ENTREZ DIRECT - QUICK TOUR

Entrez Direct (EDirect) is an advanced method for accessing NCBI's interconnected data domains (publication, nucleotide, protein, structure, variation, expression, etc.) from a terminal window. Multi-step queries can be built incrementally, and functions take search terms from command-line arguments.

All that is necessary for an Entrez search is the name of the database and the query string. For example:

```
  esearch -db pubmed -query "capsaicin cancer pain management"
```

constructs the appropriate Entrez Utilities (EUtils) URL from the query terms and executes the search.

Terms can be qualified by an indexed field abbreviation (e.g., [AUTH], [JOUR], [MESH], [TITL]). Boolean operators and parentheses can be used in the query expression for more complex searches. For example:

```
  "Tager H [AUTH] AND glucagon [TIAB]"
```

in pubmed or:

```
  "alcohol dehydrogenase [PROT] NOT (bacteria [ORGN] OR fungi [ORGN])"
```

in protein.

The vertical bar ("|") UNIX pipe character is used to send the results to the next step. This example is a request to look up related articles (precomputed PubMed neighbors) of the initial results:

```
  esearch -db pubmed -query "capsaicin cancer pain management" | elink -related
```

(The elink -target argument retrieves associated records between databases.)

The same set of commands can be written on multiple lines with the backslash ("\") UNIX line continuation character, for improved readability and easier editing:

```
  esearch -db pubmed -query "capsaicin cancer pain management" | \
  elink -related
```

The list of neighbors can be refined by further term searching in Entrez:

```
  esearch -db pubmed -query "capsaicin cancer pain management" | \
  elink -related | \
  efilter -query "conotoxin NOT morphine"
```

A small structured XML object is passed between each command. The Count field gives the number of records returned by the previous step. A good measure of query success is a reasonable (non-zero) count value:

```
  <ENTREZ_DIRECT>
    <Db>pubmed</Db>
    <WebEnv>NCID_1_4586177_172.16.22.25_5555_1366675055_388385336</WebEnv>
    <QueryKey>4</QueryKey>
    <Count>10</Count>
    <Step>3</Step>
  </ENTREZ_DIRECT>
```

Checking the result count at each step can help avoid unsuccessful queries.

Record retrieval or formatting is separate from searching. Piping to:

```
  efetch -format abstract
```

or:

```
  efetch -format medline
```

returns traditional formats that can be read by a person or input to bibliographic database programs, respectively.

EDirect also provides document summaries and other result forms that are returned in structured XML format. For example:

```
  esearch -db pubmed -query "capsaicin cancer pain management" | \
  elink -related | \
  efilter -query "conotoxin NOT morphine" | \
  esummary
```

will generate an XML document summary set:

```
  <eSummaryResult>
    <DocumentSummarySet status="OK">
      <DocumentSummary uid="23536870">
        <Id>23536870</Id>
        <PubDate>2013</PubDate>
        <EPubDate>2013 Mar 25</EPubDate>
        <Source>PLoS One</Source>
        <Authors>
          <Author>
            <Name>Sousa SR</Name>
            <AuthType>Author</AuthType>
            <ClusterID>0</ClusterID>
          </Author>
          <Author>
          ...
```

The advantage of XML is that many pieces of information are in specific locations in a well-defined data hierarchy. Assembling individual units of data that are fielded by name, such as:

```
  <Source>PLoS One</Source>
  ...
  <Volume>8</Volume>
  <Issue>3</Issue>
  <Pages>e59293</Pages>
```

is much easier than parsing the units from a long, complex string:

```
  1. PLoS One. 2013;8(3):e59293 ...
```

The xtract function uses command-line arguments to direct the selective conversion of XML data into a tab-delimited table. The -pattern argument divides the results into rows, while placement of data into columns is controlled by -element.

Additional arguments can limit data extraction to specified regions of the XML, filter by data content, and customize the table presentation. These will be discussed in the examples that follow.

Piping a document summary set to:

```
  xtract -outline
```

will give a simplified overview of XML structure hierarchy:

```
  DocumentSummarySet
    DocumentSummary
      Id
      PubDate
      EPubDate
      Source
      Authors
        Author
          Name
          AuthType
          ClusterID
        Author
        ...
      LastAuthor
      Title
      SortTitle
      Volume
      Issue
      Pages
      Lang
        string
      NlmUniqueID
      ISSN
      ESSN
      PubType
        flag
      RecordStatus
      ...
```

The outline can help in deciding what arguments to send to xtract, so:

```
  esearch -db pubmed -query "capsaicin cancer pain management" | \
  elink -related | \
  efilter -query "conotoxin NOT morphine" | \
  esummary | \
  xtract -pattern DocumentSummary -element Id SortFirstAuthor Title
```

returns the PubMed identifier, first author name, and article title:

```
  23536870  Sousa SR  Expression and Pharmacology of Endogenous Cav Channels ...
  22410003  Vetter I  Characterisation of Na(v) types endogenously expressed ...
  18956616  FÃ¼rst Z  [Central and peripheral mechanisms in antinociception: ...
  12566085  Lo YK     Effect of arvanil (N-arachidonoyl-vanillyl-amine), a n ...
  ...
```

A tab-delimited table can be processed by many UNIX utiltiies. For example:

```
  esearch -db pubmed -query "capsaicin cancer pain management" | \
  elink -related | \
  efilter -query "conotoxin NOT morphine" | \
  esummary | \
  xtract -pattern DocumentSummary -element Id SortFirstAuthor Title | \
  sort -t $'\t' -k 2,2f -k 3,3f
```

sorts first by author name and then (for the same author) alphabetically by title:

```
  11000661  Chiang JS  New developments in cancer pain therapy.
  18956616  FÃ¼rst Z   [Central and peripheral mechanisms in antinociception ...
  10864900  Jerman JC  Characterization using FLIPR of rat vanilloid recepto ...
  12566085  Lo YK      Effect of arvanil (N-arachidonoyl-vanillyl-amine), a  ...
  ...
```

Or the title words can be extracted and sent through a series of UNIX commands:

```
  esearch -db pubmed -query "capsaicin cancer pain management" | \
  elink -related | \
  efilter -query "conotoxin NOT morphine" | \
  esummary | \
  xtract -pattern DocumentSummary -element Title | \
  sed 's/[^a-zA-Z0-9]/ /g' | tr 'A-Z' 'a-z' | xargs -n 1 | \
  sort | uniq -c | sort -k 1,1nr -k 2,2f
```

to give a table of occurrence counts for each word:

```
  8 of
  7 in
  4 cells
  4 human
  4 neuroblastoma
  4 sh
  4 sy5y
  3 and
  3 receptor
  2 a
  2 calcium
  2 channels
  ...
```

The PubmedArticle has a more detailed structure than the document summary:

```
  esearch -db pubmed -query "capsaicin cancer pain management" | \
  elink -related | \
  efilter -query "conotoxin NOT morphine" | \
  efetch -format xml | \
  xtract -outline
```

and more information is fielded, including author names:

```
  PubmedArticleSet
    PubmedArticle
      MedlineCitation
        PMID
        DateCreated
          Year
          Month
          Day
        Article
          Journal
            ISSN
            JournalIssue
              Volume
              Issue
              PubDate
                Year
            Title
            ISOAbbreviation
          ArticleTitle
          Pagination
            MedlinePgn
          ELocationID
          Abstract
            AbstractText
          Affiliation
          AuthorList
            Author
              LastName
              ForeName
              Initials
            Author
            ...
```

Using this information to craft a new xtract statement:

```
  esearch -db pubmed -query "capsaicin cancer pain management" | \
  elink -related | \
  efilter -query "conotoxin NOT morphine" | \
  efetch -format xml | \
  xtract -pattern PubmedArticle -element "MedlineCitation/PMID" LastName
```

results in a table of all authors for each record:

```
  23536870  Sousa   Vetter  Ragnarsson  Lewis
  22410003  Vetter  Mozar   Durek       Wingerd  Alewood  Christie  Lewis
  18956616  FÃ¼rst
  12566085  Lo      Chiang  Wu
  ...
```

Individual PubmedArticles can be retrieved directly by efetch:

```
  efetch -db pubmed -id 1937004 -format xml
```

and contain a list of Medical Subject Headings (MeSH Terms):

```
  ...
  <MeshHeadingList>
    <MeshHeading>
      <DescriptorName>Adenosine Triphosphatases</DescriptorName>
    </MeshHeading>
    <MeshHeading>
      <DescriptorName>Amino Acid Sequence</DescriptorName>
    </MeshHeading>
    <MeshHeading>
      <DescriptorName>Base Sequence</DescriptorName>
    </MeshHeading>
    ...
```

Visiting each MeSH term with a -block statement, and customizing the output format (changing the tab between columns into a newline character):

```
  efetch -db pubmed -id 1937004 -format xml | \
  xtract -pattern PubmedArticle -tab "\n" -element "MedlineCitation/PMID" \
    -block MeshHeading -tab "\n" -element DescriptorName
```

produces a list of MeSH terms, one per line:

```
  1937004
  Adenosine Triphosphatases
  Amino Acid Sequence
  Base Sequence
  ...
  Recombination, Genetic
  Saccharomyces cerevisiae
  Saccharomyces cerevisiae Proteins
```

MeSH terms can have one or more subheadings:

```
    ...
    <MeshHeading>
      <DescriptorName>Recombination, Genetic</DescriptorName>
      <QualifierName>genetics</QualifierName>
    </MeshHeading>
    <MeshHeading>
      <DescriptorName>Saccharomyces cerevisiae</DescriptorName>
      <QualifierName>genetics</QualifierName>
      <QualifierName>radiation effects</QualifierName>
    </MeshHeading>
    <MeshHeading>
      <DescriptorName>Saccharomyces cerevisiae Proteins</DescriptorName>
    </MeshHeading>
  </MeshHeadingList>
  ...
```

Adding a -subset exploration statement within the -block allows nested exploration of the subheadings for each MeSH term:

```
  efetch -db pubmed -id 1937004 -format xml | \
  xtract -pattern PubmedArticle -tab "\n" -element "MedlineCitation/PMID" \
    -block MeshHeading -tab "\n" -element DescriptorName \
      -subset QualifierName -tab "\n" -pfx "  " -element QualifierName
```

and results in a hierarchical list of MeSH terms and indented subheadings:

```
  1937004
  Adenosine Triphosphatases
  Amino Acid Sequence
  Base Sequence
  ...
  Recombination, Genetic
    genetics
  Saccharomyces cerevisiae
    genetics
    radiation effects
  Saccharomyces cerevisiae Proteins
```

MeSH terms and subheadings actually have major topic attributes:

```
  <MeshHeading>
    <DescriptorName MajorTopicYN="N">Saccharomyces cerevisiae</DescriptorName>
    <QualifierName MajorTopicYN="Y">genetics</QualifierName>
    <QualifierName MajorTopicYN="N">radiation effects</QualifierName>
  </MeshHeading>
```

that can be selected by:

```
  -element DescriptorName@MajorTopicYN
```

or:

```
  -element QualifierName@MajorTopicYN
```

Multiple -block commands can be used to explore different areas of the XML. Combining fields with commas allows them to be treated as sets, and the tab that normally separates these can be replaced with a -sep argument:

```
  efetch -db pubmed -id 781293,2678811,6301692,8332518 -format xml | \
  xtract -pattern PubmedArticle -element "MedlineCitation/PMID" \
    -block AuthorList -sep "|" -element LastName "#Author" \
    -block DateCreated -sep "-" -element Year,Month,Day | \
  sort -t $'\t' -k 3,3n | column -s $'\t' -t
```

produces a table that allows easy parsing of author names, counts the number of authors present, and prints the date each record was was entered into PubMed, sorting the results by the computed author count:

```
  781293   Casadaban                         1  1976-10-02
  6301692  Krasnow|Cozzarelli                2  1983-06-17
  8332518  Benson|Lipman|Ostell              3  1993-08-17
  2678811  Mortimer|Schild|Contopoulou|Kans  4  1989-11-22
```

Prefixing a field name with "#" counts the number of elements, and prefixing with "%" returns the number of characters in the element string value. The latter is useful for sorting by computed peptide length in protein sequence records.

Frequently used, long, or complicated search phrases can be saved in a file to avoid having to retype (or copy and paste) the full text for each query. Each line of the file has a shortcut keyword, a tab character, and the expanded search term. Shortcuts are referenced by placing them in parentheses and prefixing with a pound ("#") sign. For example, given a file named "query_aliases" containing:

```
  jour_filt    [MULT] AND ncbijournals [FILT]
  trans_imm    (transposition OR target) immunity
```

the esearch line in:

```
  esearch -alias query_aliases -db nlmcatalog -query "Science (#jour_filt)" | \
  esummary | \
  xtract -pattern DocumentSummary -element ISOAbbreviation \
    -subset ISSNInfo -sep "|" -element issn,issntype | \
  column -s $'\t' -t
```

will be expanded to:

```
  esearch -db nlmcatalog -query "Science [MULT] AND ncbijournals [FILT]"
```

with the query producing:

```
  J. Zhejiang Univ. Sci.  1009-3095|Print  1009-3095|Linking
  Science (80- )          0193-4511|Print  0193-4511|Linking
  Science                 0036-8075|Print  1095-9203|Electronic  ...
```

Taking an adventurous plunge into the world of programming, EDirect queries can be automated with shell scripts. And creative use of the "sh" and "xargs" commands can obtain the same behavior from the command line, without the need to write separate script files. For example:

```
  einfo -dbs | xtract -pattern DbName -element DbName | sort | \
  xargs -n 1 sh -c 'einfo -db "$0" | \
    xtract -pattern DbInfo -tab "\n\n" -element DbName \
      -block Field -pfx "[" -sep "]\t" -tab "\n" -element Name,FullName | \
    sed "s/ /*/g" | sort -k 2,2f | sed "s/*/ /g" | expand'
```

will display the indexed fields for each Entrez database:

```
  ...
  pubmed
  [AFFL]  Affiliation
  [ALL]   All Fields
  [AUTH]  Author
  [COLN]  Author - Corporate
  [FAUT]  Author - First
  [FULL]  Author - Full
  [LAUT]  Author - Last
  [AUCL]  Author Cluster ID
  [BOOK]  Book
  [CDAT]  Date - Completion
  [CRDT]  Date - Create
  [EDAT]  Date - Entrez
  [MHDA]  Date - MeSH
  [MDAT]  Date - Modification
  [PDAT]  Date - Publication
  [ECNO]  EC/RN Number
  [ED]    Editor
  [EPDT]  Electronic Publication Date
  [EID]   Extended PMID
  [FILT]  Filter
  [GRNT]  Grant Number
  [INVR]  Investigator
  [FINV]  Investigator - Full
  [ISBN]  ISBN
  [ISS]   Issue
  [JOUR]  Journal
  [LANG]  Language
  [LID]   Location ID
  [MAJR]  MeSH Major Topic
  [SUBH]  MeSH Subheading
  [MESH]  MeSH Terms
  [OTRM]  Other Term
  [PAGE]  Pagination
  [PAPX]  Pharmacological Action
  [CNTY]  Place of Publication
  [PPDT]  Print Publication Date
  [PTYP]  Publication Type
  [PUBN]  Publisher
  [PID]   Publisher ID
  [SI]    Secondary Source ID
  [SUBS]  Supplementary Concept
  [WORD]  Text Word
  [TITL]  Title
  [TIAB]  Title/Abstract
  [TT]    Transliterated Title
  [UID]   UID
  [VOL]   Volume
  ...
```