.

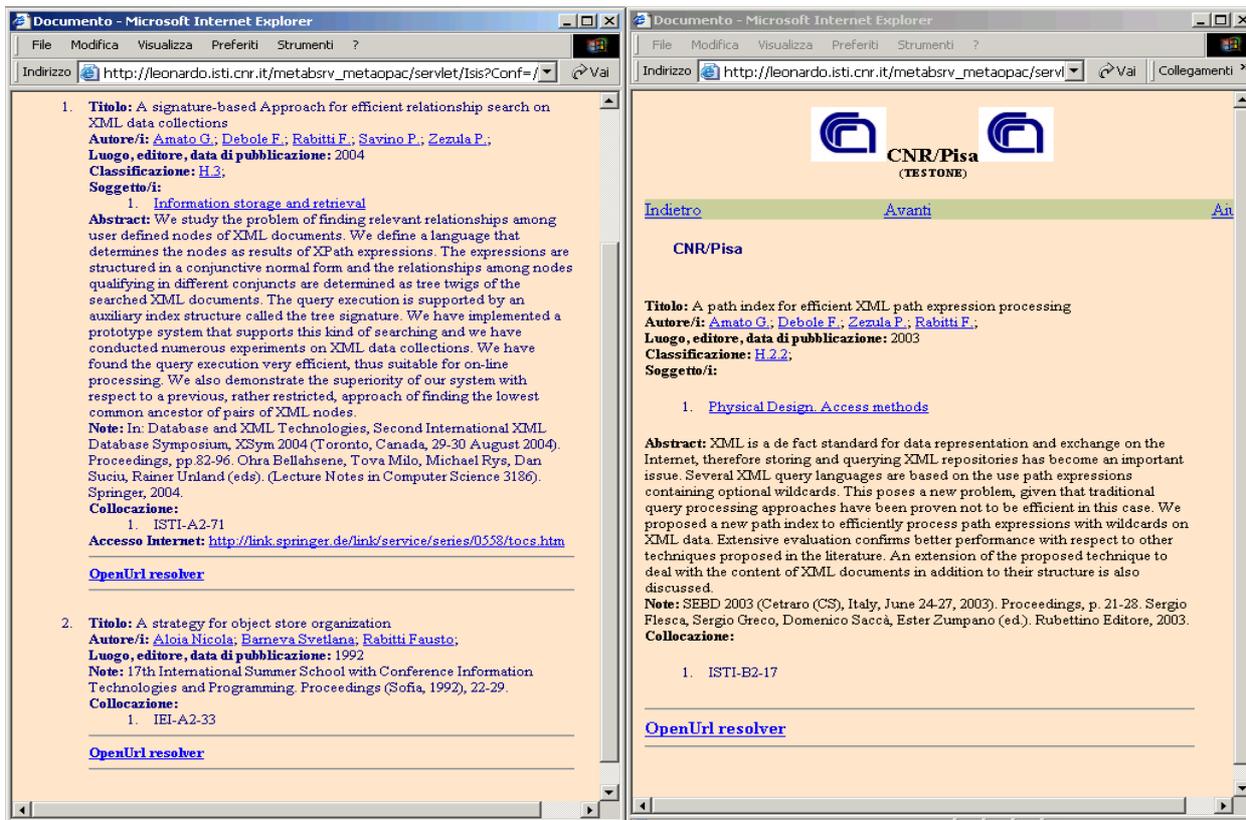


The screenshot shows a web browser window displaying the CNR/Pisa website. The page title is "CNR/Pisa (test bed version)". The search results are displayed in a list format, with the following entries:

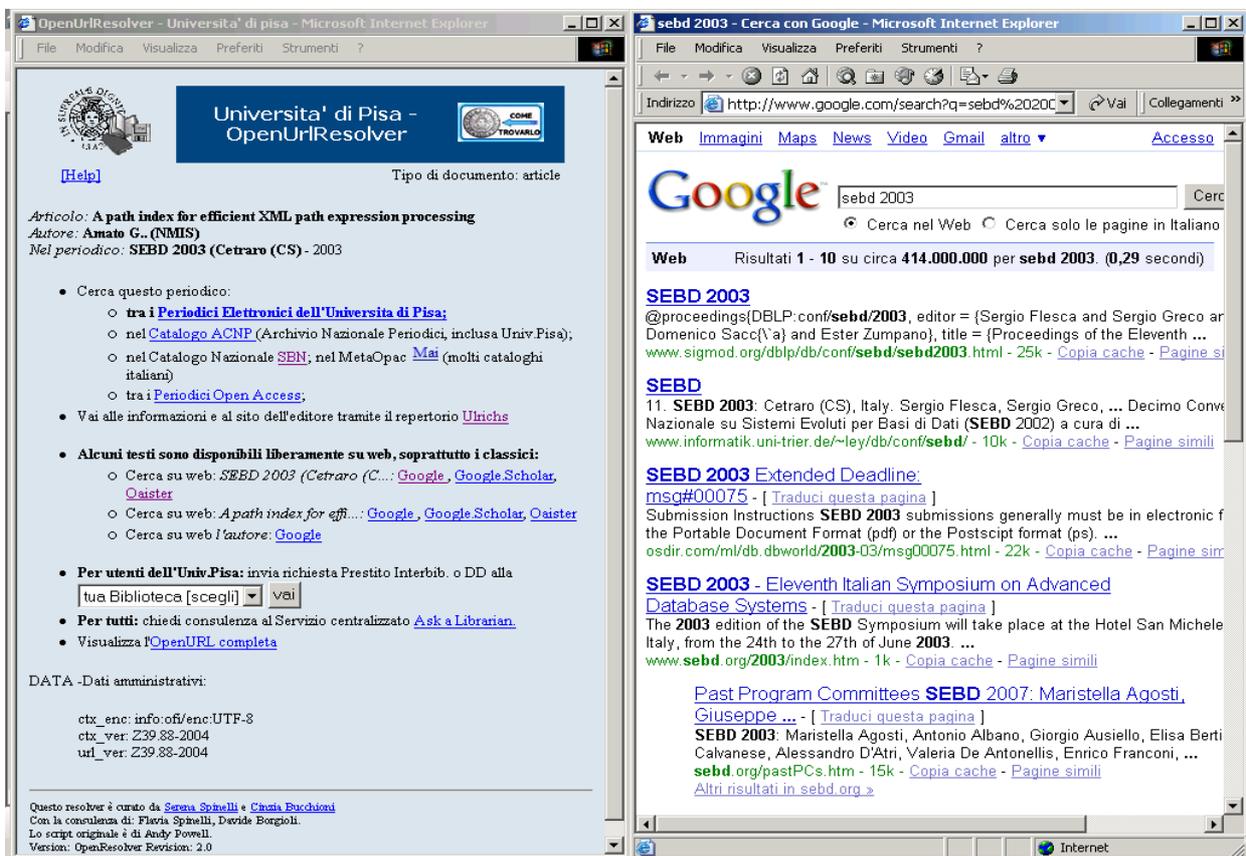
- 1.  [A digital library framework for reusing e-learning video documents --- OpenUrl resolver](#)
- 2.  [A distributed form management system with global query facilities --- OpenUrl resolver](#)
- 3.  [A distributed incremental nearest neighbor algorithm --- OpenUrl resolver](#)
- 4.  [A framework for the compilation of parametric transactions in a distributed database system --- OpenUrl resolver](#)
- 5.  [A model for multimedia documents --- OpenUrl resolver](#)
- 6.  [A model of authorization for next-generation database systems --- OpenUrl resolver](#)
- 7.  [A model of authorization for object-oriented and semantic databases --- OpenUrl resolver](#)
- 8.  [A multimedia document server --- OpenUrl resolver](#)
- 9.  [A path index for efficient XML path expression processing --- OpenUrl resolver](#)
- 10.  [A signature-based Approach for efficient relationship search on XML data collections --- OpenUrl resolver](#)
- 11.  [A system for object-oriented data organization --- OpenUrl resolver](#)

The search criteria are: "rabitti". The search results are displayed in a list format, with the following entries:

A sinistra il display di una lista di record e a destra di uno specifico record:



E cliccando sull'anchor OpenURL resolver della precedente immagine destra:



Si accede alla pagina dell'**OpenURL resolver** selezionato tra una lista di possibili.

Nel caso particolare: <http://resolver.humnet.unipi.it/cgi-bin/openresolver.cgi> che è il resolver dell'Università di Pisa. Il resolver dell'Università di Pisa, accetta i parametri inviati dall'utente e prospetta una serie di possibili accessi informativi. Cliccando sull'accesso google (*SEBD 2003 (Cetraro (C...:)*) si seleziona quanto nell'immagine destra in basso della pagina precedente.

Così come specificato nel paragrafo **4.1.Parametri Generali**, utilizzando i parametri **sfxAllowed** e **sfxAllowedOnSearch** è possibile attivare il supporto OpenURL :

**sfxAllowed (false/true)** default: definisce l'attivazione di un OpenUrl resolver cui passare una richiesta secondo il protocollo OpenUrl: NISO standard Z39.88 (vedi alla seguente URL <http://www.exlibrisgroup.com/category/sfxopenurl> )

**sfxAllowedOnSearch (false/true)** : definisce se l'attivazione dell'OpenUrl resolver http-pointer deve essere eseguita a livello di ogni singolo risultato della lista prodotta dall'opzione **search** e/o semplicemente durante la presentazione di ogni record (opzione **get**)

Nel caso di **sfxAllowed=true** e **sfxAllowedOnSearch=false** viene utilizzato un formato di presentazione, durante l'operazione **get**, che contiene i comandi di Print Format per preparare la chiamata all'OpenURL resolver.

Nel caso, invece, **sfxAllowedOnSearch=true** viene utilizzato un formato di presentazione, durante l'operazione **search**, anche esso contenente i comandi di cui sopra.

Il generico formato da includere nei formati di presentazione già esistenti coerenti con la struttura del database del Catalogo della Biblioteca dell'Area della Ricerca del CNR di Pisa è contenuto nel file **<distrib>/unix/cnrConf.tgz/command\_sfx.pft**. Un esempio di formato utilizzato è **<distrib>/unix/cnrConf.tgz/unipiTiL.pft\_sfx**

```
mhl,if val(v1001) = val(v1031) then'<NumberOfSelectedRecords>',/,F((val(v1032)-
val(v1031)+1),3,0),':',v1002,/,',</NumberOfSelectedRecords>',/,
'<StartForm>',/,',&nbsp;<FORM method=GET name=Boxes
action="javascript:top.GetFromBoxes();">',/,',</StartForm>',/,',<Start>',/,',<NewCmd>',/,',<input
class="button" type=submit value=inizia><input type=reset class="button" value=ripristina><input
class="button" onclick=doCheckAll() type=button value="seleziona tutto">',/,',</NewCmd>',/,fi,
mhl,'<Record>',/,
ref(['cnr']val(v1^m),'&nbsp;<b><input type=checkbox name="Checker" value=',F(val(mfn),0,0),')<a
href=javascript:top.WindowOpen("<$OurServer$>?
<$OurConf$>&Obj=@unipiSh.pft&Opt=get&Type=Doc&Id=',mfn,'"',"Doc");>',',if p(v30) then v30,| / |v40,
else 'No title' fi,'</a>',
,if p(v904) then,
,,,,' --- <a href=javascript:top.WindowOpen("<$OpenUrlResolver$>?',
,,,,'ctx_enc=info:ofi%2Fenc=UTF-8&ctx_tim=2006-2-22T15:11:30CET&ctx_ver=Z39.88-2004',
,,,,'&rft.genre=',if v904='am' OR v904='AM' then 'book' fi,
,,,,if v904='as' OR v904='AS' then 'journal' fi,
,,,,if v904='AP' OR v904='AP' or v904='PU' OR v904='pu' then 'article' fi,
,,,,'&rft_val_fmt=info:ofi%2Ffmt:kev:mtx:book',
,,,,'&url_ctx_fmt=info:ofi:fmt:kev:mtx:ctx&url_ver=Z39.88-2004&rft_id=MOP',
,,,,if p(v30) then,
,,,,,,if v904='AP' OR v904='AP' or v904='PU' OR v904='pu' then,,
,,,,,,if p(v160) then '&rft.title=',,,,,,,if v160:'ln:' then v160*4 fi,
```

```

,,,,,,,,,,,,,'&ft.atitle=',v30,
,,,,,,,,,fi,,,,,,,,,else '&ft.title=',v30,fi,
,,,,fi,
,,,,if p(v70) then '&ft.au=',v70[1] fi,
,,,,if a(v70) AND p(v90) then '&ft.au=',v90[1] fi,
,,,,if a(v70) AND a(v90) AND p(v110) then '&ft.au=',v110[1] fi,
,,,,if p(v120) then '&ft.publisher=',if type(right(v120,4))='N' then left(v120,size(v120)-6) else v120 fi fi,
,,,,if p(v120) AND type(right(v120,4))='N' then '&ft.date=',right(v120,4) fi,
,,,,if p(v170) then if v904='am' OR v904='AM' then '&ft.isbn=' v170 fi,
,,,,,if v904='as' OR v904='AS' then '&ft.issn=' v170 fi,
,,,,fi
,,,,,"", "our");><strong>OpenUrl resolver</strong></a>',
,fi),/,</Record>',/,
if val(v1001) = val(v1032) OR val(v1001) = val(v1002) then '<p><input type=submit value=inizia
class="button"><input type=reset value=ripristina class="button"><input onclick=doCheckAll()
type=button value="seleziona tutto" class="button" ></FORM>' fi,/,

```

I formati coinvolti nell'applicazione sono:

```

cnr1TiL.pft_sfx
unipiRTFbox.pft_sfx
unipiRTF.pft_sfx
unipiSh.pft_sfx
unipiTiL0.pft_sfx
unipiTiL.pft_sfx

```

e sono tutti disponibili nella directory **<distrib>/unix/cnrConf.tgz**.

Alla

URL:[http://leonardo.isti.cnr.it/metabsrv\\_metaopac/servlet/Isis?Conf=/export/home/metaopac/mpisa/cnrConf/cnr.sys\\_metabsrv.file](http://leonardo.isti.cnr.it/metabsrv_metaopac/servlet/Isis?Conf=/export/home/metaopac/mpisa/cnrConf/cnr.sys_metabsrv.file)

E' disponibile una applicazione OpenURL protocol come supporto al database del Catalogo del CNR di Pisa.

Nella stampa del precedente formato è contenuto il metodo GET per l'attivazione dell'OpneURL resolver selezionato con tutti i parametri specifici da derivare dalla referenza bibliografica e inviare al server.

La stringa:

```
<a href=javascript:top.WindowOpen("<$OpenUrlResolver$>?'
```

contiene il parametro **<\$OpenUrlResolver\$>** che al run-time conterrà l'indirizzo e il servizio dell'OpenURL resolver server selezionato (nel nostro esempio esso è <http://resolver.humnet.unipi.it/cgi-bin/openresolver.cgi>).

All'inizio della sessione l'OpenURL resolver support non è impostato. L'impostazione dell'OpenURL resolver viene eseguita mediante il servlet **ExtGASsfxLCP** che verifica l'esistenza di un cookie, sulla pagina dell'utente web, di nome **user-OpenURL** che contiene l'indirizzo di un OpenURL resolver. Nel caso non ce ne fosse uno impostato è possibile selezionarlo da una lista e attivarlo. Il servlet **ExtGASsfxLCP** viene attivato nella seguente maniera:

[http://leonardo.isti.cnr.it/metabsrv\\_metaopac/servlet/ExtGASsfxLCP?](http://leonardo.isti.cnr.it/metabsrv_metaopac/servlet/ExtGASsfxLCP?)

CPLfile=/export/home/metaopac/mpisa/Conf/resolver.html&defaultResolver=http://resolver.humnet.unipi.it/cgi-bin/openresolver.cgi&MaxAge=-1

Dove:

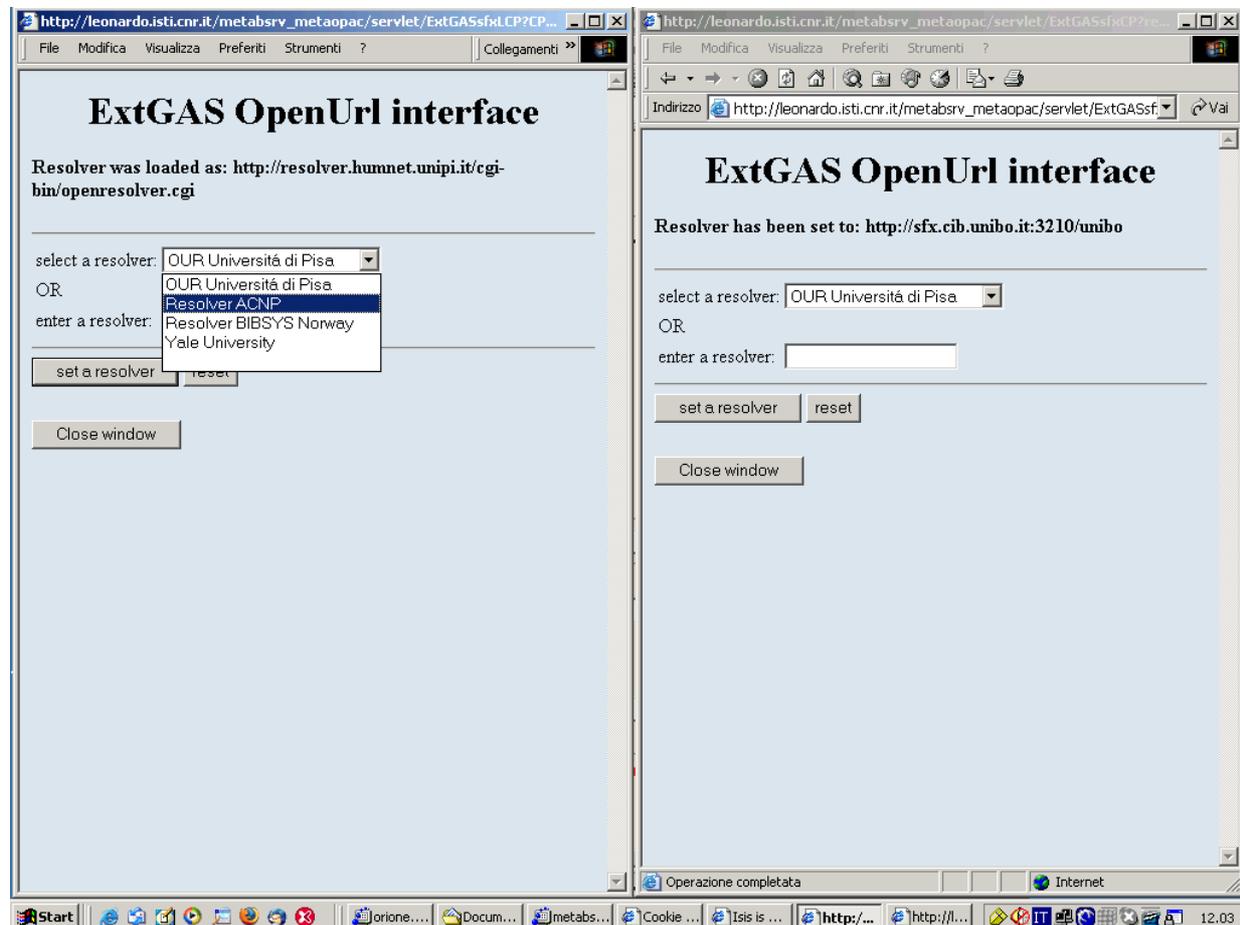
**CPLfile** è il file html (**resolver.html**) che contiene le specifiche di presentazione della transazione. Il file **resolver.html** può essere modificato tenendo conto delle sue caratteristiche. Fondamentalmente è possibile aggiungere o eliminare OpenURL resolver server dalla lista di quelli disponibili.

**defaultResolver** è l'eventuale OpenURL resolver da impostare automaticamente (se il parametro non fosse passato il valore default sarebbe impostato a "", cioè vuoto).

**MaxAge** definisce il time to live del cookie che verrà impostato (-1 significa il default del web browser dell'utente: cancellare al termine della sessione); MaxAge è espressa in secondi.

Nel caso si volesse impostare un OpenURL resolver diverso da quello già impostato o impostarlo per la prima volta viene attivato il servlet **ExtGASsfxCP** che esegue l'operazione di **cookie-pusher**. **ExtGASsfxCP** accetta gli stessi parametri di impostazione di **ExtGASsfxLCP**.

Nell'immagine successiva il lato sinistro ivia informazioni sul OpenURL resolver impostato e gli altri server disponibili; il lato destro imposta l'OpenURL resolver di ACNP



## 6. Il file di distribuzione del software

Il distribution file contiene:

1. La directory **windows** contenente esempi, dati e procedure di installazione.
2. La directory **BiremePrograms.zip** che contiene le utility Isis scritte da BIREME
3. La directory **BiremePrograms\_4\_3\_a.zip** che contiene le utility Isis, aggiornate, scritte da BIREME
4. La directory **BiremePrograms\_wxis.zip** che contiene il software wxis che è la API versione 5 di wwwisis
5. La Directory **unix** che contiene una applicazione unix completa in ambiente Unix Linux e cui si fa riferimento nel manuale,
6. La Directory **Doc** che contiene i file pdf di documentazione.
7. La Directory **ExtGas.src** che contiene l'ultima versione del software.
  - a. Il file **ExtGAS.src.new\_4.tgz** che è il sistema ExtGAS (**ExtGAS.src**)
  - b. Il file **ExtGASConfGenerator.tgz** che è il generatore di applicazioni ExtGAS
  - c. Il file **ExtGASsfx.src.tgz** che è il supporto OpenURL per ExtGAS
  - d. Il file **ExtGASoai.src.tgz** che è il server OAI per il sistema CDS/ISIS, e che viene usato in congiunzione con ExtGAS
8. La Directory **JavaPGM** che contiene software di conversione per produrre file CDS/ISIS caricabile (iso2709 format).
9. Il file **READMEFIRST.txt** che contiene un dettaglio del contenuto del distribution file
10. La directory **wwwisis-fedora** che contiene le librerie e l'install file per rendere allineato il software Bireme & CISIS al sistema operativo ospite per Linux Fedora.

Nel seguito descriveremo il contenuto della Directory **IsisGas.src. IsisGas.src** contiene i file .class e .java delle classi implementate

- **AccessIsis** è la classe che implementa la chiamata alle API Isis wwwisis versione 3
- **AccessIsisViaWxis** è la classe che implementa la chiamata alle API Isis wwwisis versione 5 o wxis
- **aixload, aixexload, aixexloadiso, aixexloadisoplus** sono utility per generare gli indici alternativi per l'accesso alle informazioni contenute nei database. Per maggiori dettagli sull'utilizzo vedi: ([Il MetaOpac Pisano: l'organizzazione logica e il mapping fisico - ISTI-B4-02, 2003](#))
- **Base64Converter** è la classe che converte stringhe di caratteri in caratteri base64: vedi RFC 1421
- **ExtGASConfGenerator** è la classe che viene usata come generatore di applicazioni ExtGAS (vedi [<istrib>/ExtGAS.src/ExtGASConfGenerator.tgz](#))
- **ExtGASgcrun** è la classe che esegue il Garbage Collector e il Run Finalization per poter riutilizzare le risorse rilasciate o che sono ancora pending.
- **ExtGASthread** è la classe che permette l'accesso e la selezione di oggetti appartenenti a Database Systems non CDS/ISIS. Per maggior dettagli vedi: ([ExtGAS: manuale di installazione e d'usoExtGAS:](#)

[manuale di installazione e d'uso ISTI-B4-16, 2005\)](#)

- **ExtMethod** è la classe che permette di visualizzare i documenti bibliografici appartenenti a Database Systems non CDS/ISIS. Per maggior dettagli vedi: ( [ExtGAS: manuale di installazione e d'uso](#)[ExtGAS: manuale di installazione e d'uso ISTI-B4-16, 2005\)](#))
  - **GASBroadcast** è la classe che attiva l'accesso parallelo ai database remoti
  - **GAShread** è la classe che implementa l'accesso al database remoto (il suo source code è contenuto in GASBroadcast.java)
  - **genproof** utility per generare un CDS/ISIS print-format che possa fare il display del contenuto reale dei record.
  - **getdataviaurl** utility per copiare via IP un file che è memorizzato su altro computer via http protocol
  - **getsteamviaurl** utility per copiare via IP un file che è memorizzato su altro computer via http protocol
  - **Isis** è la classe driver che costituisce il servlet Isis
  - **IsisGetParms** è la classe che esegue il parser dei parametri del servlet
  - **IsisGet** è la classe che implementa la funzione GET
  - **IsisXMLGet** è la classe che implementa la funzione GET nel caso di WWW browser di tipo XML e di restituzione grafica richiesta.
  - **IsisBrowse** è la classe che implementa la funzione BROWSE
  - **IsisSearch** è la classe che implementa la funzione SEARCH
  - **IsisSysTable** è la classe che imposta i parametri del sistema e li legge dal file di definizione
  - **IsisUtil** è una utility class
  - **IsisWriteDisplay** è la classe che costruisce la Display Area dei Panel
  - **IsisWriteDummyArea** è la classe che costruisce una Area Dummy del Panel delegata a futuro uso
  - **IsisWriteErrorPanel** è la classe delegata a inviare l'Error Panel
  - **IsisWriteHeader** è la classe che costruisce gli Header dei Panel
  - **IsisWriteTrailer** è la classe che costruisce i Trailer dei Panel
  - **IsisWriteWelcomePanel** è la classe che costruisce il Welcome Panel
  - **IsisWriteQueryArea** è la classe che costruisce la Query Area (SEARCH e BROWSE functions)
  - **IsisWriteResults** è la classe che costruisce l'HTML dall'XML file, a sua volta prodotto, come output, dalle API Isis.
  - **IsisWriteXMLResults** è la classe che costruisce dall'XML file di output delle API Isis l'XML adatto alla presentazione e alla costruzione scalare della rappresentazione grafica delle struttura degli archivi
  - **isoTranslateTable** è una utility che permette di preparare file iso2709 che contengono i caratteri da convertire durante le operazioni di stampa, indicizzazione e caricamento dei dati che contengono caratteri diacritici. Vedi file **<istrib>unix/gizmodb.tgz** e **<istrib>/BiremePrograms/wwwisis.rtf**
  - **ItlsWorking** è la classe di test per l'Apache Jserv con cui verificare il funzionamento del server e dell'interfaccia servlet/httpd
  - **renamedateFile** è una utility per modificare il nome di un file introducendo nel nome stesso una data.
  - **TestServer** è una utility per testare le funzionalità di uno o più server
- ExtGAS**

## 7.Funzionamento del sistema

Il sistema viene attivato tramite i metodi GET delle varie opzioni specificate sopra nel **Capitolo 1**. Ogni richiesta costruisce un file di input **<WorkDir>/Isis.\*.query\*** che contiene i parametri impostati per l'API Isis (wwwi32.exe, wwwisis). L'API Isis processa i parametri e produce uno o più file di out di nome **<WorkDir>/Isis.\*.out\***. Di seguito sono presentati i contenuti dei file **<WorkDir>/Isis.\*.query\*** a seconda di alcune richieste:

1. Search sul database **<istrib>/windows/IsisDbdir/Arezzo** per la parola "dante" nel campo "autore", viene richiesto l'ordinamento per Titolo. Contenuto del file di input:

```
db=Ar
bool=( dante/(9,10,12,13) )
gizmo=html
h1=1
h2=100
freq=@Aro.pft
prolog=@prolog.Ar
epilog=@epilog.Ar
pft=@Arsl.pft
cipar=C:/TEMP/IsisConf/Ar.txt
out=C:/TEMP/Isis.1010770723667.0.out
```

2. Browse sul database **<istrib>/windows/IsisDbdir/Arezzo** per accedere l'Inverted File. Contenuto del file di input:

```
db=Ar
prolog=@prolog.Ar
epilog=@epilog.Ar
gizmo=html
k1=a
k2=zz
count=200
pft=@ArInv.pft
cipar=C:/TEMP/IsisConf/Ar.txt
out=C:/TEMP/Isis.1010770806857.0.out
```

3. Browse sul database **<istrib>/windows/IsisDbdir/Arezzo** per accedere i record del database via Master File Record Number. Contenuto del file di input:

```
db=Ar
prolog=@prolog.Ar
epilog=@epilog.Ar
gizmo=html
from=1
count=100
pft=@Ar.pft
cipar=C:/TEMP/IsisConf/Ar.txt
out=C:/TEMP/Isis.1010771150811.0.out
```

4. Browse sul database **<istrib>/windows/IsisDbdir/Archi** per accedere l'Inverted File come indice dei Toponimi (il prefisso per il toponimo è t=).  
Contenuto del file di input:

```
db=Archi
prolog=@prolog.Arch
epilog=@epilog.Arch
gizmo=html
k1=t=a
k2=t=zz
count=20
pft=@ArchiT.pft
cipar=C:/TEMP/IsisConf/Archi.txt
out=C:/TEMP/Isis.1010771547422.0.out
```

5. Search sul database **<istrib>/unix/IsisDbdir.cnr** per selezionare nel sottoinsieme "Pubblicazioni ISTI" (database **cnr-pub**, come sottoinsieme di **cnr**, descritto nel **DatabaseSystemFile cnr-pub-list.sys.file\_x** definito in **<istrib>/unix/isti\_piConf**) le pubblicazioni dell'anno 2006 in modo che vengano prodotte delle liste ordinate per tipo, per ordine decrescente di anno e ordinate alfabeticamente per titolo. Contenuto dei file di input per le query:

Prima query (file /tmp/Isis.1170262116283.0.query.src.query) per accertare il numero degli oggetti esistenti con quelle caratteristiche:

**OldSortedSearchStyle=false**

```
db=cnr-pub
bool=( 2006/(120) ) AND ( IEI/(900) or CNU$/(900) or ISTI/(900) ) AND ( AP/
(904) )
h1=1
h2=1
prolog=@prolog.cnr-pub
epilog=@epilog.cnr-pub
pft=mhl,v1002,/
cipar=/export/home/metaopac/mpisa/isti_piConf/cnr.txt
out=/tmp/Isis.1170262116283.0.query.src.out
```

Seconda query (file /tmp/Isis.1170262116283.0.query) per selezionare e stampare gli oggetti selezionati:

```
db=cnr-pub
bool=( 2006/(120) ) AND ( IEI/(900) or CNU$/(900) or ISTI/(900) ) AND ( AP/
(904) )
gizmo=html
h1=1
h2=261
freq=@cnrSSty_r_yetiJCR.pft
prolog=@prolog.cnr-pub
epilog=@epilog.cnr-pub
pft=@cnr-pListtibibnyJCR.pft
```

cipar=/export/home/metaopac/mpisa/isti\_piConf/cnr.txt  
out=/tmp/Isis.1170262116283.0.out

si considerino le seguenti definizioni nel cipar file  
<distrib>/unix/isti\_piConf/cnr.txt del database cnr-pub

cnr.\*=/export/home/metaopac/mpisa/IsisDbdir.cnr/cnr.\*  
cnr-pub.\*=/export/home/metaopac/mpisa/IsisDbdir.cnr/cnr.\*

esse puntano al database completo <distrib>/unix/IsisDbdir.cnr

L'ordinamento viene definito dal formato:  
<distrib>/unix/cnrConf/cnrSSty\_r\_yetiJCR.pft

e la stampa prodotta dal formato:  
<distrib>/unix/cnrConf/cnr-pListtibibnyJCR.pft

Entrambi i formati non sono sulla directory <distrib>/unix/isti\_piConf come si evince dal cipar file <distrib>/unix/isti\_piConf/cnr.txt mediante le seguenti definizioni

cnr-pListtibibnyJCR.pft=/export/home/metaopac/mpisa/cnrConf/cnr-pListtibibnyJCR.pft  
cnrSSty\_r\_yetiJCR.pft=/export/home/metaopac/mpisa/cnrConf/cnrSSty\_r\_yetiJCR.pft

La selezione viene effettuata sul sottoinsieme "ISTI or CNUCE or IEI" tramite le seguenti definizioni nel **DatabaseSystemFile cnr-pub-list.sys.file\_x** definito in <distrib>/unix/isti\_piConf.

QueryText <p class="menu\_query">.....<br><input type=hidden name=Field5 value="IEI CNU\$ ISTI"><input.....

Field5 /(900) s or

Nella definizione dei formati di stampa \*.pft file e nella definizione dei parametri dei file di configurazione (**ConfigurationFile**), è possibile utilizzare alcune notazioni per rendere dinamica la costruzione degli http pointer che referenziano il servlet **ExtGAS** stesso.

- ❑ La stringa <\$OurServer\$> rappresenta il nome del server e il servlet che lo attiva:

http://leonardo.isti.cnr.it/metaopac/servlet/ExtGAS

- ❑ La stringa <\$OurConf\$> rappresenta il file di configurazione all'interno del server che viene utilizzato dal relativo servlet **Isis/ExtGAS**, prefissata dalla identificazione del file di configurazione

Conf=C:/Isis/IsisConf/Ar.sys.file

- ❑ La stringa **<\$OurConfNold\$>** rappresenta il file di configurazione all'interno del server che viene utilizzato dal relativo servlet **Isis/ExtGAS**

C:/Isis/IsisConf/Ar.sys.file

- ❑ Il carattere **!** che viene utilizzato al posto del carattere “

Dare una guardata a:

**<istrib>/windows/IsisConf/Arsl.pft,  
<istrib>/windows/IsisConf/ArInv.pft,  
<istrib>/windows/IsisConf/Ar.sys.file,  
<istrib>/windows/IsisConf/Luzz.sys.file**

## **8.Installazione di un generico database**

1. Costruire il cipar file usando come guida il file **<istrib>/unix/cnrConf/cnr.txt**
  - Definire i pft file per la presentazione. Attenzione, di suggerisce di usare, almeno la prima volta pft file elementari: gli errori di formato non sono facilmente identificabile e la API Isis alcune volte va in crash di sistema.
  - Definire i prolog e epilog print format file
  - Verificare l'usabilità del gizmo file: se non è sufficiente aggiornarlo.
2. Costruire il Configuration file relativo al database in oggetto. E' consigliabile usare un configuration file già definito, tipo il file **<istrib>/unix/IsisConf/cnr.sys.file** e andare a modificare uno per uno i parametri.
3. Attivare il servlet così come specificato
4. Se si verificano errori: abilitare il TRACE (TRACE on nel Configuration File) cercare di capire cosa accade e aggiornare le definizioni.
5. Tenere sempre presente che il database Ar e il database cnr, rispettivamente su Windows e su Unix/Linux sono funzionanti.

## Bibliografia

1. Giuseppe A. Romano - Isis/GAS : un sistema multi piattaforma per l'accesso a basi di dati testuali via Internet - ISTI-B4-28, 2003 - URL: <http://leonardo.isti.cnr.it/metaopac/IsisGas.pdf>
2. Giuseppe A. Romano - Isis/GAS : manuale di installazione e d'uso - ISTI-B4-31, 2003 - URL: <http://leonardo.isti.cnr.it/metaopac/IsisGasInstall.pdf>
3. Giuseppe A. Romano - Il MetaOpac Pisano: l'organizzazione logica e il mapping fisico - ISTI-B4-02, 2003 - URL: <http://leonardo.isti.cnr.it/metaopac/MetaOpacLogic.pdf>
4. Giuseppe A. Romano - Il MetaOpac Pisano: descrizione tecnica e procedure di recovery - ISTI-B4-30, 2003 - URL: <http://leonardo.isti.cnr.it/metaopac/MetaopacPisano.pdf>
5. Giuseppe A. Romano - Il metaopac: controllo e recovery del sistema - ISTI-B4-07, 2004 - URL: [http://leonardo.isti.cnr.it/metaopac/IsisGAS\\_BriefMan.pdf](http://leonardo.isti.cnr.it/metaopac/IsisGAS_BriefMan.pdf)
6. Biagioni S., Giannini S., Giuseppe A. Romano - Il Metaopac del CNR di Pisa e la sua collocazione nel Metaopac pisano - ISTI-B4-14, 2004 - URL: <http://leonardo.isti.cnr.it/metaopac/OpacCNR.pdf>
7. Pierlugi Niccolai, Enrico Venturini - TECA: Applicazione di CDS/ISIS per i dati bibliografici - Titivillus , Firenze 1993
8. Stefania Biagioni, Carlo Carlesi, Daniela Musa - Sistema Atlas: catalogazione derivata da registrazioni in formato UNIMARC - IEI-B4-30 : Istituto di Elaborazione della Informazione, settembre 1994
9. CDS/ISIS - software, tools, descriptions – URL: [http://portal.unesco.org/ci/ev.php?URL\\_ID=2071&URL\\_DO=DO\\_TOPIC&URL\\_SECTION=201&reload=1082372136&PHPSESSID=7462f4450cf4d320877976df1ae94292](http://portal.unesco.org/ci/ev.php?URL_ID=2071&URL_DO=DO_TOPIC&URL_SECTION=201&reload=1082372136&PHPSESSID=7462f4450cf4d320877976df1ae94292)
10. BIREME - CISIS reference manual – URL: <http://productos.bvsalud.org/html/en/home.html>
11. BIREME - wwwisis version 3.0 - URL: <http://productos.bvsalud.org/html/en/home.html>
12. LIBERO - software – URL: <http://www.infologic.it/ita/libero.htm>
13. Antonio Scolari - Unimarc, Roma, AIB, 2000 Enciclopedia Tascabile ETIS
14. Giuseppe A. Romano – Il MetaOpacPisano : organizzazione, controllo e recovery del sistema per l'anno – ISITI-B4-17,2005 URL: [http://leonardo.isti.cnr.it/metaopac/MOP\\_BriefMan.pdf](http://leonardo.isti.cnr.it/metaopac/MOP_BriefMan.pdf)
15. Giuseppe A. Romano – ExtGAS un sistema multipiattaforma per l'accesso ai cataloghi di sistemi biblioteconomici eterogenei – ISTI-B4-15,2005 – URL: <http://leonardo.isti.cnr.it/metaopac/ExtGas.pdf>
16. Giuseppe A. Romano – ExtGAS manuale di installazione e d'uso – ISTI-B4-16,2005 – URL: <http://leonardo.isti.cnr.it/metaopac/ExtGasInstall.pdf>