Specifications for the statistical functionality of the Bernstein workspace

Prepared by: Vlad Atanasiu (editor), Alois Haidinger, Viktor Kharnaukov, Maria Stieglecker, Emanuel Wenger

Recipient: TU Graz

1. The need for statistics

The statistical functionality is one of the two fundamental ways in which users can apprehend the data on paper history made available through Bernstein: either at the object level of individual paper descriptions, or at group level of selected papers according to the user's criteria. A quantitative description of the user's selection provides an insight into the structure of the data and allows its interpretation. The quantitative approach is essential for the historical research, expertise and cartography of papers.

2. Reply formats to users' requests

There are four formats for representing the reply of the Bernstein workspace to a user's request: (1) list of objects in the integrated databases matching the request's criteria, (2) numerical statistics on the reply, (3) graphical representation of the statistics, (4) cartographic representation of the statistics. For each request made, users can select any number and combinations of these representations. Additionally users should have the option to download the statistical data for further processing and correlation with own data.

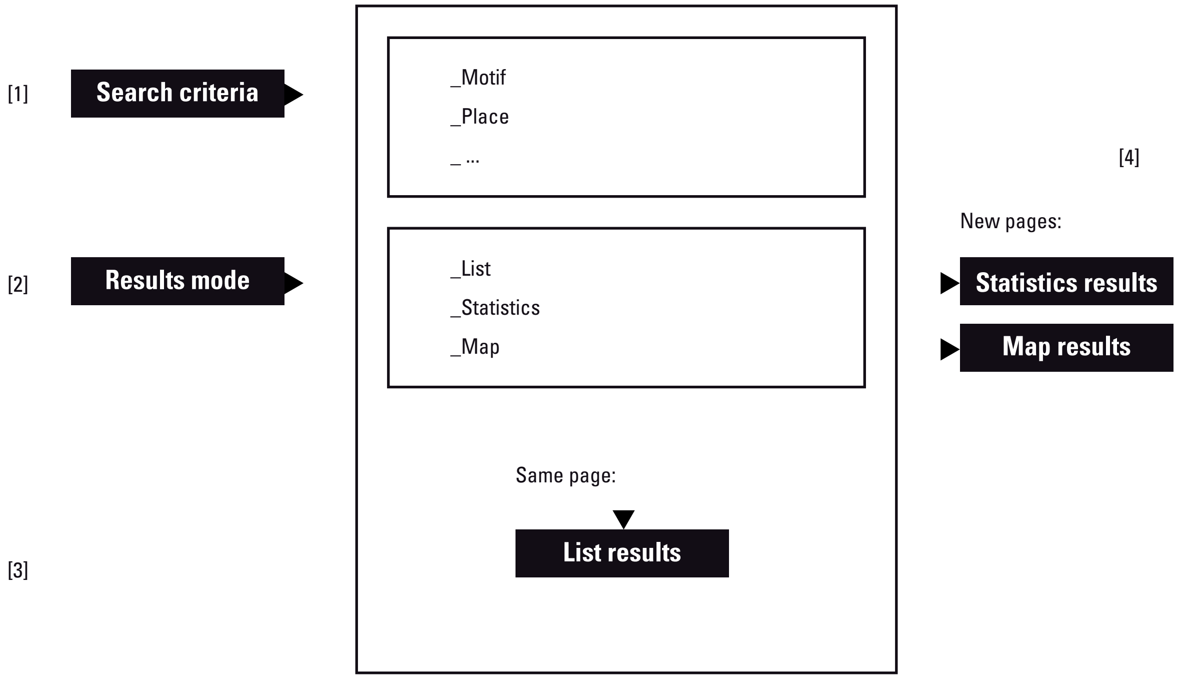
3. Navigation on the search pages

Following is the detail of the steps involved in the request/reply process.

Preliminaries – Statistical capabilities are provided for "simple search", "advanced search" and "component model search". Statistics are accessible trough the search pages – there will be no more a 'statistics' page on the website menu.

1. The following schema presents the structure of the simple and advanced search pages and the navigation therein.

SEARCH PAGE STRUCTURE



2. The user selects search criteria, either through the simple search input box or through the advanced options of the advanced search [1].

3. The user selects the modes for displaying the results (links) [2]:

   1. List of items
   2. Numerical and graphical statistics
   3. Cartographic representation

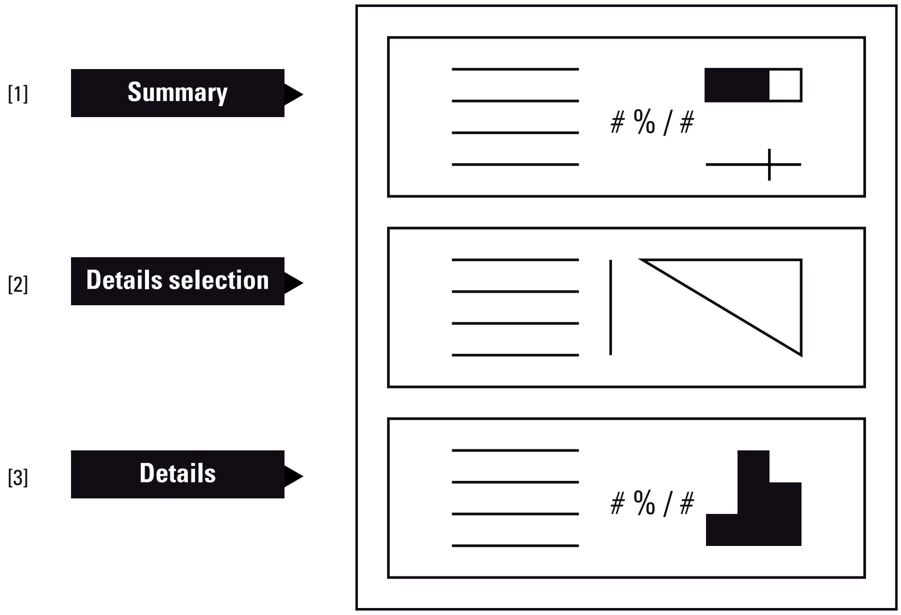
4. After the user hits one of the above links a message is displayed 'Searching the databases…' until the machine has finished searching.

5. If the chosen display mode is 'list', the results are displayed on the same page [3]. For 'statistics' and 'map' new pages are generated [4].

4. Navigation on the statistics page

1. The statistics page has three elements: a summary [1], a choice section [2] and a detailed statistics section [3].

STATISTICS PAGE STRUCTURE



2. The summary [1] gives for the user's data subset basic statistics for each searchable criteria.

Criteria		*Summary statistics*
Motif of watermark:		
	– Type	Quantity of types in selection (absolute values) / Percentage of types in selection out of the total number of types in the databases (relative values) / Total number of types in the databases – Diagram(*)
	– Subtype	Quantity (Percentage) / Total - Diagram
Place of paper use:		
	– Settlement	*Same as above*
	– NUTS 4	*Same as above*
	– NUTS 3	*Same as above*
	– NUTS 2	*Same as above*
	– NUTS 1	*Same as above*
	– Country	*Same as above*
Depository:		*Same as above*
	– Settlement	*Same as above*
	– NUTS 4	*Same as above*
	– NUTS 3	*Same as above*
	– NUTS 2	*Same as above*
	– NUTS 1	*Same as above*
	– Country	*Same as above*
Date of paper use		First date, last date in selection
Height of watermark		Minimum, interquartile mean, maximum - Diagram
Width of watermark		*Same as above*
Density of laid lines		*Same as above*
Distance between chain lines		*Same as above*

(*) The 'Diagram' provides a visual representation of the numbers. This is a small empty box or line, which's length denotes the total amount of data in the databases and the solid box inside the empty box or the vertical stroke on the line represents by its length or position the quantity of data in the results. For height, width, density and distance the diagram shows the position of the interquartile mean in regard to the minimum and maximum.

3. The choice section [2] allows the user to get more detailed statistics on results. It consists of radio buttons for selection of single and paired criteria (see list in next section).

4. If the user makes a choice in the extended statistics section, the results are displayed further down the page [3].

5. Detailed statistics for motif, place or depository with a single value choice (not paired values, see table in next section) are generated only if there is more than one motif, place or depository in the selection. Only up to 15 individual places and depositories are displayed in the statistics page. If there are more, a link to the cartographical representation is generated for displaying all. For motifs however, all of them are displayed in the statistics page.

6. The detailed statistics for motif, place or depository consists in the same statistics as shown in the summary, but broken down by individual motif, place or depository. Also given is the numerical value and the graphical representation of the percentage of each item in regard to the total in the reply.

7. For date, height, width, density and distance, the summary statistics are extended with data on range, mean, standard deviation, skew, kurtosis of the selected data.

8. As a reference, the user is also provided for each of the numerical criteria in the last paragraph with the same statistics characterizing the Bernstein databases as a whole.

9. The detailed data for date, height, width, density and distance is a numerical and graphical histogram. The user should be able to modify the number of bins according to the type and scale of analyzed data. For example dated could be binned by one year, ten years, fifty years or a century and the bins could have edges computed from the minimum and maximum in the data (example for date: 1447-1457-…-1497) or have semantically more meaningful edges (1440-1450-1460-…-1500).

9. An 'Export' button allows the user to download the numerical data and the graphic for further usages.

10. If the user makes a paired selection between two statistical criteria, the results are displayed in a two-dimensional table and the diagram is a bubbles diagram.

5. Criteria with statistical output

Currently there are 19 criteria for which a statistical output can be requested by users. Statistics for each of these criteria can be generated, as well as for pairs as follows.

		Single values	Paired values							
			Motif	Place	Depository	Date	Height	Width	Density	Distance
	Motif of watermark:									
1.	– Type	×	–	×	×	×	×	×	×	×
2.	– Subtype	×	–	×	×	×	×	×	×	×
	Place of paper use:									
3.	– Settlement	×		–	×	×	×	×	×	×
4.	– NUTS 4	×		–	×	×	×	×	×	×
5.	– NUTS 3	×		–	×	×	×	×	×	×
6.	– NUTS 2	×		–	×	×	×	×	×	×
7.	– NUTS 1	×		–	×	×	×	×	×	×
8.	– Country	×		–	×	×	×	×	×	×
	Depository:									
9.	– Settlement	×			–	×	–	–	–	–
10.	– NUTS 4	×			–	×	–	–	–	–
11.	– NUTS 3	×			–	×	–	–	–	–
12.	– NUTS 2	×			–	×	–	–	–	–
13.	– NUTS 1	×			–	×	–	–	–	–
14.	– Country	×			–	×	–	–	–	–
15.	Date of paper use	×				–	×	×	×	×
16.	Height of watermark	×					–	×	–	–
17.	Width of watermark	×						–	–	–
18.	Density of laid lines	×							–	×
19.	Distance between chain lines	×								–

6. Handling of dates

Dates in the Bernstein databases, and for the historical expertise in general, are characterized by a degree of uncertainty. While some paper documents are dated with certainty to a specific year ('made in 1500'), other are dated with approximation to a certain period, more or less well defined ('around 1450', or 'before 1675'). It is however necessary to include this type of dates in quantitative data evaluations. We show here how it is possible to provide a numerical and graphical representation of uncertain data.

Types of dates

Following are the usual wordings found in Bernstein databases regarding dates.

– in [1500]
– between [1450-1500]
– first/second half [of the 15th century]
– first/early, second/mid, third/last/late third [of the 15th century]
– first/second/third/fourth/last quarter [of the 15th century]
– first/.../sixth/.../last deceny [of the 15th century]
– before/after [1500]
– around [1500] [1450-1500]

It is possible to express any of these variants through a generalization:

date = certainty { certain | uncertain }
+ extension { on | between | before | after }
+ range { start year – end year }.

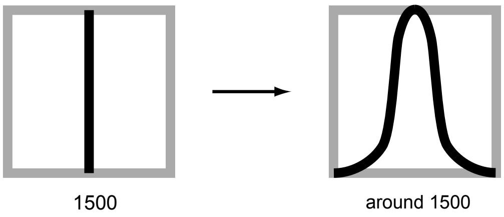
A compact notation for file format and screen display can look like:

Field 1	Field 2	Field 3	Wording
=	1500		in year 1500
~	1500		circa 1500
<	1500		before 1500
>	1500		after 1500
~	1450		mid 15th century *
~	1430	1470	mid 15th century *
=	1430	1470	between 1430-1470 *
~	1430	1470	circa 1430-1470
=	1490	1500	1490's
=	1466	1500	late 15th century *

You can see above a number of ambiguities marked with an asterisk. They arise because the exact meaning given to date wording can be different from person to person. But once a definition is provided, the conversion between meanings is possible.

Rendering uncertain dates

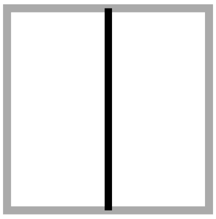
A date such as 'around 1500' is rendered by distributing values 'around' the specified date, in order to express the likelihood of a date different of the numerical value provided in the wording.



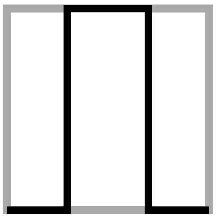
Numerically we transition from a vector to a matrix, with the date in the first column and the likelihood in the second column:

```
date = [1500 1]   >        date = [ 1480     0.2
                                    1481     0.25
                                    …
                                    1499     0.95
                                    1500     1
                                    …
                                    1420     0.2 ]
```

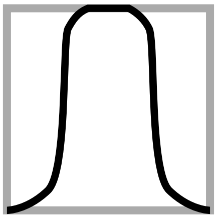
We can now visualize the different date types outlined above.



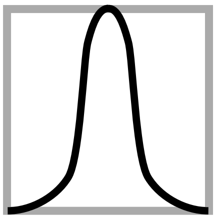
1500
single value



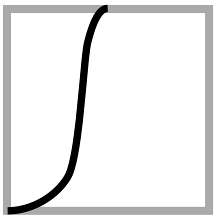
1450-1500
step range



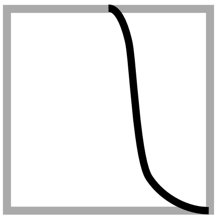
circa 1450-1500
bell shape



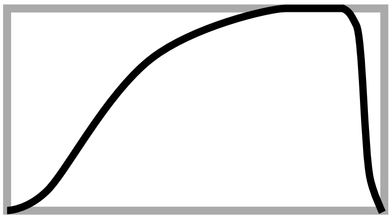
around 1500
gaussian



before 1500
left-hand o pen gaussian



after 1500
rig ht-hand o pen gaussian



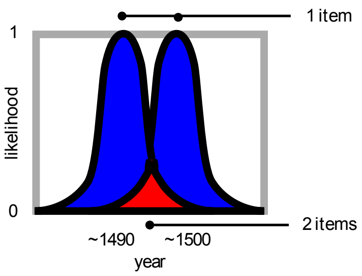
late 15th centu ry
asymmet ric bell shape

Types of uncertainty functions

The definition of the transfer function from a wording to a numerical and graphical output depends on those who generate the data, as well as the data itself. In the context of Bernstein two functions along with the exact date are expected to fulfill the needs: a 'step' and a 'bell' function. The step function gives values of 1 for the given range. The bell function is composed of a left-hand and a right hand Gaussian and a flat middle region of value 1. By modifying the middle region, one can obtain a Gaussian curve or half of it (left- or right-hand). The two sides don't have to be necessarily symmetrical.

For simplicity's sake we suggest to use for Bernstein a linear trapezoidal and a spline transfer function. The four points that define these shapes are intuitive: the extremities of the curve where the probability is 0 and the points between which the probability is 1. By modifying the parameters all needed shapes can be generated: distinct parameters yield a flat top line/curve (range dates), identical inside points remove the flat top (equivalent of the wording 'on', or 'around'), two identical left hand or right hand points produce a one hand line/curve ('before', 'after').

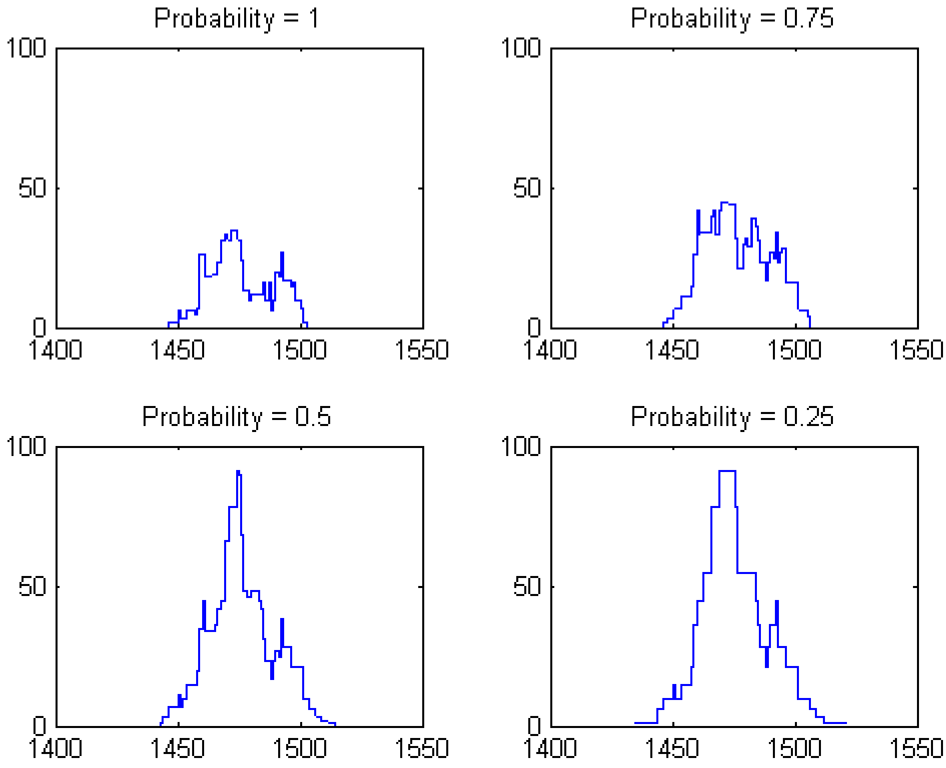
Representing quantities of uncertain dates



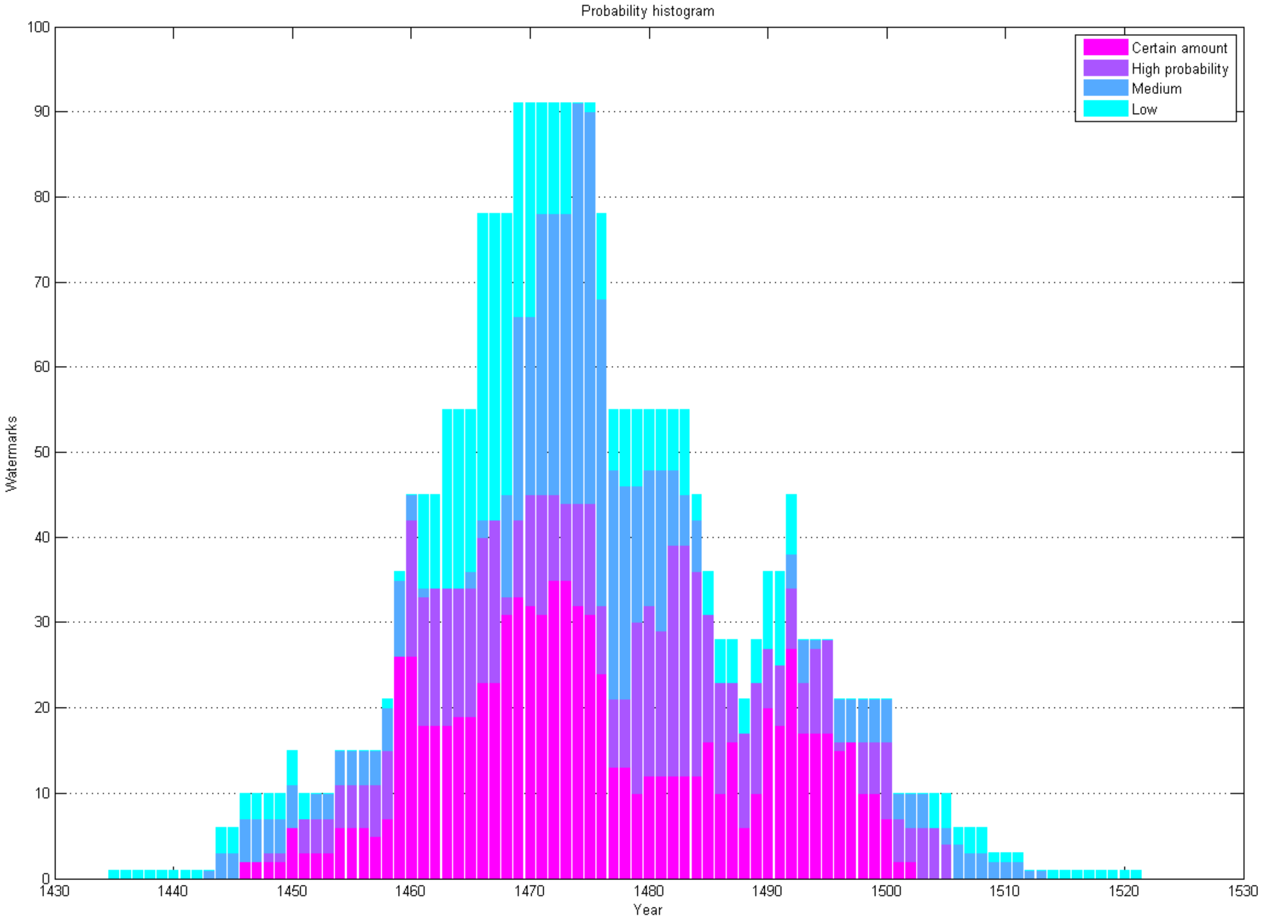
When two curves representing likelihoods of uncertain dates overlap, the area which is common takes the value 2. The process is additive for any given number of items.

A stairs histogram, stacked bar histogram or a color coded grid can be used to represent the distribution of number of items per year. Users should have the possibility to choose between 3 and 10 degrees of likelihood. 3 is more intuitive ('high/medium/low likelihood'), while 10 allows better numerical precision.

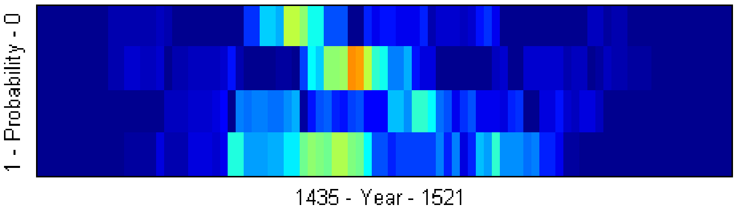
The diagram below shows the 'stairs' representation. If the user is interested in papers dated with absolute creativity, he considers the histogram with probability 1, in the upper left corner. If he chooses to include also dates with uncertain localization on the temporal axis, then he considers the other histograms. It is evident from the visual material that for year considered there are less precisely dated papers, than papers that might or might not have been dated to the specific year. To make a comparison, there is more money to be gained in a lottery if one chooses tickets with low odds of winning.



This is the 'stacked bars' diagram, where the four histograms of the proceeding page are superposed. Pink represents values with probability 1 (unambiguous in the sources) and the blue shades stand for values between >0, 0.3, 0.6 <1 (ambiguous dates). In the second diagram quantities are color coded.



The next diagram is the 'matrix' representation. Instead of representing quantities by areas of bars, here they are color coded, red indicating more values, blue less. The y-axis gives the four probability ranges as defined above. This is another way to represent the same data – some users might find it useful. (It needs some clean-up, like adding a colorbar and positioning the year and probability labels more clearly.)



Elements of programming

A suggested way to deal with dates would imply three data files and a transfer function.

1. The 'source data file' contains the dates as extracted from the databases according to the user request. It is made of four fields:
   1. the 'extension': expressed in words {'on', 'between', 'before', 'after'} or by symbols {'=', '~', '<', '>'};
   2. the 'range': containing one year (scalar) or two years (vector);
   3. the 'source id': the name of the source database, so as to be able to apply to each date the transfer function specific to the database to which it belongs.

2. The 'transfer function definition file' gives for each source the type of transfer function used for each extension type and the parameters of the function (for step functions the width, for Gaussians the sigma).

3. The 'histogram file' is the matrix used to draw the graphical representations. The first column gives the years, the ten subsequent columns give for each likelihood degree the quantity of items for a specific year.

4. The 'transfer function' is the algorithm that transforms the dates as stored in the databases into a histogram file and generates the graphical representations.

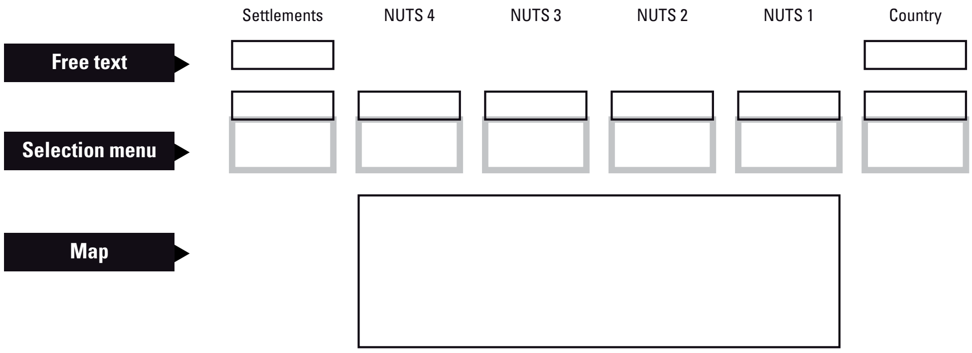
Code – See in the Annex the Matlab code used to generate the sample diagrams.

7. Handling of place names

As in the case of dates, place names are also ambiguous: for the same place there are different spellings in the databases, homonyms abound and the same place can refer to different things in different databases. We present here a model how disambiguation can be achieved at user and machine level.

1. *User interface* – The user has the choice to search for place names by typing free text in an input box or by selecting items from a menu with multiple choices. The free text search is available for settlements and countries. Selection menus are available for the settlements, the four NUTS region levels and the countries names. NUTS can also be selected by using a map (sensitive gif).

INTERFACE STRUCTURE FOR SEARCH OF PLACE NAMES AND DEPOSITORIES



2. *Query procedure* – This verbalized description is also presented in a graphical manner on the next page.

We identify three query steps. The first [1] is performed 'off-line' before the actual user's query and consists in updating the geographical names list stored in the workspace. The second step [2] is generating the search interface and processing the user's query. The last step [3] is the actual query of records from the databases and their presentation to the user as end products of the query.

2.1 *Update geodata* – This step presupposes that geographical references are stored in the databases and not in the workspace. Its role is to give the workspace the list of place names necessary to build the selection menus for the user. Materially it consists in a list of place names (settlements, regions 4 to 1 and countries) [5] and a list of translations in various languages of the countries names [6]. As a procedure this step requests from the databases updated lists of place names [4] and stores them in the workspace for further use. The update is either triggered by the workspace on a regular basis (once a month) or performed at the request of the database owners. Place names are transferred to the workspace only is a change in the place names list within the databases is observed at the moment of the regular check.

2.2 *Search interface* – This step starts with the request of the user's browser for the search page of the workspace [7]. When such a request for a html file is received by the Bernstein server, it generates *inter alia* the place names search interface. This is done by filling the selection menus with the appropriate place names extracted from the geographical names list of the workspace, updated in the previous step, and the country names list according to the user's Bernstein interface langue.

After the user made s/he choices, the workspace process the request to generate the request that will be send to the databases [9-13]. This processing is performed on requests on settlements [9-12] and on countries [13].

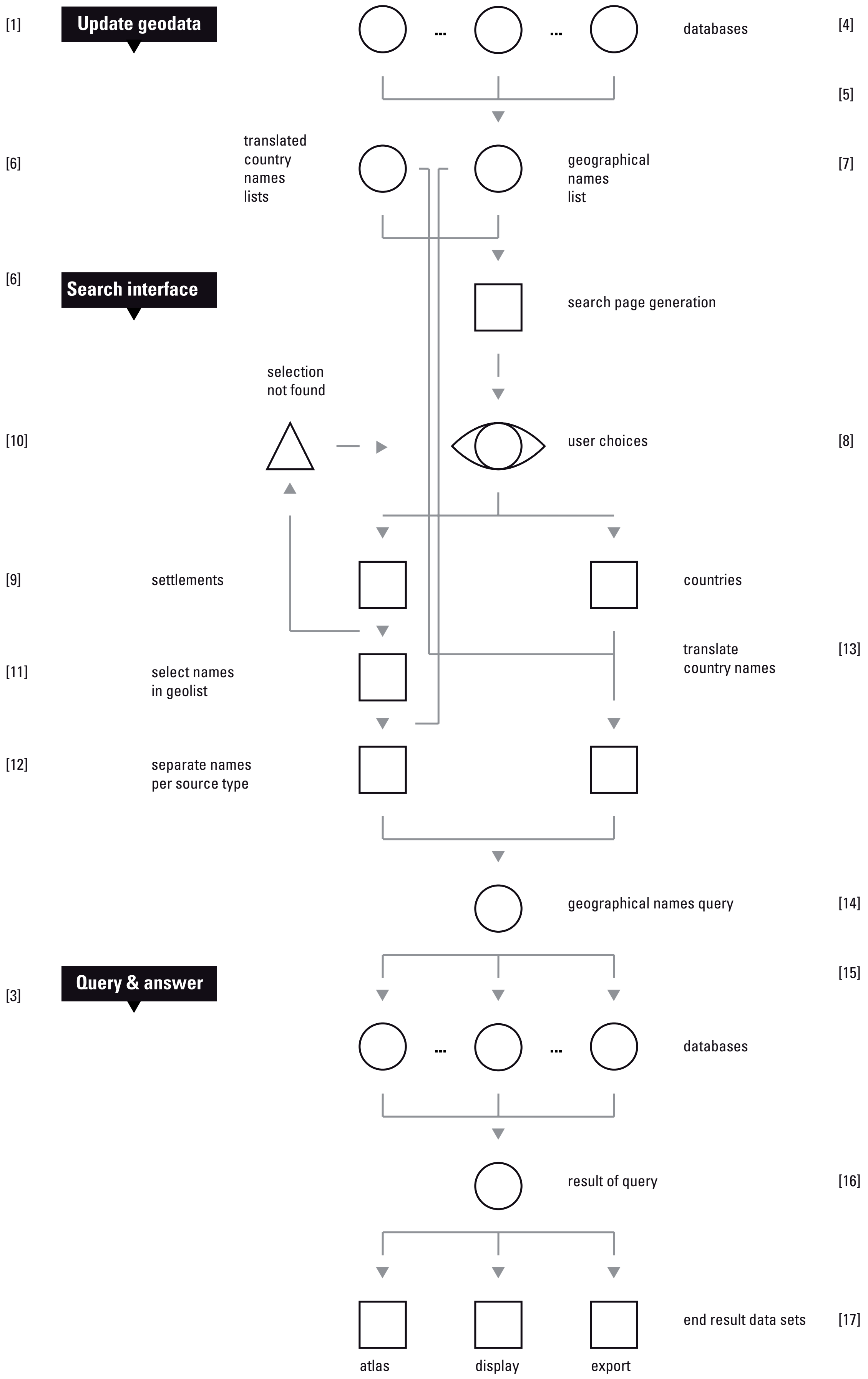
In the case of free text settlements input, it is checked in the geo names list, columns 'original name' and 'normalized name' that the input exists in the databases [9]. If not, a message is generated, notifying the user that s/he's request doesn't exist in Bernstein and that as an alternative s/he can use the selection menu [10]. Next, those place names are selected that correspond in the columns 'original name' and 'normalized name' to the request [11]. Then, the place names not corresponding to the selected database sources are discarded.

As countries appear in translation in the selection menu and in the native language in the 'country' field of the place names list, it is necessary to use the translation list to make the conversion [13].

Once these two conversion steps have been performed, the geographical query names are ready to be aggregated with the other user's criteria and send to the databases [14].

2.3 *Query and answer* – In this step the workspace sends the query to the databases [15], gets a resulting list of records [16] and process them for display on the screen, for export as tables and diagrams and as a file that can be used by the cartographical software [17].

PROCEDURE FOR REQUESTING GEOGRAPHICAL NAMES



8. Aspects of the interface

To accommodate the needs of the statistics functionality, the Atlas, and the query procedure, some aspects of the interface have to be taken into consideration.

1. *Watermarks classification* – The classification of the watermarks is and will continue to change due to new insights into the material or addition of new data. The workspace has to be designed in such a way as to accommodate changes in the watermark classification. This is a fundamental requirement. Otherwise the workspace will not be able to operate since the structure of the underlying data has changed.

2. *Indices* – Like for the place names, for textual search fields indices of available search values should be provided. These fields are: motif, places, depositories, creator.

3. *Numerical ranges* – Searches on numerical ranges should be made possible on a scalar ('1500'), a scalar with a ± year range ('1500 ±5'), and a range vector ('1450-1500'). The later is the new option made expressed as two input boxes.

9. Additional notes

WZMA – In the field 'Date' the value '9999' denotes an absence of information regarding temporal aspects of the record.

Export – Alphanumerical results can be exported as tab-delimited text files, UTF-16 encoded.

Export – Graphical results (the statistical diagrams) should be exportable in some format convenient for printing. Vector graphics (eps) are preferred, bitmaps are acceptable if high resolution (600 dpi tiff or png for binary images, 300 dpi for color images).

10. Annex – Programming code

```
% -----------------------------------
% Probability histogram
% -----------------------------------
% This code produces a numerical and graphical
% represenatation quantities of values characterized
% by uncertainity. It allows a quantitative assesment
% of approximations common in natural languages,
% such as 'circa', 'sometime before'.
%
% Summary:
% - transfer functions are extracted from databases
% - data subsets are extracted from databases
% - transform symbolism in numerical values
% - disambiguate the dates
% - merge the data subsets
% - bin probabilities
% - display results
%
% Requirements:
% Matlab 2007a, Fuzzy Logic Toolbox
%
% Vlad Atanasiu * 2008.03.09
% Last revision > 2008.03.10


% data sets: symbolism, year low, year high
db{1,1} = {...
    '=', '1450', '';
    '~', '1467', '';
    '~', '1453', '';
    '<', '1500', '';
    '=', '1480', '1500';
    '~', '1472', '1496';
    '~', '1499', '';
    '=', '1475', '';
    '>', '1470', '';
    '~', '1475', '';
    };

db{2,1} = {...
    '~', '1461', '';
    '=', '1469', '';
    '=', '1485', '1487';
    '~', '1469', '';
    '<', '1474', '';
    '~', '1474', '';
    '=', '1490', '';
    '=', '1489', '1495';
    };

db{3,1} = {...
    '~', '1468', '';
    '=', '1460', '';
    '~', '1459', '';
    '<', '1485', '';
    '~', '1487', '';
    '~', '1490', '1497';
    '=', '1492', '';
    };

% symbolism
sy = {...
    '=';...
    '~';...
    '<';...
    '>'...
    };
```

```
% transfer functions corresponding to the symbols
tf{1,1} = {...
    [0 0 0 0], 'pimf(a:d,[a b c d])';...
    [10 3 3 10], 'pimf(a:d,[a b c d])';...
    [10 3 0 0], 'pimf(a:d,[a b c d])';...
    [0 0 3 10], 'pimf(a:d,[a b c d])';...
    }; % smooth with flat top (spline)

tf{2,1} = {...
    [0 0 0 0], 'trapmf(a:d,[a b c d])';...
    [15 15 15 15], 'trapmf(a:d,[a b c d])';...
    [15 15 0 0], 'trapmf(a:d,[a b c d])';...
    [0 0 15 15], 'trapmf(a:d,[a b c d])';...
    }; % linear step

tf{3,1} = {...
    [0 0 0 0], 'trapmf(a:d,[a b c d])';...
    [25 0 0 25], 'trapmf(a:d,[a b c d])';...
    [25 0 0 0], 'trapmf(a:d,[a b c d])';...
    [0 0 0 25], 'trapmf(a:d,[a b c d])';...
    }; % linear triangular

% data sets transformed to matrices
% (transformation process not shown)
db{1,1} = [...
    1, 1450, NaN;
    2, 1467, NaN;
    2, 1453, NaN;
    3, 1500, NaN;
    1, 1480, 1500;
    2, 1472, 1496;
    2, 1499, NaN;
    1, 1475, NaN;
    4, 1470, NaN;
    2, 1475, NaN;
    ];
db{1,2} = [];

db{2,1} = [...
    2, 1461, NaN;
    1, 1469, NaN;
    1, 1485, 1487;
    2, 1469, NaN;
    3, 1474, NaN;
    2, 1474, NaN;
    1, 1490, NaN;
    1, 1489, 1495;
    ];
db{2,2} = [];

db{3,1} = [...
    2, 1468, NaN;
    1, 1460, NaN;
    2, 1459, NaN;
    3, 1485, NaN;
    2, 1487, NaN;
    2, 1490, 1497;
    1, 1492, NaN;
    ];
db{3,2} = [];

% disambiguate dates
for m = 1:size(db,1) % loop databases
    for n = 1:size(db{m,1},1) % loop records

        % get year & transfer function parameters
        y1 = db{m,1}(n,2);
        if isnan(db{m,1}(n,3)) % single date
```

```
         y2 = y1;
      else % range date
         y2 = db{m,1}(n,3);
      end
      v = cell2mat(tf{m,1}(db{m,1}(n,1),1));
      a = y1-v(1);
      b = y1-v(2);
      c = y2+v(3);
      d = y2+v(4);

      % get probabilities of transform date
      p = eval(char(tf{m,1}(db{m,1}(n,1),2)))';

      % get dates range
      y = (a:d)';

      % save results
      db{m,2} = [db{m,2}; y, p];
   end
end

% remove years with zero probability
for m = 1:size(db,1) % loop databases
   k = find(db{m,2}(:,2) == 0);
   db{m,2}(k,:) = [];
end

% merge databases
y = [];
p = [];
for m = 1:size(db,1) % loop databases
   y = [y; db{m,2}(:,1)];
   p = [p; db{m,2}(:,2)];
end

% bin probabilities
b = 3; % probability levels <--- USER INPUT
y1 = min(y);
y2 = max(y);
h = zeros(y2-y1+1,b+1);
x = y1:y2;

for m = y1:y2
   for n = 1:b
      h(m-y1+1,b+1-n+1) = sum( find( ...
         p( find( y == m ) ) >= (n-1)/b & ...
         p( find( y == m ) ) < n/b ...
         ) );
   end
   h(m-y1+1,1) = sum( find( ...
      p( find( y == m ) ) == 1 ...
      ) );
end

% cumulative probabilities
hc = zeros(size(h,1), size(h,2));
for m = 1:size(h,1)
   for n = 1:size(h,2)
      hc(m,n) = sum(h(m,1:n));
   end
end

% display results
% stairs histogram
hfg1 = figure('Name','Stairs representation');
ymax = max(max(hc))+max(max(hc))/10;
for k = 1:size(hc,2)
   subplot(ceil(size(hc,2)/2),2,k)
   stairs(gca, x, hc(:,k),...
      'Color','b',...
      'LineWidth',1);
   ylim([0 ymax])
   if k == 1
      title([ 'Probability = 1' ])
   elseif k == 2
      title([ 'Probability < 1' ])
   else
      ps = num2str( (b+1-k+1)/b );
      title([ 'Probability < ', ps(1:3) ])
   end
end

% stacked bar diagram
hfg2 = figure('Name','Stacked bars representation');
hhg = bar(x,h,'stack');
set(hhg, 'BarWidth', 0.8)
set(gca,'YGrid','on')
colormap(flipud(cool(b+1)))
title('Probability histogram')
xlabel('Year')
ylabel('Watermarks')
legend({'Certain amount','High probabil-
ity','Medium','Low'},1);

% matrix
hfg3 = figure('Name','Matrix representation');
pos1 = get(hfg2, 'Position');
pos2 = get(hfg3, 'Position');
% set(hfg3, 'Position', ...
%     [pos1(1), pos1(2)-80-pos2(4)/3, pos1(3), pos2(4)/3])
image(flipud(h'))
set(gca,'XTick',[]);
set(gca,'YTick',[]);
xlabel([num2str(y1), ' - Year - ', num2str(y2)])
ylabel('1 - Probability - 0')
```