VAMP: Visualization and Analysis of CGH arrays, transcriptome and other Molecular Profiles

User Manual

Institut Curie Bioinformatics Unit

January 11, 2006

Contents

1 Introduction

VAMP is a graphical user interface for the Visualization and Analysis of array-CGH, transcriptome and other Molecular Profiles. VAMP was primarily developed for visualization and analysis of tumor sample array-CGH profiles. Several types of visualization are proposed: for example classical CGH karyotype view or the genome-wide multi-tumor comparison views are available and allow the user to easily compare different arrays. Additional information concerning each clone or DNA region can be retrieved interactively from different public databases through external links. VAMP allows the user to confront the display of different types of molecular profiles such as array CGH profiles, transcriptome arrays, SNP (Single Nucleotide Polymorphism) arrays, Loss of Heterozygosity results (LOH), and Chromatin ImmunoPrecipitation arrays (ChIP chips). Many functions for analyzing CGH or other data are provided within the interface, including looking for recurrent regions of alterations, confrontation of genome data with transcriptome data or clinical information, clustering, synteny visualization...

VAMP consists of a graphical interface written in Java 1.4.2. The software retrieves information from XML files, making it easy to install it in any laboratory.

VAMP is provided with public data sets (Snijders et al., 2005; Gysin et al., 2005; Douglas et al., 2004; de Leeuw et al., 2004; Nakao et al., 2004; Veltman et al., 2003; Pollack et al., 2002; Snijders et al., 2001) as examples which can be directly used. This document describes the installation, the configuration and the main functions of VAMP.

Note that three movies present the main functions of VAMP:

- http://bioinfo-out.curie.fr/vamp/doc/tutorial/vamp-demo1.html (Basic functions)
- http://bioinfo-out.curie.fr/vamp/doc/tutorial/vamp-demo2.html (Data Analysis)
- http://bioinfo-out.curie.fr/vamp/doc/tutorial/vamp-demo3.html (Synteny analysis)

2 System requirements

VAMP is a Java applet. Therefore the Java Runtime Environment (JRE) must be installed on your computer before running VAMP. We recommend to use Java 1.5.0 (for better rendering and performances), see http://java.sun.com/j2se/1.5.0/ for download. Java 1.4.2 (http://java.sun.com/j2se/1.4.2/) is an alternative. For the best use of this applet, you can install Java 1.5.0 (recommended for better rendering and performances) : http://java.sun.com/j2se/1.5.0/

To visualize a large number of profiles simultaneously the java parameters regarding memory allocation must be correctly set.

For example, if your computer disposes of 1 Gb of RAM memory the java virtual machine memory can be set with the following parameters:

Java virtual machine parameter setting: `-Xms400000000 -Xmx800000000`

If you have more (or less) memory, these parameters must be adjusted accordingly. 512 Mb of memory is considered as a minimum, and we strongly recommend to dispose of 1Gb.

`-Xms` defines the minimal memory size allocated to the java virtual machine and `-Xmx` defines the maximal memory size (note that the `-Xmx` can not exceed the size of the RAM memory). In the example below the memory allocation ranges from 400Mo up to 800Mo.

With these parameters the users can load up to 700 microarrays (each with 3500 probes) simultaneously.

For more details please visit : http://bioinfo.curie.fr/tutorial/vamp/Java-configuration.html

3 Installation and configuration

3.1 Installation

VAMP can be installed either in standalone version or in intranet server version:

standalone version (Windows): once you have retrieved and extracted the VAMP.tar.gz file, just copy the VAMP directory into the `C:` directory. If you want to install the software in a different directory you have to replace the default path `C:/VAMP` with the new path in the following configuration files: `cgh.html`, `configuration/xml/syscfg.xml` and `configuration/xml/print-report.xml`.

standalone version (Mac OS X/Unix/Linux): once you have retrieved and extracted the VAMP.tar.gz file, just copy the VAMP directory in any directory. You have to replace the default path `C:/VAMP` with your new path in the following configuration files: `cgh.html`, `configuration/xml/syscfg.xml`, `configuration/xml/print-report.xml`. For example if you have copied the VAMP directory into `/usr/local` you replace `file:///C:/VAMP` with `file:///usr/local/VAMP`.

Intranet server version: once you have retrieved and extracted the VAMP.tar.gz file, just copy the VAMP directory into the root directory of your webserver (for example in the directory `/http/hosted/myHome.com/html` of your apache server). The following files should be modified : cgh.html, configuration/xml/syscfg.xml, configuration/xml/print-report.xml, in order to replace the default path `C:/VAMP` with the new path. For example let us assume you have copied the VAMP directory into `/http/hosted/myHome.com/html`. Then replace `file:///C:/VAMP` with `http://myHome.com/VAMP`.

Note that the standalone version uses the file protocol whereas the intranet server version uses the http protocol.

3.2 How to launch VAMP

In the case of a standalone version, just open the file index.html in the VAMP directory and in the case of an intranet server version just open the url `http://myHome.com/VAMP`.

3.3 Configuration

There are two configuration files in VAMP: the administrator configuration file which is the most important and the user configuration file.

3.3.1 Administrator configuration

The main configuration file (`syscfg.xml`)can be modified by the administrator. It contains the initialization of the resources (i.e. access path to the data, url address to your favorite public databases, cytogenetic banding informations, ...) which allows VAMP to be launched correctly.

A sample administrator file is given below: it can be modified to customize the contextual menus regarding the information of the clones, the transcriptome probes, SNPs, ...

syscfg.xml

```
<Parameter key="min:memory" value="200"/> : minimum size for a standard use (a message is displayed if the user memory is below this threshold)
<Proxy host='www-cache' port='1234'/> : proxy address for the URL

<TranscriptomeURLTemplate>
file:///C:/VAMP/data/xml/trs/#ProjectId#/#NumHisto#/chr/chr#ChrAlias#.xml
</TranscriptomeURLTemplate>
Used for locating the transcriptome data from a CGH array (#variables# are set from the XML CGH array data files)
```

```
<LOHURLTemplate>
file:///C:/VAMP/data/xml/microsat/#ProjectId#/#NumHisto#/chr/chr#ChrAlias#.xml
</LOHURLTemplate>
Used for locating the LOH data from a CGH array (#variables# are set from the XML CGH array data files)

 Parameterization of the contextual menu of an object of Clone type: how to add an hypertext link:
 <PropertyElementMenu object='DataElement' type='Clone'>
  <MenuItem type='separator'/>
  <MenuItem type='menu' title='External Links'>
       <MenuItem
            type='url'
            title='NCBI Clone Viewer'
url='http://www.ncbi.nlm.nih.gov/genome/clone/clname.cgi?stype=Name&amp;list=#Name#'
            target='_blank'/>
       <MenuItem
            type='url'
            title='NCBI Map Viewer'
url='http://www.ncbi.nlm.nih.gov/mapview/maps.cgi?ORG=hum&amp;query=#Name#&amp;MAPS=cntg-r,clone,sts,genes,comp&amp;CHR=#Chr#&amp;ABS_ZOOM=6M'
            target='_blank'/>
       <MenuItem
            type='url'
            title='Working Draft (UCSC)'
    url='@data'
            target='_blank'>
<token>
http://www.genome.ucsc.edu/cgi-bin/hgTracks?position=chr#Chr#:</token>
<sub>
<property>Position</property>
<token>150000</token>
</sub>
<token>-</token>
<add>
<property>Position</property>
<token>150000</token>
</add>
       </MenuItem>
       <MenuItem
            type='url'
            title='Ensembl ContigView'
    url='@data'
            target='_blank'>
<token>
http://www.ensembl.org/Homo_sapiens/contigview?chr=#Chr#&amp;vc_start=</token>
<sub>
<property>Position</property>
<token>150000</token>
</sub>
<token>&amp;vc_end=</token>
<add>
<property>Position</property>
<token>150000</token>
</add>
       </MenuItem>
       <MenuItem
            type='url'
            title='Ensembl CytoView'
    url='@data'
            target='_blank'>
<token>
http://www.ensembl.org/Homo_sapiens/cytoview?chr=#Chr#&amp;vc_start=</token>
<sub>
<property>Position</property>
<token>150000</token>
</sub>
<token>&amp;vc_end=</token>
<add>
<property>Position</property>
<token>150000</token>
</add>
       </MenuItem>
   </MenuItem>
 </PropertyElementMenu>


 Parameterization of the contextual menu of an object of ProbeSet type: how to add an hypertext link:
  <PropertyElementMenu object='DataElement' type='ProbeSet'>
  <MenuItem type='separator'/>
  <MenuItem type='menu' title='External Links'>
       <MenuItem
            type='url'
            title='Genecards Viewer'
    url='http://genecards.curie.fr/cgi-genecards/cardsearch.pl?search=#SourceID#'
            target='_blank'/>
  </MenuItem>
 </PropertyElementMenu>


 Parameterization of the contextual menu of the "Minimap: how to add an hypertext link:
<PropertyElementMenu object='MiniMapChr'>
   <MenuItem type='menu' title='External Links'>
       <MenuItem
            type='url'
            title='NCBI Map Viewer'
    url='http://www.ncbi.nlm.nih.gov/mapview/maps.cgi?ORG=human&amp;MAPS=cntg-r,clone,sts,genes,comp&amp;CHR=#Name#'
            target='_blank'/>
   </MenuItem>
 </PropertyElementMenu>

 <PropertyElementMenu object='MiniMapBand'>

   <MenuItem type='menu' title='External Links'>
       <MenuItem
            type='url'
```

```
          title='NCBI Cancer Chromosomes'
          url='http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=cancerchromosomes&amp;term=%2B#Chr##Arm##Name#&amp;cmd=Search'
          target='_blank'/>
   </MenuItem>
 </PropertyElementMenu>

 <UserDocumentation url='...'/> : indicates the physical localization of the user's guide

several examples of how to call a cgi (does not function locally) :
<Parameter key="eXcel:URL" value="http://bioinfo.curie.fr/cgi-bin/cghaia/export.pl?format=xl"/>
<Parameter key="eXcelChrAvg:URL" value="http://bioinfo.curie.fr/cgi-bin/cghaia/average_chr.pl?format=xl"/>
<Parameter key="cluster:URL" value="http://bioinfo.curie.fr/cgi-bin/cghaia/vamp_plugin.pl?action=clust"/>

root of the physical localization of the data xml files :
<Parameter key="importData:baseURL" value="file:///C:/VAMP/data/xml/cgh/"/>


indicates the physical localization of the image file which are used by the applet :
<Parameter key="applet_home_img:URL" value="file:///C:/VAMP/images/applet_home.png"/>
<Parameter key="cytoband:URL" value="file:///C:/VAMP/data/xml/cytoband/human"/>
<GraphElementIcon type="CGH Array" url="file:///C:/VAMP/images/vamp_cgh.jpg"/>
<GraphElementIcon type="CGH Chromosome Merge" url="file:///C:/VAMP/images/vamp_cgh.jpg"/>
<GraphElementIcon type="CGH Array Merge" url="file:///C:/VAMP/images/vamp_cgh.jpg"/>
<GraphElementIcon type="CGH Average" url="file:///C:/VAMP/images/vamp_cgh.jpg"/>
<GraphElementIcon type="Transcriptome" url="file:///C:/VAMP/images/trs.jpg"/>
<GraphElementIcon type="Transcriptome Average" url="file:///C:/VAMP/images/trs.jpg"/>
<GraphElementIcon type="Transcriptome Relative" url="file:///C:/VAMP/images/trs.jpg"/>
<GraphElementIcon type="Transcriptome Merge Relative" url="file:///C:/VAMP/images/trs.jpg"/>
<GraphElementIcon type="LOH" url="VAMP/images/LOH.jpg"/>

indicates the physical localization of cytogenetics descriptions :
<Cytoband organism="Human"
          url="file:///C:/VAMP/data/xml/cytoband/human/mai_2004"
          resolutions="400:550:850"
          default_resolution="400"/>
<Cytoband organism="Mouse"
          url="file:///C:/VAMP/data/xml/cytoband/mouse"
          resolutions="400"/>

indicates the physical localization of a file used for customized printing of a reporting :
<PrintPageTemplate url="file:///C:/VAMP/configuration/xml/print-report.xml"/>
```

3.3.2 User configuration

It is possible to customize the visual rendering of VAMP. The top-left panel provides a user-friendly interface for customizing the most important visual features (see section 5.4.2). Once the user has chosen his favorite parameters, he can save his configuration into an XML file, using `File` → `Configuration` → `Save`

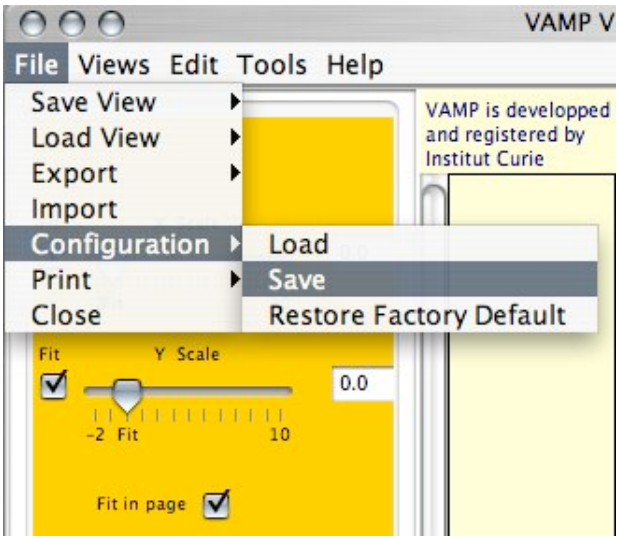

Figure 1: `File` → `Configuration` → `Save`.

Any saved configuration can be restored using `File` → `Configuration` → `Load`.

This XML configuration files also stores features that cannot be accessed directly within VAMP menus. To modify such features, just edit the XML configuration file. For example, if you change line

```
<CanvasBG>ccccff</CanvasBG>
```

to :

```
<CanvasBG>GREY</CanvasBG>
```

the main frame background color will switch from light blue to grey.

A sample user configuration file is given below :

```
<?xml version='1.0' encoding='iso-8859-1'?>
<CGHConfig>
<AxisBG>ORANGE</AxisBG>
<AxisEastSize>40</AxisEastSize>
<AxisFG>BLACK</AxisFG>
<AxisKaryoDisplayerFont>MonoSpaced:PLAIN:8</AxisKaryoDisplayerFont>
<AxisKaryoNameDisplayerFont>Serif:BOLD:9</AxisKaryoNameDisplayerFont>
<AxisKaryoSmallDisplayerFont>MonoSpaced:BOLD:7</AxisKaryoSmallDisplayerFont>
<AxisLabelFG>BLACK</AxisLabelFG>
<AxisLineFG>LIGHT_GRAY</AxisLineFG>
<AxisNorthSize>45</AxisNorthSize>
<AxisSouthSize>40</AxisSouthSize>
<AxisTranscriptomeFG>GRAY</AxisTranscriptomeFG>
<AxisTranscriptomeLabelFG>GRAY</AxisTranscriptomeLabelFG>
<AxisTranscriptomeLineFG>BLUE</AxisTranscriptomeLineFG>
<AxisTranscriptomeReferenceFG>770033</AxisTranscriptomeReferenceFG>
<AxisTranscriptomeReferenceLabelFG>770033</AxisTranscriptomeReferenceLabelFG>
<AxisTranscriptomeReferenceLineFG>BLUE</AxisTranscriptomeReferenceLineFG>
<AxisTranscriptomeRelativeFG>BLUE</AxisTranscriptomeRelativeFG>
<AxisTranscriptomeRelativeLabelFG>GRAY</AxisTranscriptomeRelativeLabelFG>
<AxisWestSize>90</AxisWestSize>
<AxisXDisplayerFont>Serif:PLAIN:9</AxisXDisplayerFont>
<AxisYCanvasPropertyNameFG>BLACK</AxisYCanvasPropertyNameFG>
<AxisYDisplayerFont>Serif:PLAIN:9</AxisYDisplayerFont>
<AxisYNameDisplayerFont>Serif:PLAIN:9</AxisYNameDisplayerFont>
<AxisYPropertyNameFG>BLUE</AxisYPropertyNameFG>
<BreakpointDashPadding>2</BreakpointDashPadding>
<BreakpointDashWidth>2</BreakpointDashWidth>
<BreakpointFG>RED</BreakpointFG>
.
.
.
<SeachPanelBG>ORANGE</SeachPanelBG>
<SearchPanelButtonBG>WHITE</SearchPanelButtonBG>
<SearchPanelButtonFont>SansSerif:PLAIN:9</SearchPanelButtonFont>
<SmoothingLineFG>BLACK</SmoothingLineFG>
<SmoothingPointFG>BLACK</SmoothingPointFG>
<SmoothingPointWidth>2</SmoothingPointWidth>
<TabBG>WHITE</TabBG>
<TabbedPaneFont>MonoSpaced:BOLD:10</TabbedPaneFont>
<ThresholdMaxYFG>RED</ThresholdMaxYFG>
<ThresholdMinYFG>GREEN</ThresholdMinYFG>
<ThresholdPanelBG>ORANGE</ThresholdPanelBG>
<ThresholdPanelButtonBG>WHITE</ThresholdPanelButtonBG>
<ThresholdPanelLabelFont>SansSerif:PLAIN:9</ThresholdPanelLabelFont>
<Threshold_CGH_MaxY>2.8</Threshold_CGH_MaxY>
<Threshold_CGH_MinY>0.01</Threshold_CGH_MinY>
<Threshold_ChIP-chip_MaxY>16.0</Threshold_ChIP-chip_MaxY>
<Threshold_ChIP-chip_MinY>0.01</Threshold_ChIP-chip_MinY>
<Threshold_SNP_MaxY>8.0</Threshold_SNP_MaxY>
<Threshold_SNP_MinY>0.1</Threshold_SNP_MinY>
<Threshold_TRSREL_MaxY>10.0</Threshold_TRSREL_MaxY>
<Threshold_TRSREL_MinY>0.25</Threshold_TRSREL_MinY>
<Threshold_TRS_MaxY>5000.0</Threshold_TRS_MaxY>
<Threshold_TRS_MinY>-10.0</Threshold_TRS_MinY>
<TitlePanelBG>ffffd0</TitlePanelBG>
<TranscriptomeMergeColorBase>WHITE</TranscriptomeMergeColorBase>
<Utr3FG>ee00</Utr3FG>
<Utr5FG>6600</Utr5FG>
<UtrHeight>2</UtrHeight>
<ViewBG>WHITE</ViewBG>
<ZoomPanelBG>ORANGE</ZoomPanelBG>
<ZoomPanelLabelFont>SansSerif:PLAIN:9</ZoomPanelLabelFont>
<ZoomPanelTextFieldFont>SansSerif:PLAIN:10</ZoomPanelTextFieldFont>
<PropertyAnnotations>
</PropertyAnnotations>
</CGHConfig>
```

3.4 Data format

The data format which is used in VAMP is XML (eXtensive Markup Language). Two types of XML file are necessary to run VAMP:

- *Array files* which contain the information related to a molecular profile: for each experiment there is a file which stores both array-level information (e.g. patient ID, number of clone, date of experiment, ...) and clone-level information (e.g. clone name, ratio value, chromosome number, location, ...). The Array files are organized according to the following hierarchy: team → projects → Array files. Each team may contain several projects and each project may contain several array files.

- the *Import file* which contains the list of projects and arrays that can be loaded within the VAMP, as described in section 5.1

VAMP distribution provides two scripts that generate these XML files automatically, as described in section 3.4.1:

`vampTxt2xml` generates Array XML files from csv (comma separated values) files.

Files are available from the Download :

`vampTxt2xml`:http://bioinfo-out.curie.fr/vamp/doc/script/vamp_txt2xml

Array XML files:http://bioinfo-out.curie.fr/vamp/doc/data/xml/douglas/Cancer3.xml

Csv (comma separated values) files:http://bioinfo-out.curie.fr/vamp/doc/data/txt/douglas/Cancer3

`vampProject2import` creates an Import XML file from the *Array files* belonging to a given directory.

Files are available from the Download :

`vampProject2import`:http://bioinfo-out.curie.fr/vamp/doc/script/vampProject2import

Import XML file:http://bioinfo-out.curie.fr/vamp/doc/data/import_data_public_xml/public.xml

In case you need to generate your *Array files* without using these scripts, subsection 3.4.2 provides a detailed description of corresponding XML tags.

3.4.1 Scripts for automated data generation

Array files can be generated by the script `vampTxt2xml`. As an input, this script takes csv (comma separated value) text files. The field names must correspond to the tags listed in section 3.4.2 and the file must be ordered by chromosome and position. We give an example of such an input file below:

```
Y,X,Chr,Name,Smt,Bkp,Out,Gnl,Weight
1.658040e-01,3247817,1,RP4-785P20,-0.020635750,0,0,0,NA
1.316050e-01,4487199,1,RP1-37J18,-0.020635750,0,0,0,NA
-9.597111e-02,5877818,1,RP11-49J3,-0.020635750,0,0,0,NA
3.588748e-02,7071571,1,RP3-438L4,-0.020635750,0,0,0,NA
-1.012979e-01,7653186,1,RP11-338N10,-0.020635750,0,0,0,NA
-1.728830e-01,9146799,1,RP3-510D11,-0.020635750,0,0,0,NA
5.558425e-02,10087260,1,RP4-575L21,-0.020635750,0,0,0,NA
.
.
.
```

This script outputs array files in the following directory structure:

- a first subdirectory structure contains pan-genomic profile of each array and chromosomic profile of all arrays in the project [project_name] (this subdirectory structure is used for Import functions)

```
/[project_name]/array/[array_name].xml
                chr/chr01.xml
                .
                .
                .
                chr/chrY.xml
```

- a second subdirectory structure contains chromosomic profile of all arrays in all projects (this subdirectory structure is used for Save and Load functions)

```
/all/chr01/[array_name].xml
      .
      .
      .
      chrY/[array_name].xml
```

The script `vampProject2import` automatically generates an import file based on the previous directory structure. An example of such an import file is given below:

```
<?xml version='1.0' encoding='iso-8859-1'?>
<FolderSet>
<Folder label="CGH ARRAYS (Team : public)">
<Folder label="snijders">
<Folder label="Arrays" type="CHR_ARRAY">
<Item label="gm00143" url="public/snijders/array/gm00143.xml"/>
<Item label="gm01524" url="public/snijders/array/gm01524.xml"/>
.
.
.
<Item label="gm13031" url="public/snijders/array/gm13031.xml"/>
<Item label="gm13330" url="public/snijders/array/gm13330.xml"/>
</Folder>

<Folder label="Chromosomes">
<Folder label="chr01" type="CHR">
<Item label="All arrays" url="public/snijders/chr/chr01.xml"/>
<Item label="gm00143" url="public/all/chr01/gm00143.xml"/>
<Item label="gm01524" url="public/all/chr01/gm01524.xml"/>
.
.
.
<Item label="gm13031" url="public/all/chr01/gm13031.xml"/>
<Item label="gm13330" url="public/all/chr01/gm13330.xml"/>
</Folder>
<Folder label="chr02" type="CHR">
<Item label="All arrays" url="public/snijders/chr/chr02.xml"/>
<Item label="gm00143" url="public/all/chr02/gm00143.xml"/>
<Item label="gm01524" url="public/all/chr02/gm01524.xml"/>
.
.
.
<Item label="gm13031" url="public/all/chr02/gm13031.xml"/>
<Item label="gm13330" url="public/all/chr02/gm13330.xml"/>
</Folder>
<Folder label="chr03" type="CHR">
<Item label="All arrays" url="public/snijders/chr/chr03.xml"/>
<Item label="gm00143" url="public/all/chr03/gm00143.xml"/>
<Item label="gm01524" url="public/all/chr03/gm01524.xml"/>
.
.
.
<Item label="gm13031" url="public/all/chr03/gm13031.xml"/>
<Item label="gm13330" url="public/all/chr03/gm13330.xml"/>
</Folder>

</Folder>
</Folder>

</FolderSet>
```

3.4.2 Array files

An Array file typically contains clone or probe-level information for one or more chromosome of one or more samples. It consists of an `ArraySet`, in which one ore more `Array` elements are embedded. Each of these `Array` elements regroups `Obj` elements, which store probe-level information for one particular sample.

The corresponding XML files has the following hierarchical structure:

```
<?xml version='1.0' encoding='iso-8859-1'?>
<ArraySet>
    <SetName>Name</SetName>
    <Array>
        // Array Properties
        <Obj>
            // Obj Properties
        </Obj>
        <Obj>
            // Obj Properties
        </Obj>
        .
        .
        .
    </Array>
    <Array>
        // Array Properties
        <Obj>
            // Obj Properties
        </Obj>
        <Obj>
            // Obj Properties
        </Obj>
        .
        .
        .
    </Array>
    .
    .
    .
</ArraySet>
```

We provide a short description of `Array` and `Obj` properties:

Array properties The following tags are mandatory:

`<Organism>` **:** species

`<Project>` **:** project name

`<Name>` **:** array name

`<Type>` **:** array type. Any type description may be used, but default VAMP functions are associated with the following types: `CGH`, `Trs`, `LOH`, `SNP`, and `ChIP`, as described in table 1

<Ratio> : the type of the signal value: ratio (M) or log ratio (L).

<Url> : the physical localization of the XML files which contains the array data for each chromosome (tag <Chr>); these Urls are used by the software for saving analysis (save command of File menu) and restore any saved analysis (load command of the File menu).

The following tags are optional, but filling them allows one to identify the patient, and therefore to link together profiles from the same patient, e.g. array-CGH and transcriptome data:

<ProjectId> : identification of the project

<NumHisto> : histological number (which corresponds to a unique patient ID)

The tag <SampleAdditionalData> links array files with additional data (e.g. clinical data) that are stored for a given sample in an XML file, as described in section 3.4.3.

Any other tag may be added in the <Array> description. They will be listed in the Info Panel of the bottom-left frame (figure 4), and will be taken into account by the Search Arrays function (section 5.3.1).

Object properties Object descriptions are encapsulated as follows:

```
<Obj>
    <Properties>
        <Type>
            // Obj Type
        </Type>
        // other Obj Properties
    </Properties>
</Obj>
```

Tag <Type> describes the object type. Any type description may be used, but the default VAMP functions are associated to the following types : Clone, Probeset, Microsat, SNP, and Probe.

There is a default type for the objects of an array, which depends on the array type as summarized in Table 1.

Data type	Array type	Object type
CGH-array	CGH	Clone
transcriptome arrays	Trs	ProbeSet
Loss Of Heterozygosity data	LOH	Microsat
Single Nucleotide Polymorphism arrays	SNP	SNP
Chromatine ImmunoPrecipitation arrays	ChIP	Probe

Table 1: Default object types in the different types of arrays

Any XML tag may be used for describing objects in VAMP, but some of them are mandatory. Some others are required to use some of the VAMP functions. The list of mandatory or required object tags depends on the array type. In the two following paragraphs we describe the mandatory and optional tags we used for the two most widely used data: array-CGH data and transcriptome array data.

Array-CGH data The following four tags are mandatory for describing array-CGH clones:

`<X>` **:** genome position (integer)

`<Y>` **:** ratio or log2-ratio value (real), consistently with the value of the array-level tag `<Ratio>`: `M` for ratio or `L` for log2-ratio.

`<Chr>` **:** chromosome name (1, 2, . . X, Y)

`<Name>` **:** clone name (e.g. `RP11-84M16`)

The remaining tags are not mandatory, but they are required to activate some of the VAMP functions, related to DNA copy number analysis:

- functions related to DNA copy number analysis:

  `<Smt>` **:** smoothing value (smoothed value of signal for a Clone)

  `<Bkp>` **:** breakpoint (a DNA breakpoint has been detected after this clone)

  `<Out>` **:** outlier (clone with signal significantly different from its neighbors: values are either -1 or 1)

  `<Gnl>` **:** clone status : amplicon, gain, normal, loss, and double loss (values are -10 for double loss, -1 for loss, 0 for normal, 1 for gain, 2 for amplicon)

  These values may be calculated for example with the GLAD algorithm (Hupé et al., 2004)

The clone size (tag `<Size>`, value in bp) is required for X-zooming in up to the clone scale (not available in the public data sets provided with this distribution).

Any other tags may be added in the `<Obj>` description. They will be listed in the Info Panel of the bottom-left frame (figure 4), and will be taken into account by the `Search Elements` function (section 5.3.2).

A sample XML file for CGH array data is given below:

```
<?xml version='1.0' encoding='iso-8859-1'?>
<ArraySet>
    <SetName>gm01535</SetName>
    <Array>
        <Team>public</Team>
<Organism>Human</Organism>
<Project>snijders</Project>
<ProjectId>15</ProjectId>
<NumHisto>4948</NumHisto>
<SampleAdditionalData URL="additional/snijders/gm01535.xml"/>
<Chr>1</Chr>
<Name>gm01535</Name>
<Date>Thu Apr  7 16:14:54 2005</Date>
<Type>CGH</Type>
<ObjKey>Name</ObjKey>
<NbSpot>6813</NbSpot>
<NbRep>3</NbRep>
<RatioScale>L</RatioScale>
<Url>public/all/chr01/gm01535.xml</Url>
<NbObj>2271</NbObj>
<Obj>
    <Properties>
        <Type>Clone</Type>
<Y>NA</Y>
<X>1000</X>
<Chr>1</Chr>
<Name>GS1-232B23</Name>
<Smt>NA</Smt>
<Bkp>NA</Bkp>
<Out>NA</Out>
<Gnl>NA</Gnl>
    </Properties>
</Obj>
<Obj>
    <Properties>
        <Type>Clone</Type>
<Y>0.009421</Y>
<X>468000</X>
<Chr>1</Chr>
<Name>RP11-82d16</Name>
<Smt>0.0193950</Smt>
<Bkp>0</Bkp>
<Out>0</Out>
```

```
<Gnl>0</Gnl>
    </Properties>
</Obj>
<Obj>
    <Properties>
        <Type>Clone</Type>
<Y>-0.021783</Y>
<X>2241000</X>
<Chr>1</Chr>
<Name>RP11-62m23</Name>
<Smt>0.0193950</Smt>
<Bkp>0</Bkp>
<Out>0</Out>
<Gnl>0</Gnl>
    </Properties>
        </Obj>
.
.
.
    </Array>
</ArraySet>
```

Transcriptome array data The following four tags are mandatory for describing transcriptome probe sets:

`<ObjectId>` **:** the name of the object

`<PosBegin>` **:** start position of the probe set on the genome

`<PosEnd>` **:** end position of the probe set on the genome

`<Signal>` **:** signal value

The following fields are optional; they provide hypertext links towards public databases within VAMP.

`<Source>` **:** origin of the object (IMAGE, GenBank, ...)

`<SourceID>` **:** object identification for the public database

Any other tags may be added in the `<Obj>` description. They will be listed in the Info Panel of the bottom-left frame (figure 4), and will be taken into account by the `Search Elements` function (section 5.3.2).

A sample XML file for transcriptome array data is given below:

```
<?xml version='1.0' encoding='iso-8859-1'?>
<ArraySet>
    <SetName>chr1</SetName>
    <Array>
        <Organism>Human</Organism>
<Project>pollack</Project>
<ProjectId>pollack</ProjectId>
<NumHisto>NORWAY_7</NumHisto>
<Chr>1</Chr>
<Name>NORWAY_7_EXPR</Name>
<Date>28/04/2004</Date>
<Type>TRS</Type>
<Url>trs/pollack/NORWAY_7/all/chr1/NORWAY_7_EXPR.xml</Url>
<Obj>
    <Properties>
<Type>ProbeSet</Type>
<ObjectId>IMAGE:322807</ObjectId>
<Signal>-0.88</Signal>
<PosBegin>13850</PosBegin>
<PosEnd>14650</PosEnd>
<Size>800</Size>
<Source>IMAGE</Source>
<SourceID>322807</SourceID>
</Properties>
</Obj>
<Obj>
    <Properties>
<Type>ProbeSet</Type>
<ObjectId>IMAGE:190915</ObjectId>
<Signal>-0.03</Signal>
<PosBegin>167764</PosBegin>
<PosEnd>168564</PosEnd>
<Size>800</Size>
<Source>IMAGE</Source>
<SourceID>190915</SourceID>
    </Properties>
</Obj>
.
.
.
    </Array>
</ArraySet>
```

3.4.3 Sample annotation data

Finally we describe the data format used for adding sample annotation data to molecular profiles.

Clinical data descriptions are encapsulated as follows:

```
<SampleAdditionalData>
        <ClinicalData>
                <Age>72</Age>
                <Sex>Male</Sex>
                <Stage>2</Stage>
                <Location>Left</Location>
                <Bat26>Stable</Bat26>
        </ClinicalData>
</SampleAdditionalData>
```

Note that any property specified by `<ClinicalData>` will be used in VAMP within functions related to 'Sample Annotation' (see section 6.5); In the example above, the properties `<Age>`, `<Sex>`, `<Stage>`, `<Location>`, `<Bat26>` were not previously known by VAMP but they will be taken into account by VAMP.

4 Overview

A typical VAMP window is divided into three areas (figure 2): the main frame consists of the graphical display of the profiles; the top left frame controls zoom, search and drawing options; the bottom left frame offers the choice between textual information on the object under the mouse pointer, or context information, called MiniMap.

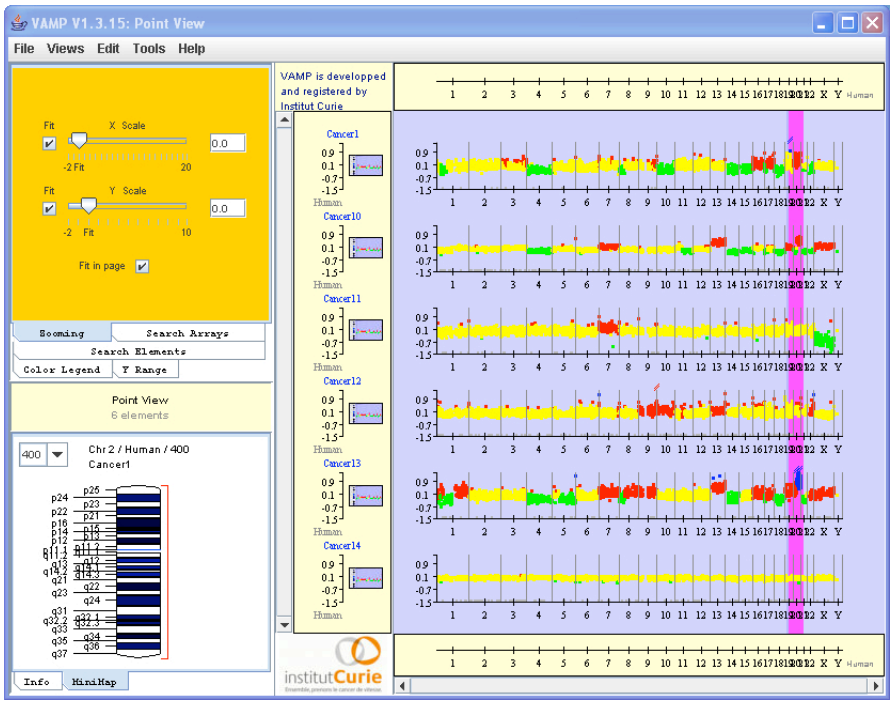

Figure 2: **Typical VAMP window.**

4.1 Top-left frame

The top-left frame is composed of several tabs controlling zooming, search and drawing options. These functionalities are described in details in subsections 5.3 and 5.4.

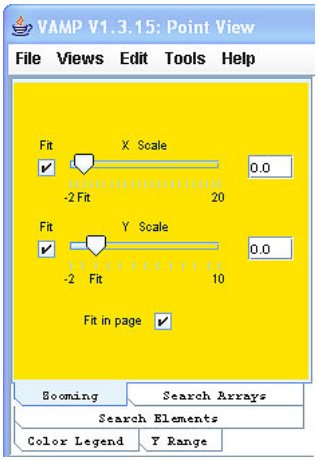

Figure 3: **Top-left Frame** - Zooming tab is selected.

4.2 Bottom-left frame

The bottom left frame offers the choice between textual information on the object under the mouse pointer, or context information, called MiniMap (figure 4).

The information panel allows to retrieve array-level information (e.g. project id, patient id), or clone-level information (e.g. ratio, chromosome, position).

The MiniMap panel allows the user to easily visualize any clone name and its chromosome location. Right-clicking on one of the cytogenetic banding allows to open a contextual menu linking to public databases. This menu can be configured as described in section 3.3.1.

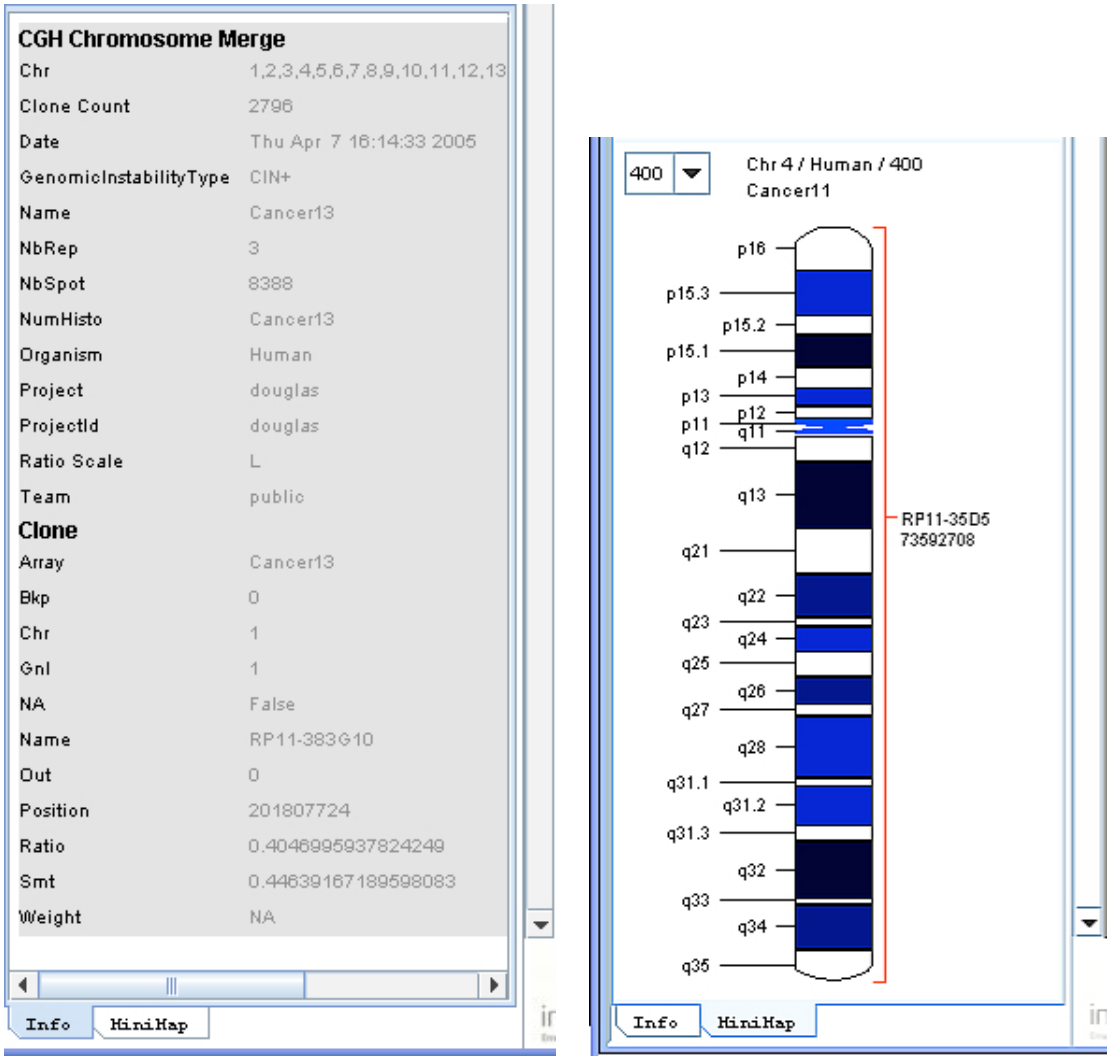

Figure 4: **Info Panel (left) and Mini Map Panel (right).**

4.3 Main frame

The main frame provides various ways of visualizing molecular profiles, which are described in details in subsection 5.2.

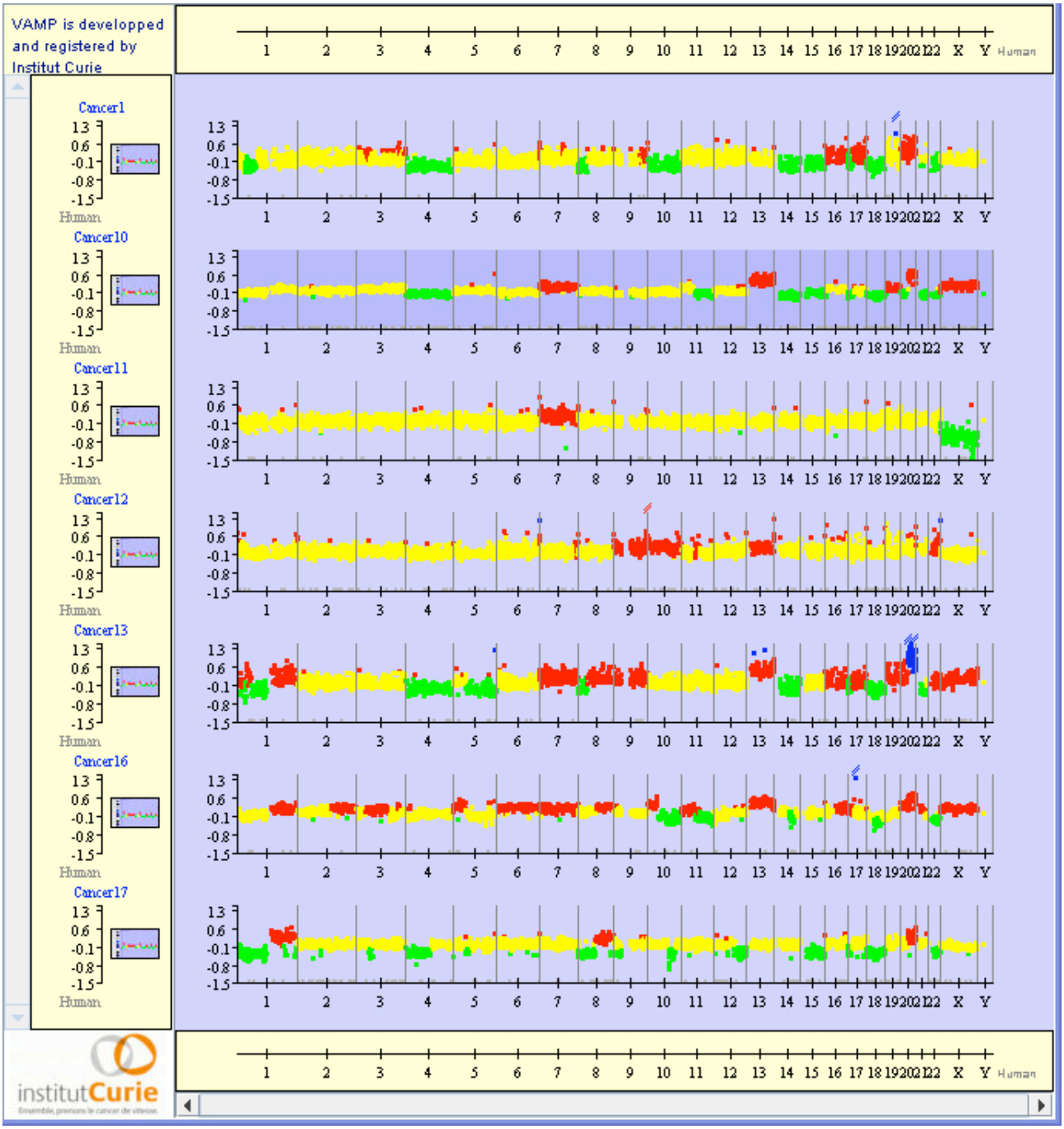

Figure 5: **Main frame** - Graphic Panel.

5 Basic functions

All user actions are accessible either through a Menu on the menu-bar, or through pointing to or clicking objects. When using VAMP, the session can be saved in local XML files. Reloading the file later on allows the continuation of the analysis within the context of the previous work, or allows the exchange of results and data with colleagues. All user preferences can also be stored in local XML files. Drag and drop capability is offered for any profile, from one window to any other window, the rendering being automatically adapted (e.g. from a dot plot view to a karyotype view). An advanced printing function is offered (see 5.5 for details).

5.1 Data import

Once XML array files and import files have been generated (see 3.4.1), data can be imported into VAMP, as shown in figure 6.

After selecting a project (figure 7), the user can import either pan-genomic profiles (8) or chromosome profiles (9) for the same project or from different projects.

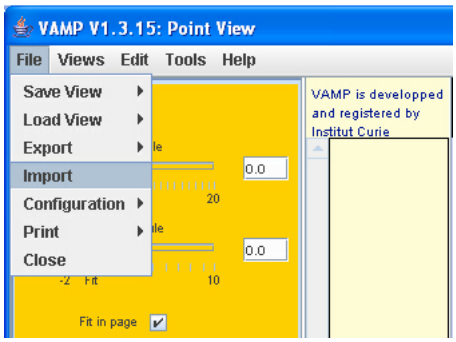

Figure 6: **Import data.**

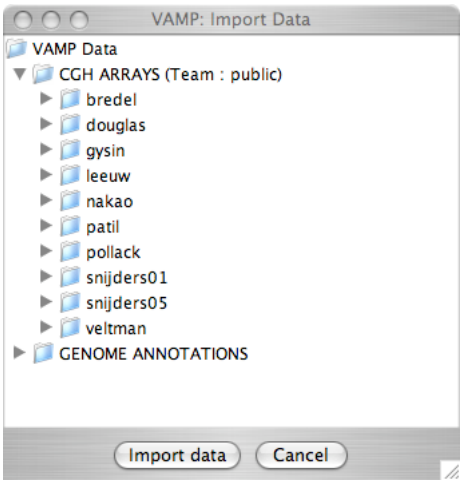

Figure 7: **Select a project.**

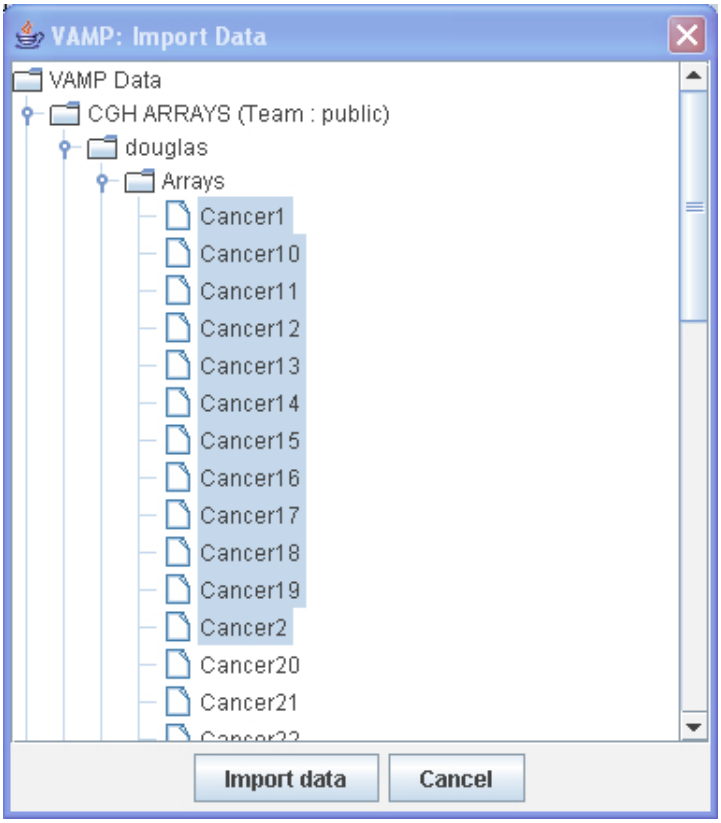

Figure 8: **Import pan-genomic profiles.**

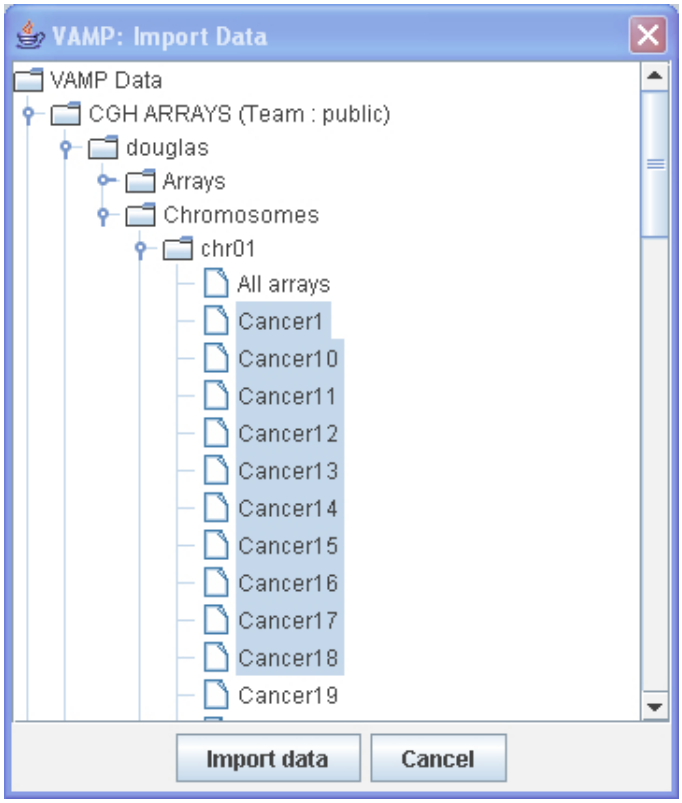

Figure 9: **Import chromosome profiles.**

5.2 Data display

VAMP currently offers several types of visualization that can be displayed in the main frame: (i) Profile View, (ii) Karyotype View, (iii) Dot Plot View, (iv) List View. These views all allow simultaneous visualization of several profiles (the only limitation is the memory size of the computer running VAMP, or more precisely, the memory allocated to the Java virtual machine (see section 2): for example with an 800 Mb Java virtual machine memory, 700 microarrays (each with 3500 probes) can be loaded simultaneously).

Whatever view is chosen, the profiles can be represented in Genomic mode or Chromosome mode. The Genomic mode simply depicts the profiles along all the concatenated chromosomes. It is the most usual representation, and allows comparison of profiles from different samples or comparison of different types of profiles from a given sample. The Chromosome mode is similar to the Genomic mode except that it only displays one particular chromosome. It is also possible to merge several chromosomes and to represent those chromosomes useful for the study.

5.2.1 Profile View

The Profile View can display the profiles as points, barplots or curves (see **Figure** 10). The Profile View can also display symbols for chromosome telomeres and centromeres, and can show the results of CGH ratio statistical analysis (e.g. breakpoints, or smoothed signal values, ...) (see **section 5.2.5**).

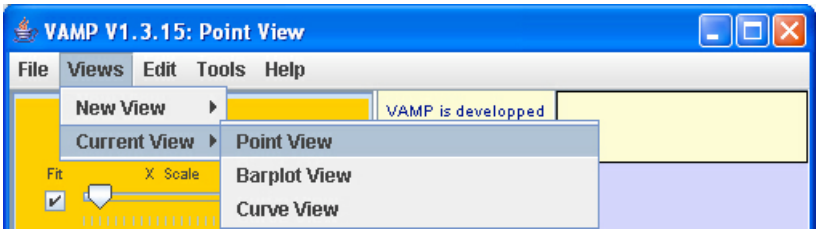

Figure 10: `Views` → `Current View` → `Point View` - After importing the genomic profiles (see section 5.1), it is possible to switch from one type of representation to another.

The main frame can be split into two frames (see **Figure 12**). The upper frame can, for example, contain a profile for reference when browsing a collection of profiles in the lower frame. The two frames have separate control of Y-scale and Y-scrolling, but have the same X-scale and X-scrolling.

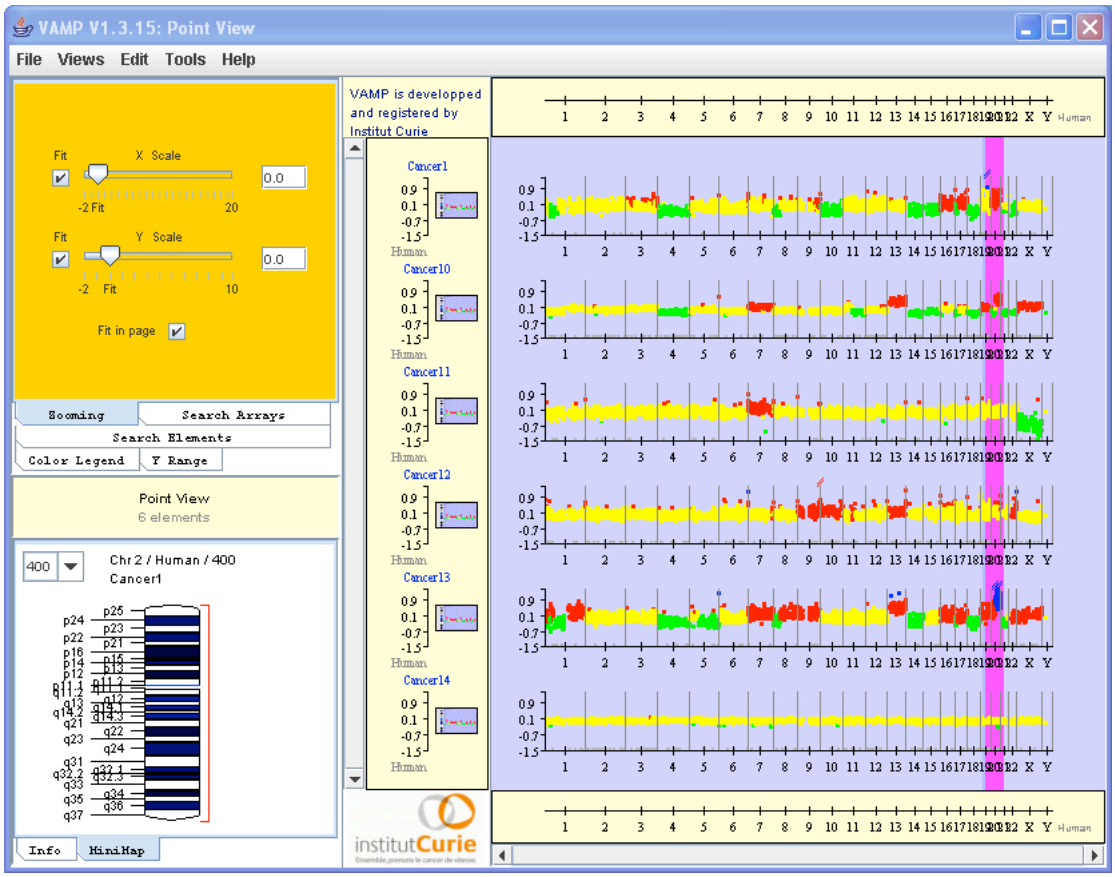

Figure 11: **Profile View (Point View).**

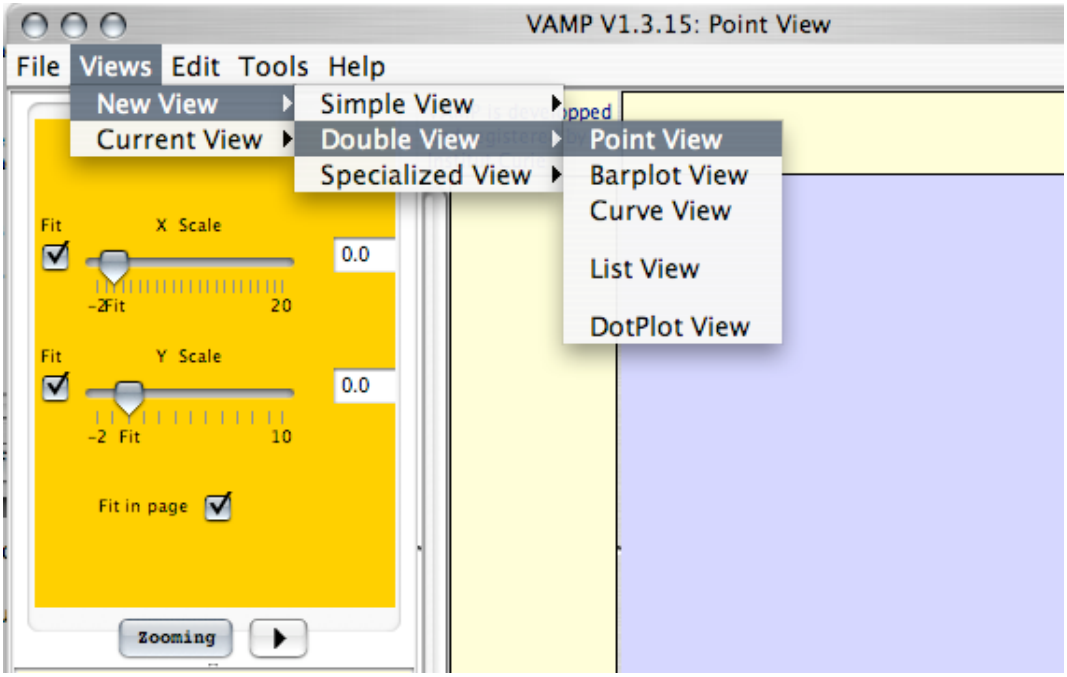

Figure 12: `Views` → `New View` → `Double View` → `Point View` - The user can open a new double view (see **Figure 45**).

5.2.2 Karyotype View

The Karyotype View displays profiles with the classical CGH rendering: vertical representations of chromosomes with cytogenetic banding and contiguous representation of sample profiles.

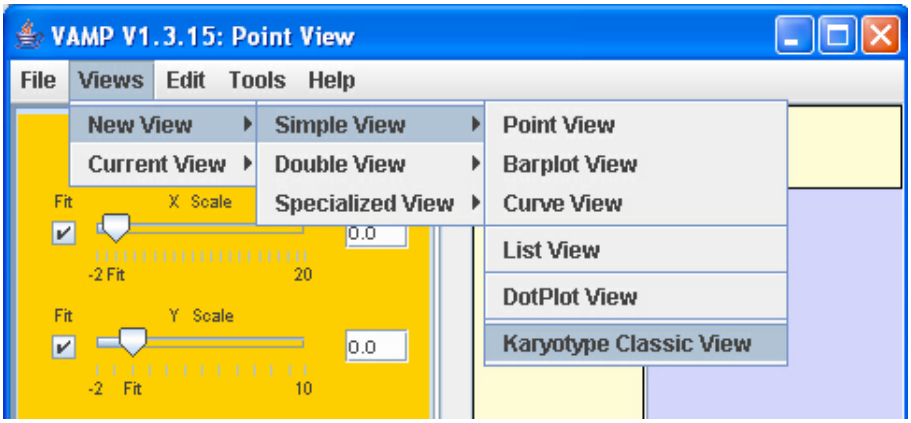

Figure 13: `Views` → `Simple View` → `Karyotype Classic View` - The user opens a new Karyotype Classic View.

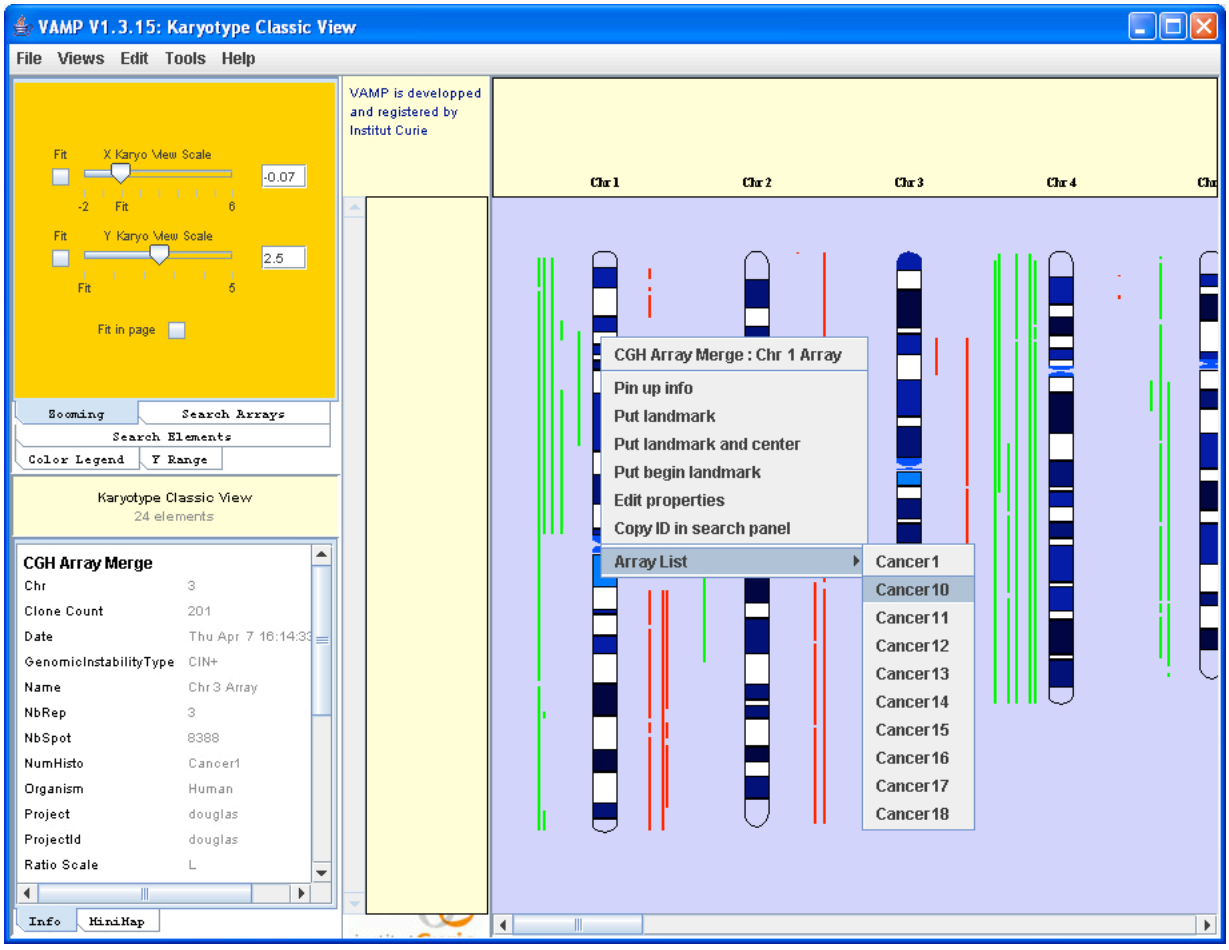

Figure 14: **Karyotype Classic View** - Right-clicking on a clone, chromosome or array, pops up a contextual menu.

5.2.3 Dot Plot View

The Dot Plot View does not consider the microarray probe positions on the genome, but only their ranks.

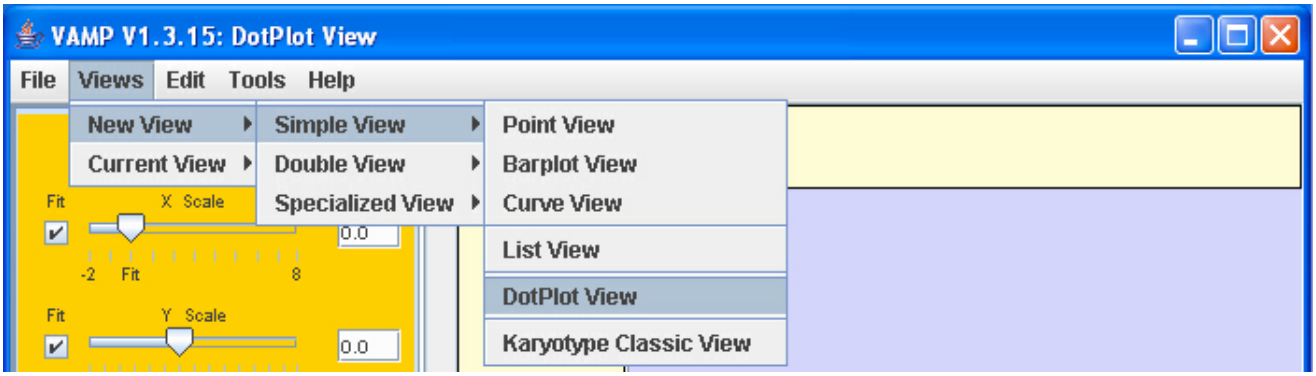

Figure 15: `Views` → `Simple View` → `DotPlot View`.

It displays a collection of samples as a heat map (see **Figure 16**) based on the level of signal for each clone or using the `Gain / Loss Color Code` (**Figure 19**).

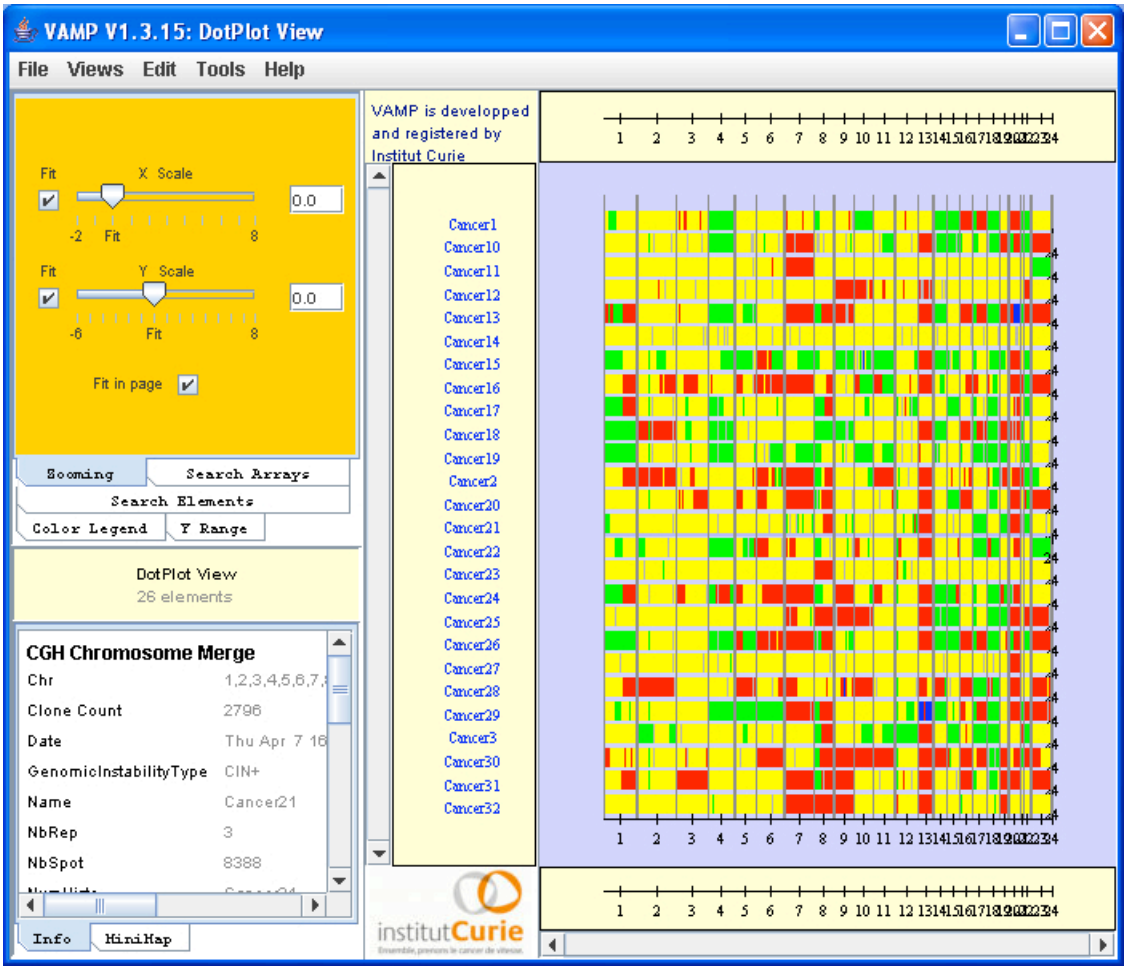

Figure 16: **Dot Plot View.**

5.2.4 List View

This view lists the names of all the arrays currently loaded and can be used for selecting or keeping track of the data under study.

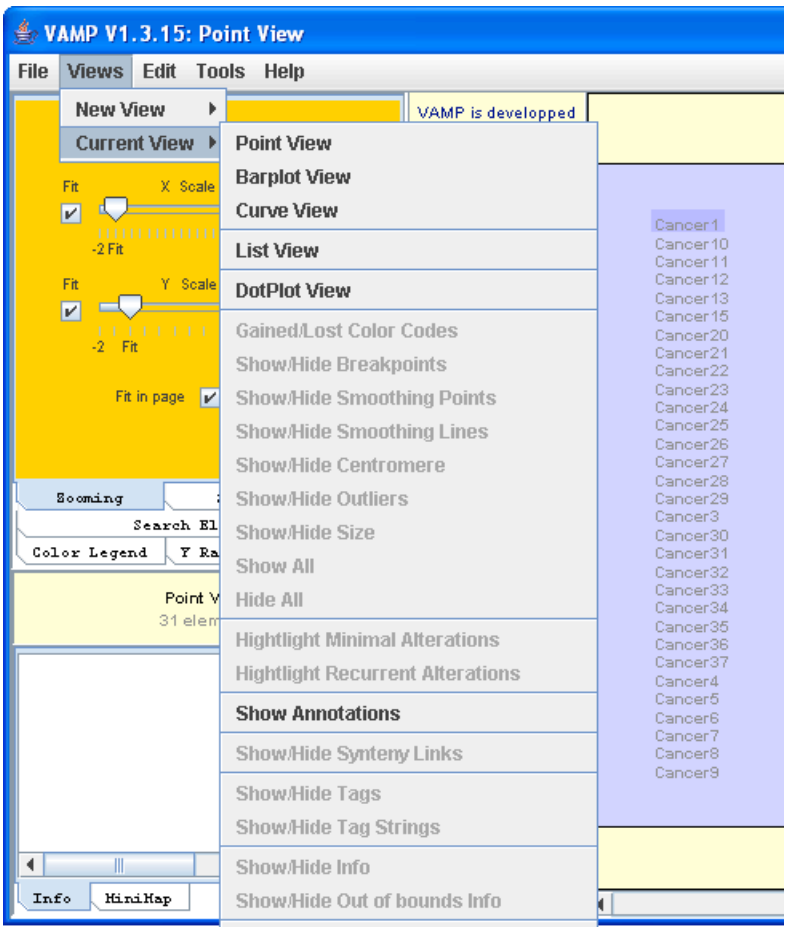

Figure 17: `Views` → `Current View` → `List View`.

5.2.5 Display additional features

Additional features regarding array CGH profiles can be displayed within VAMP, provided that these features are available in the *Array XML files* (see 3.4.2). For example, the profiles could have been preprocessed by any segmentation algorithm like GLAD (Hupé et al., 2004), the following additional features are available within the menu `View` → `Current View`:

- `Gain / Loss Color Code`: each clone is colored according to its status (loss in green, normal in yellow, gain in red and amplicon in blue)
- `Show Breakpoints`: a vertical red dashed line is plotted and represents the breakpoint location
- `Show Smoothing Line` and `Show Smoothing Points`: the piecewise constant function estimated by the segmentation algorithm is plotted
- `Show Outliers`: outliers (see page 12 for a definition of outliers) are black circled.

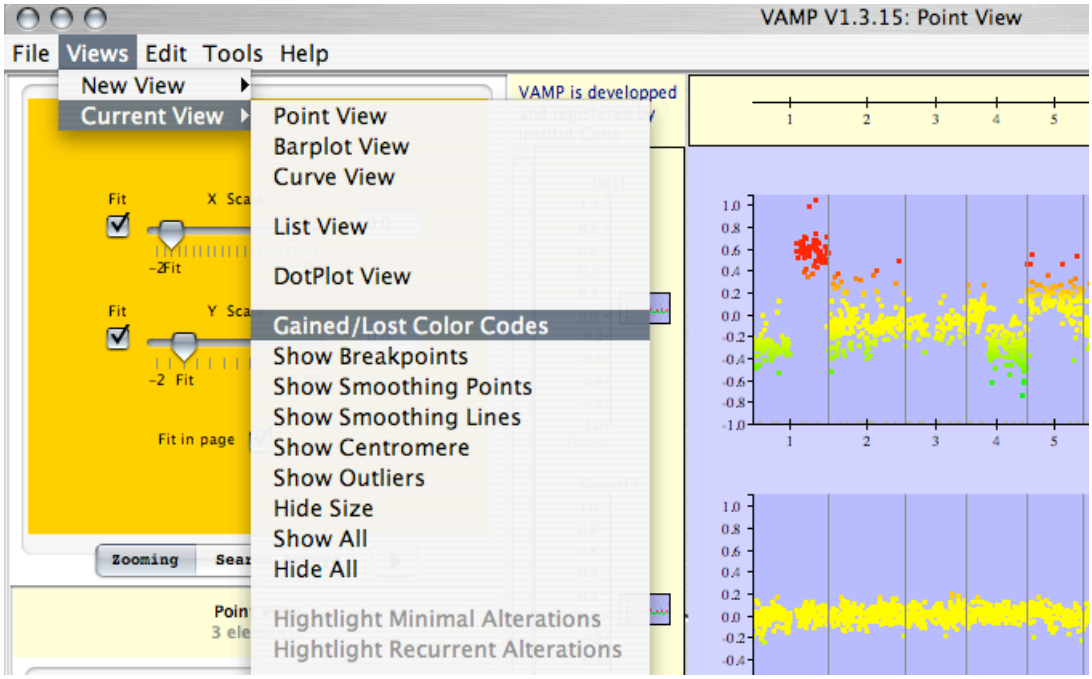

Figure 18: `Views` → `Current View`.

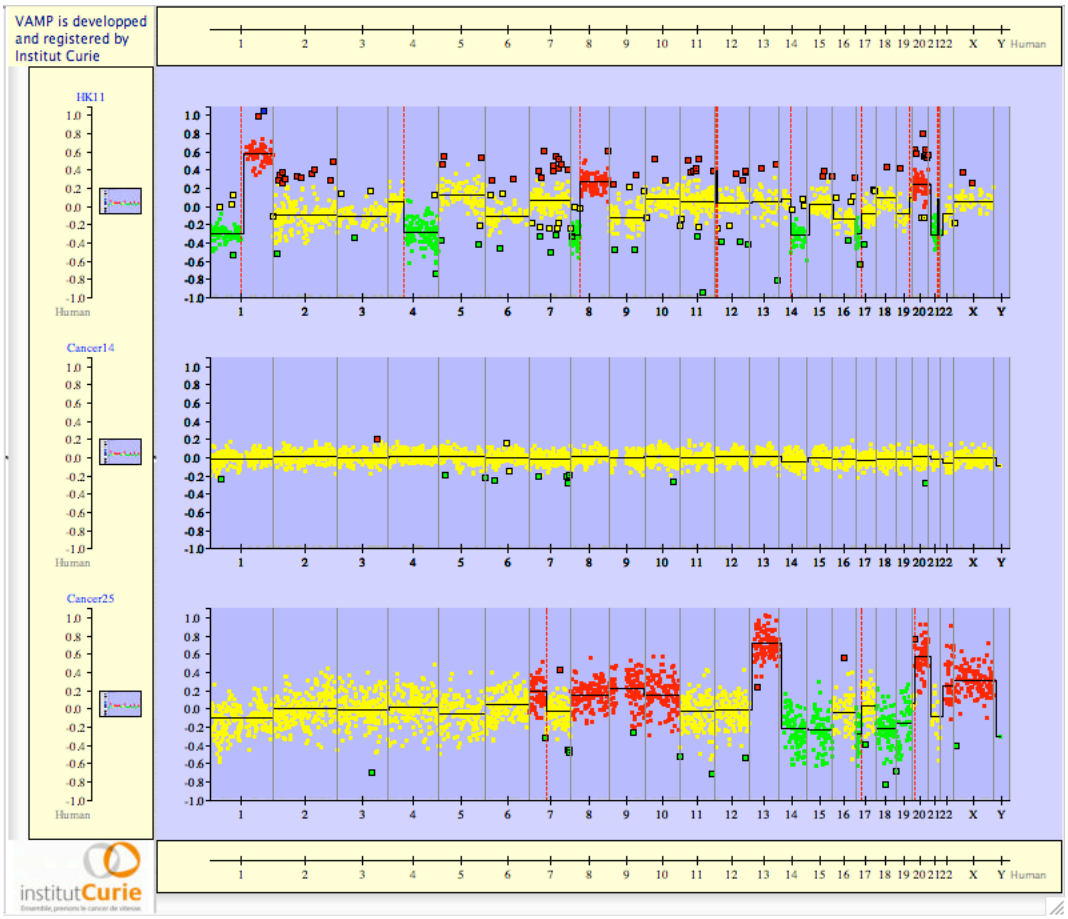

Figure 19: **Additional information with** `Gained/Lost Color Codes`, `Show Breakpoints`, `Show Smoothing Lines`, **and** `Show Outliers`.

5.3 Search

A function is available to search for any array-level of probe-level information in the data.

5.3.1 Search for arrays

The user can search for arrays matching his criterion, such as: `Name = Cancer11`.

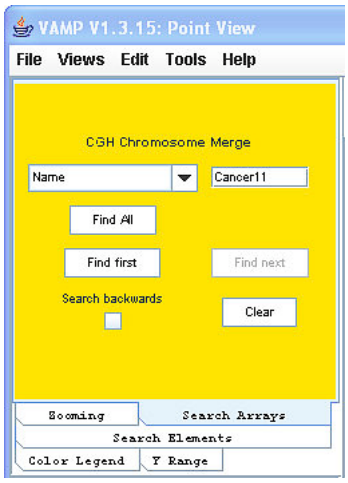

Figure 20: **Search Arrays Panel.**

The first array matching the search criterion is highlighted in the main frame.

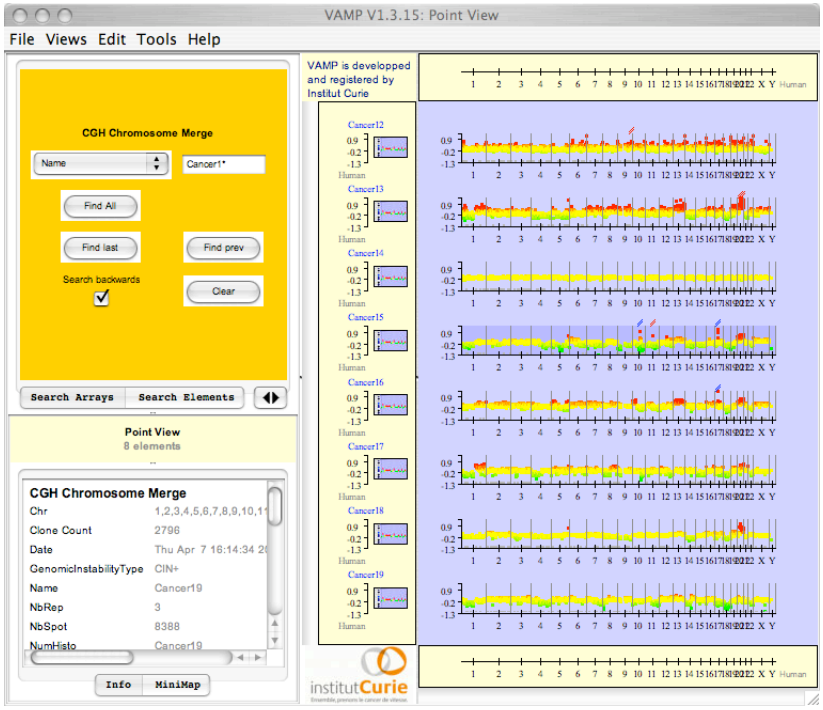

Figure 21: **Search Arrays Panel and array highlighted in the main frame.**

5.3.2 Search for elements

The user can search for elements (or objects, this refers to the probes on the array, e.g. clones) matching his criterion, such as: `Name = RP11-140M23`.

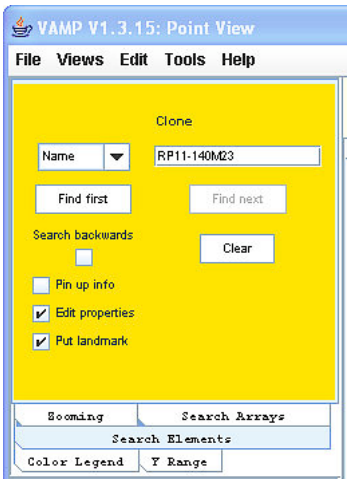

Figure 22: **Search Elements Panel.**

The first clone matching the search criterion is highlighted in the main frame. It is possible to simply pass from the first clone found to the following. Three check buttons control the search output (`Pin up info`, `Edit properties`, `Put landmark`), as shown in figure 23.

Figure 23: **Search Elements Panel.**

5.4 Visual rendering

5.4.1 Zoom

The user can zoom in or zoom out on molecular profiles (zoom scale is logarithmic). `Fit` checkboxes force the horizontal and/or vertical scale to fit the window size.

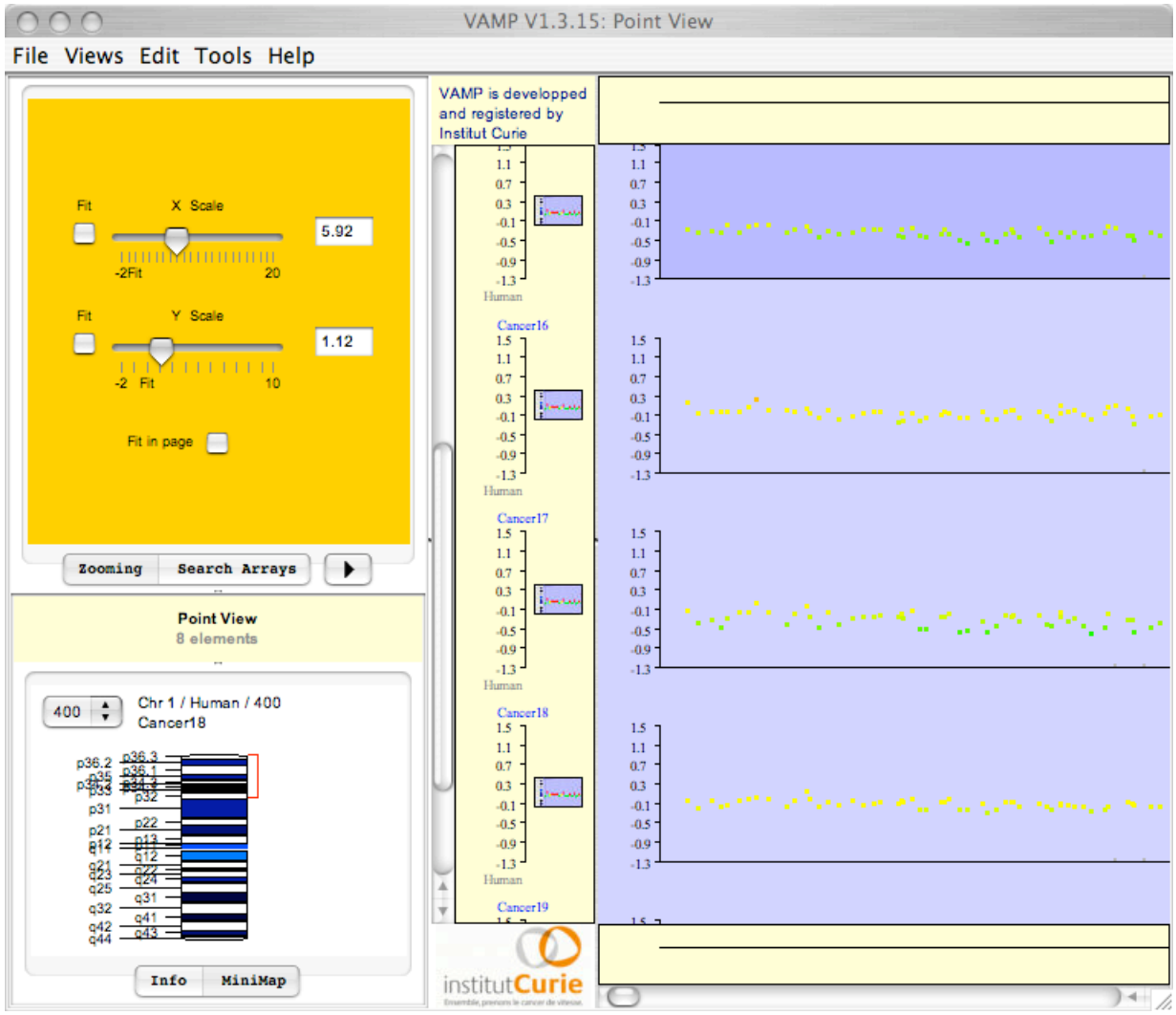

Figure 24: **Zooming Panel.**

5.4.2 Color codes

By default, points or barplots are colored according to the signal intensity (generally using ratios of the two channels or log-ratios) using a continuous scale from red to yellow to green. The user can easily change the thresholds and colors of the clone ratios, either for one particular array (`Local` option) or for all arrays displayed (`Global` option).

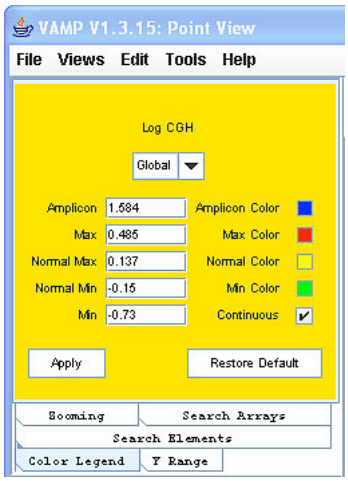

Figure 25: **Color Legend Panel.**

In the following example, the user is setting the color code for amplicons.

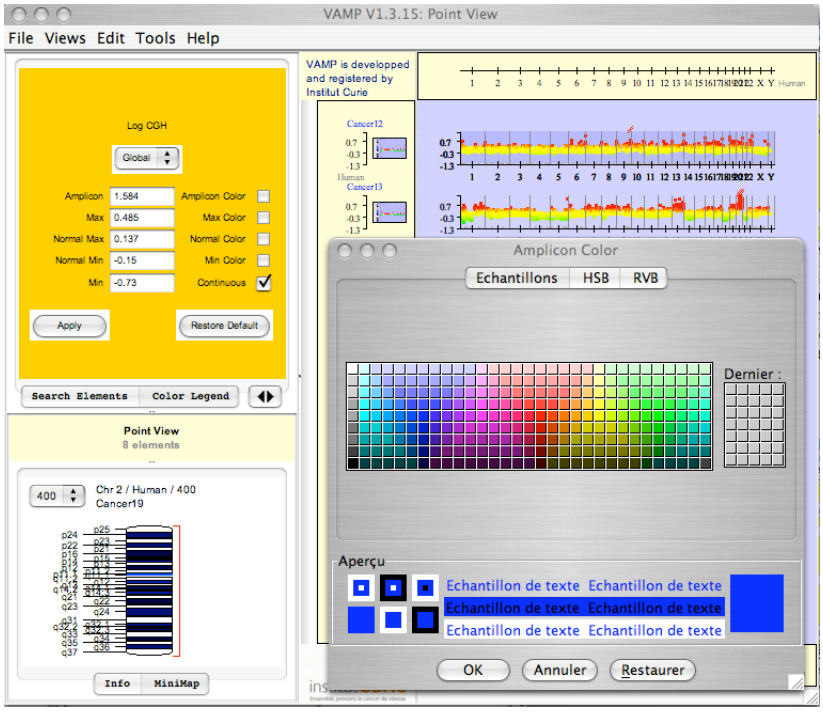

Figure 26: **Color Legend Panel.**

For each of the views described in section 5.2, molecular profiles can be colored according to the results of array-CGH data analysis (see section 5.2.5).

5.4.3 X and Y scaling

The user may change the vertical range, either for one particular array (`Local` option) or for all arrays displayed (`Global` option).

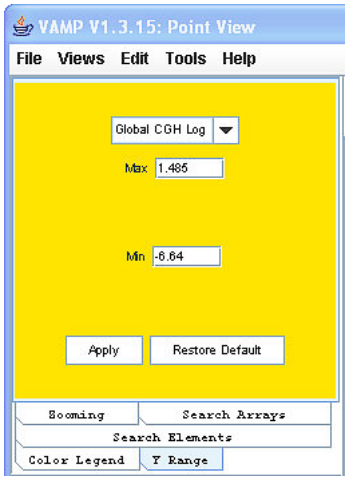

Figure 27: **Y Range Panel.**

In the following example the user widened the visible range, so that all amplicons are displayed.

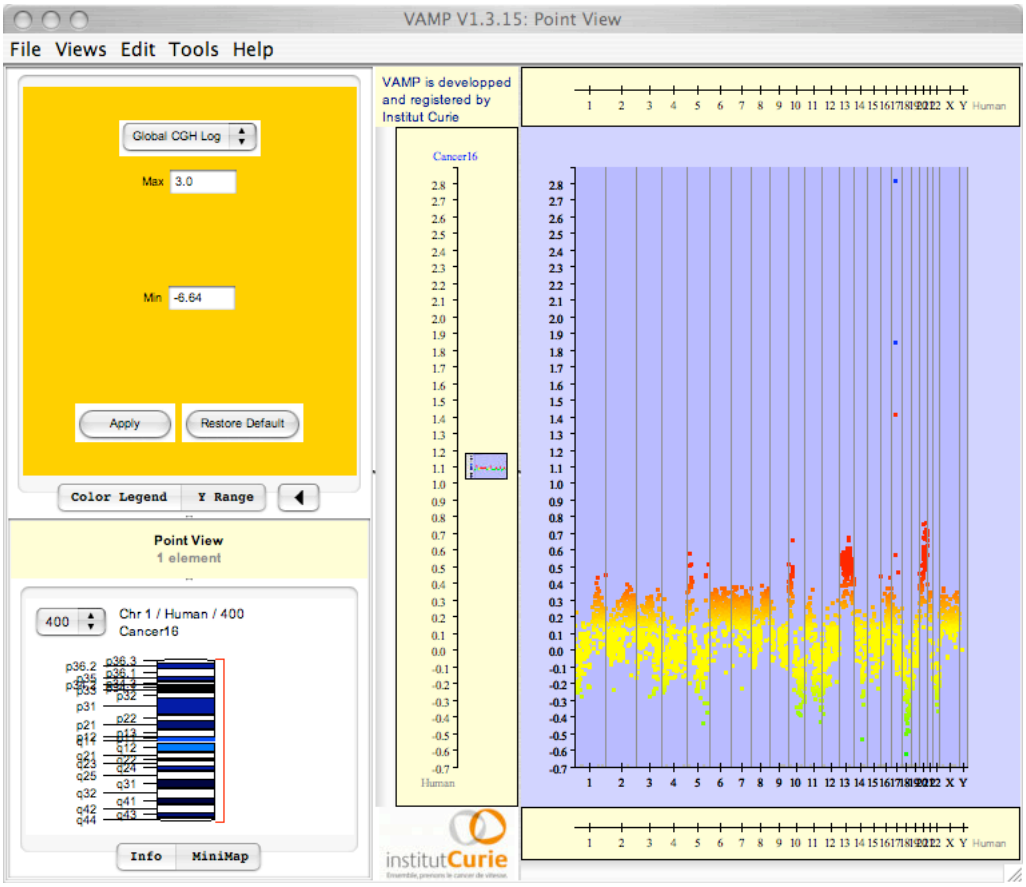

Figure 28: **Y Range Panel.**

5.5 Print

An advanced printing function is offered, either in visible mode (only the profiles that are visible on the screen are printed), or in global mode (all profiles in the view are printed). A template is offered for defining the output of the printing (this can, for example, include several frames in an arbitrary composition, to which text or images can be added). It can be used for defining and printing standardized outputs. The user can also interactively monitor the print preferences. In any case, the print function is WYSIWYG through an intermediate preview which can be edited by the user. The print menu `File` → `Print` → `Standard Report Template` opens the default printing template, and allows the user to print or export into PNG format.

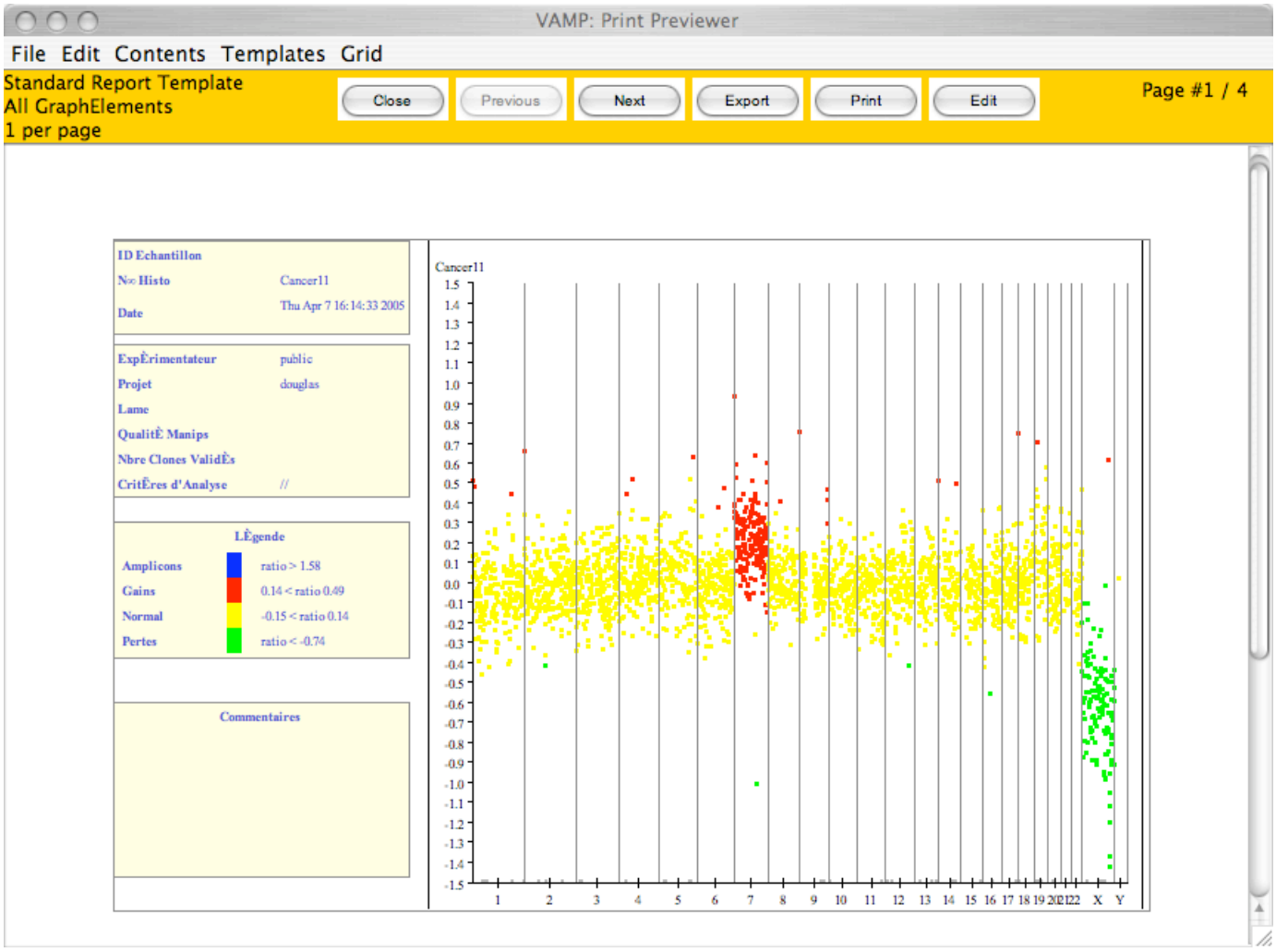

Figure 29: `File` → `Print` → `Standard Report Template`

¿From this default template it is possible to customize templates, and to save them for later use. *After switching to 'Edit' mode* (by clicking on the `Edit` button), you can add (or remove) several types of objects, such as text, molecular profiles, images, ... These objects can be dragged and dropped inside the template.

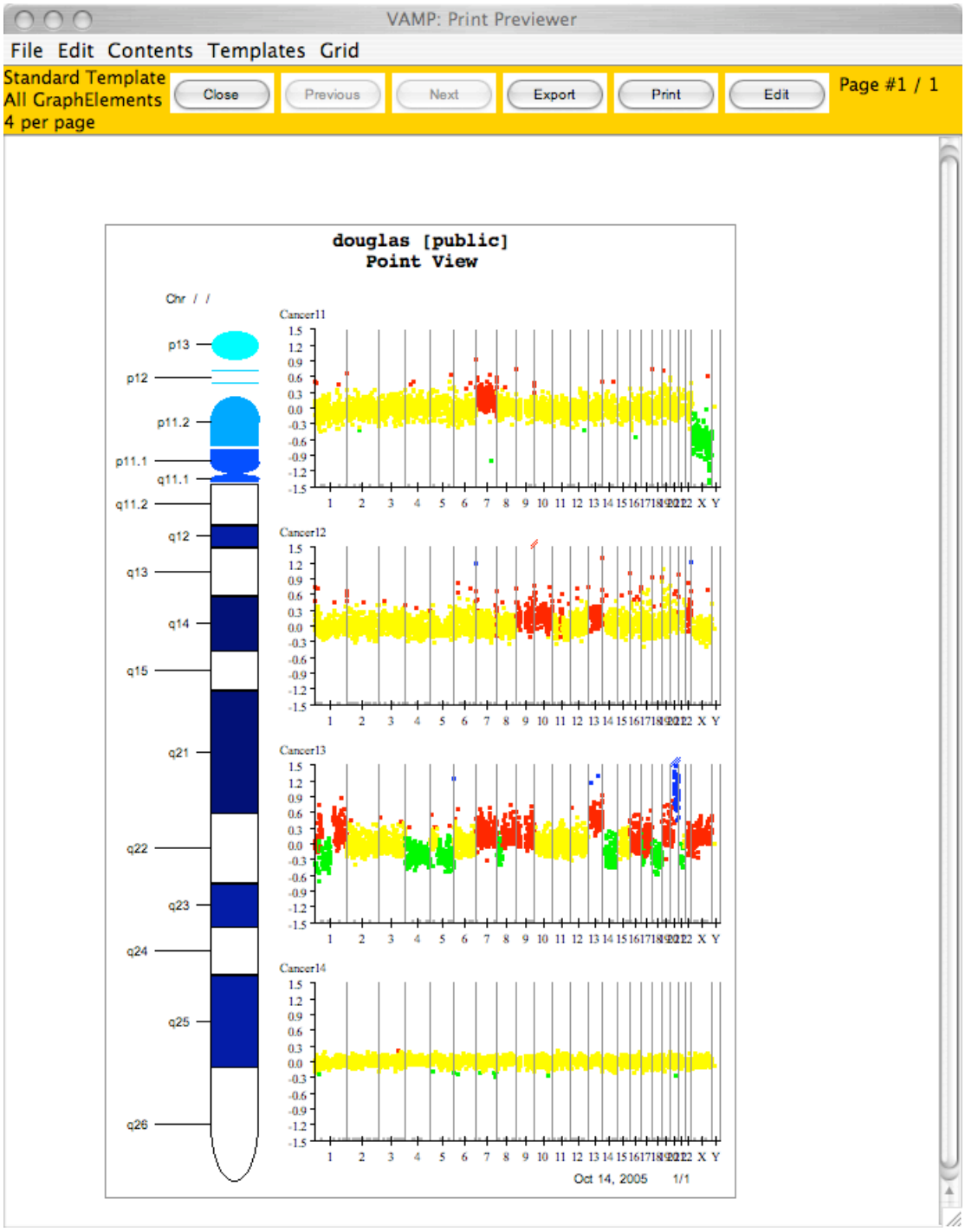

Figure 30: **Customized template** - A minimap has been added to the standard template.

The menu `File` → `Print` → `Print preview` loads the last used template.

5.6 Save and load

The session can be saved in XML files. Reloading the file later on allows the continuation of the analysis within the context of the previous work, or the exchange of results and data with colleagues.

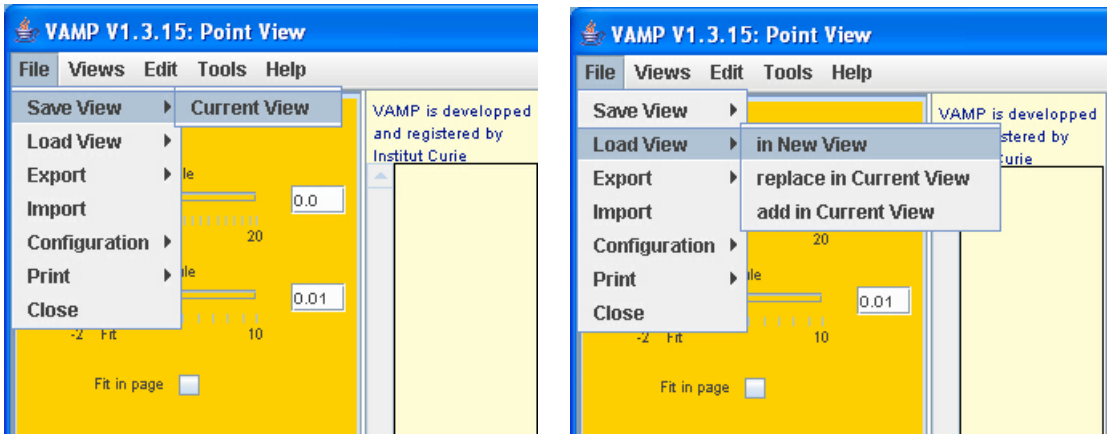

Figure 31: **Save and Load menus**

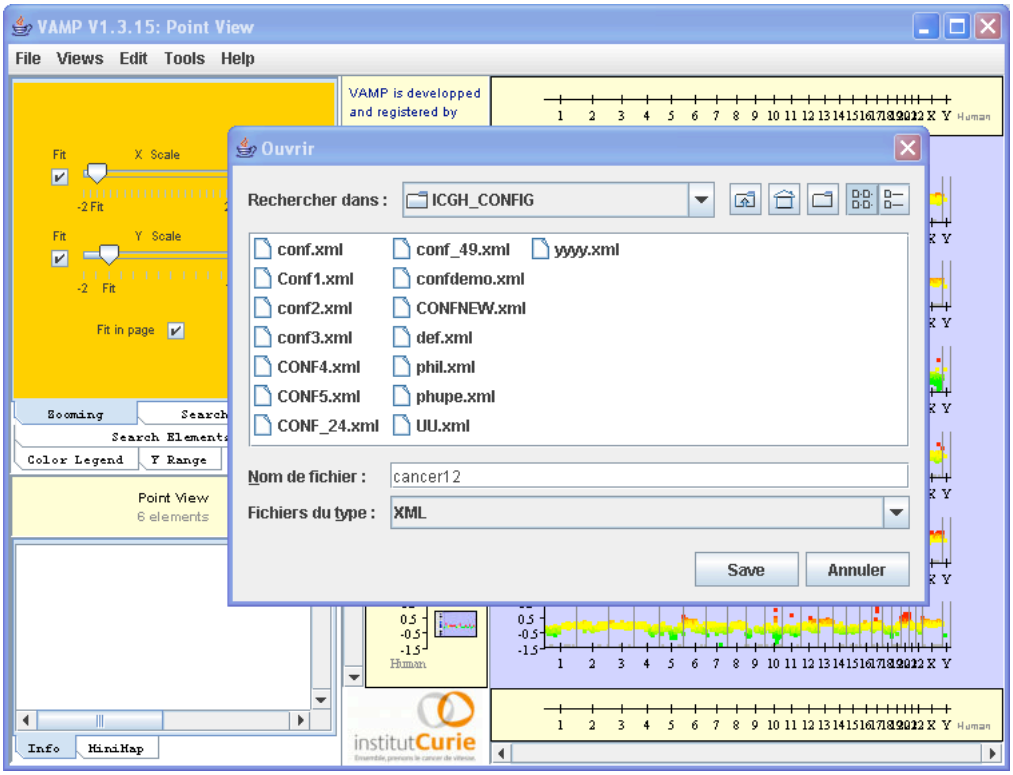

Figure 32: `Save dialog window.`

There are three possibilities to load a saved session: `in New View`, `replace in Current View` or `add in Current View`.

6 Data analysis

The VAMP software allows a large variety of analysis to be performed by the users. Some of the tools provided are profile-specific, e.g. tools dedicated to array-CGH or transcriptome data analysis. Other generic tools can be used with any type of molecular profile. You will find below a list of the main analysis tools available within VAMP.

6.1 Manual analysis

Among the numerous functions of VAMP, the user has the possibility to put his own marks and regions. This is done by right-clicking on clone/probe element under the mouse pointer of any molecular profile. A menu appears as shown in **Figure 33** allowing various actions to be performed:

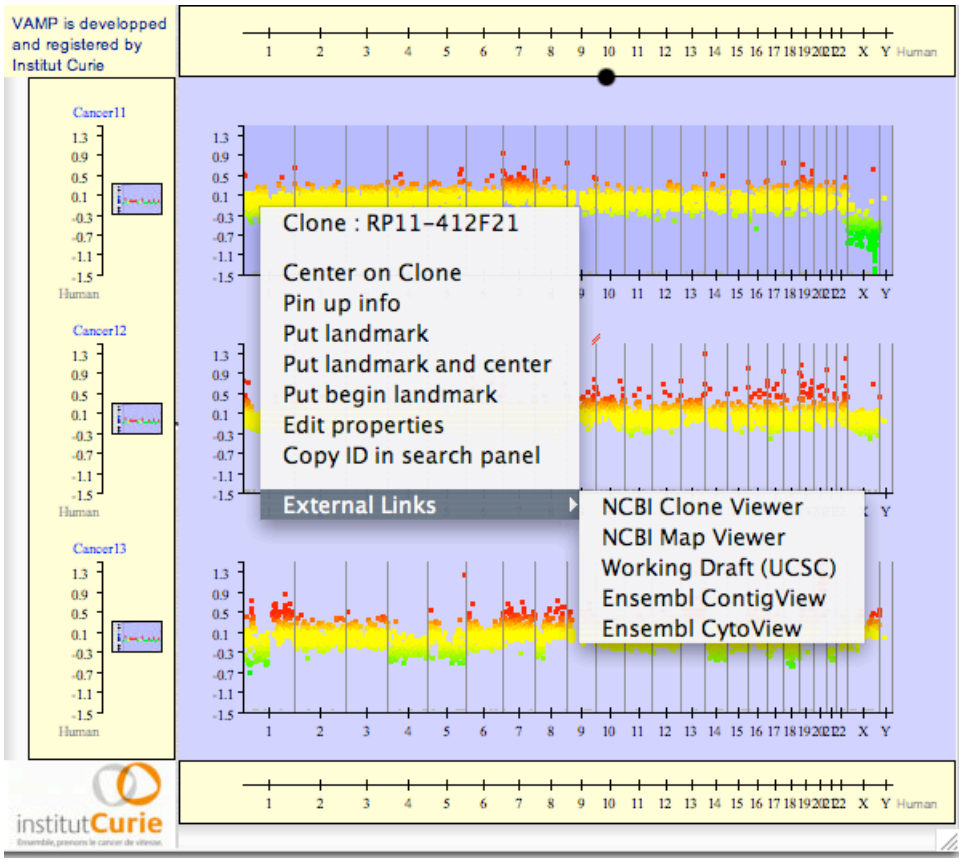

Figure 33: **Contextual Menu** - The user can easily put landmarks and regions of biological interest.

Within the contextual menu, it is possible:

`Center on Clone/Probe`: to center the profile around the current position.

`Put landmark`: to draw a vertical bar trough all the profiles to define a locus.

`Put begin landmark`, `Put end landmark`: to draw a region over the set of profiles.

`External links`: to retrieve any information from public or local databases.It is possible to add your favorite database (see **section 3.3.1**).

The user can customize colors by right-clicking onto landmarks and regions and then using `Set colors` for a better visualization (see **Figure 34**).

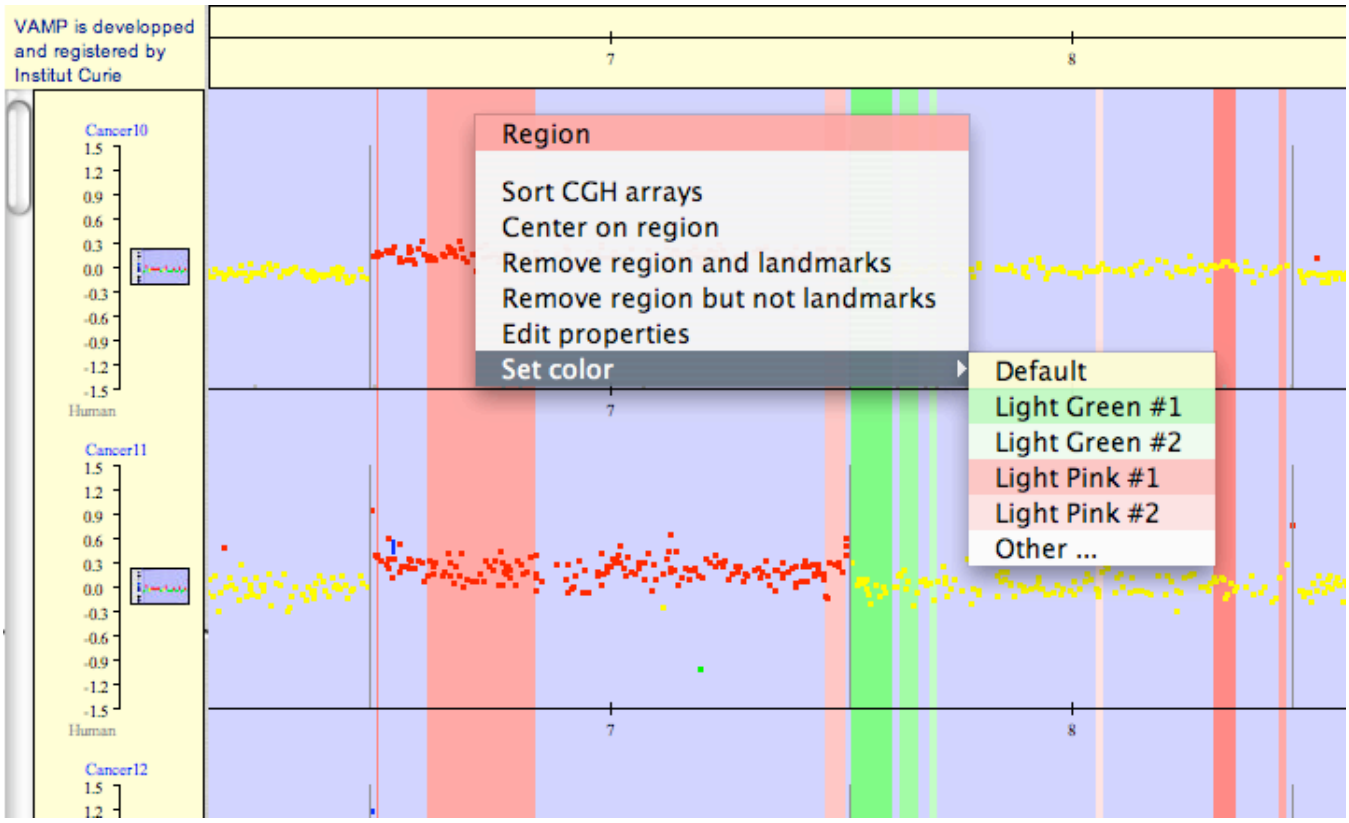

Figure 34: **Customize Analysis** - Choose your favorite color.

6.2 Finding common alterations among a collection of CGH- array profiles

Finding common alterations over a set of tumors is the main motivation of array CGH analysis. Two algorithms are proposed within VAMP. It is necessary that breakpoints, gain, loss and amplicon regions have been previously detected by any algorithm (Hupé et al., 2004). We propose two different approaches to identify such regions of biological interest:

- `Tools` → `CGH` → `Minimal Alterations`: Minimal Alterations are extracted by intersecting the profiles of many tumors and looking for a sufficient number of alterations in the tumors (this parameter is set by the user) over the smallest possible region of the profile (see **Figures 35, 36 and 37**).

- `Tools` →`CGH` →`Recurrent Alterations`: In a given tumor, an alteration is bounded by two extremities, which can be a breakpoint or a chromosome end; when an alteration is present in a sufficient number of tumors with the same extremities, it is a recurrent alteration (see **Figure 38, 39 and 40**).

For both Recurrent Alterations and Minimal Alterations, the user has to set the `Minimum support` required, that is the minimum number or percentage of tumors showing the alteration for considering it as significant.

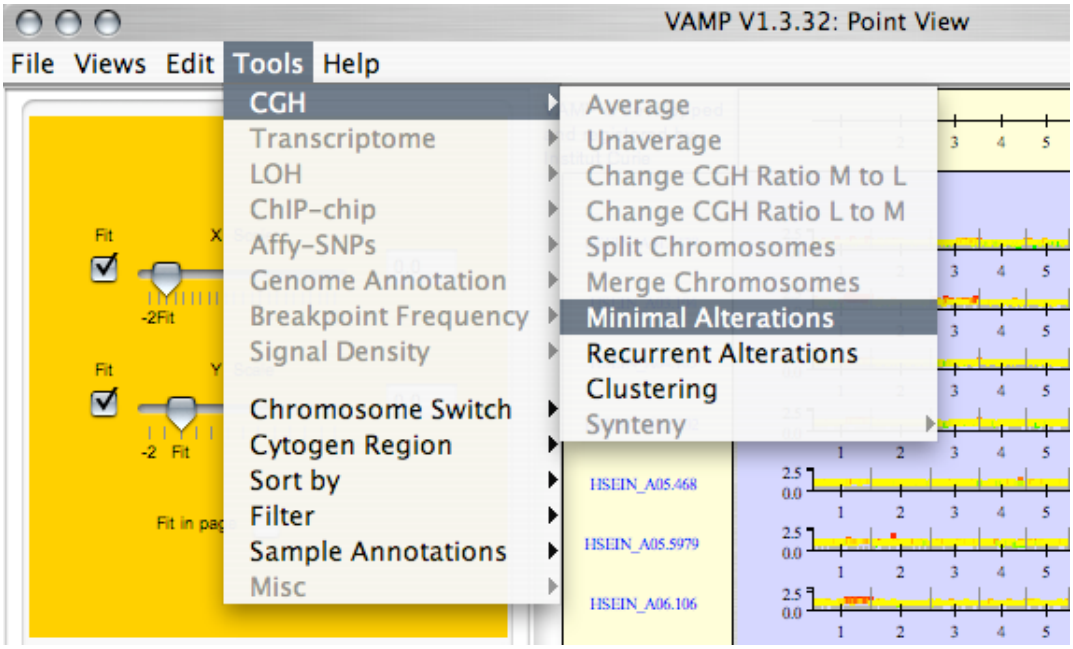

Figure 35: `Tools` → `CGH` → `Minimal Alterations` - The user opens a new dialog window to set the parameters for Minimal Alterations.

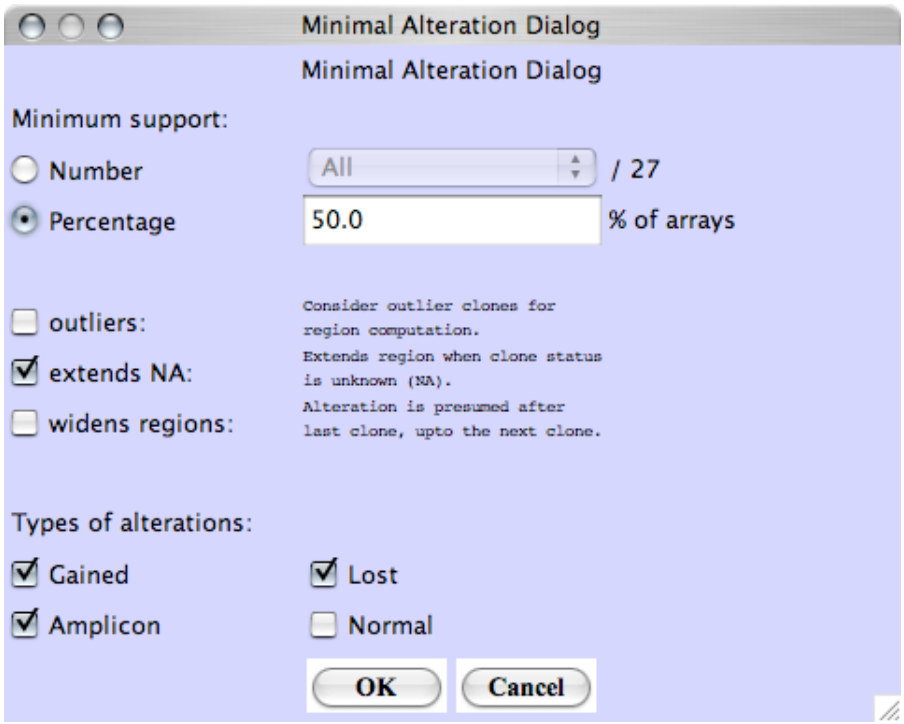

Figure 36: **Minimal Alterations** - At least 50% of the profiles must share the same alterations.

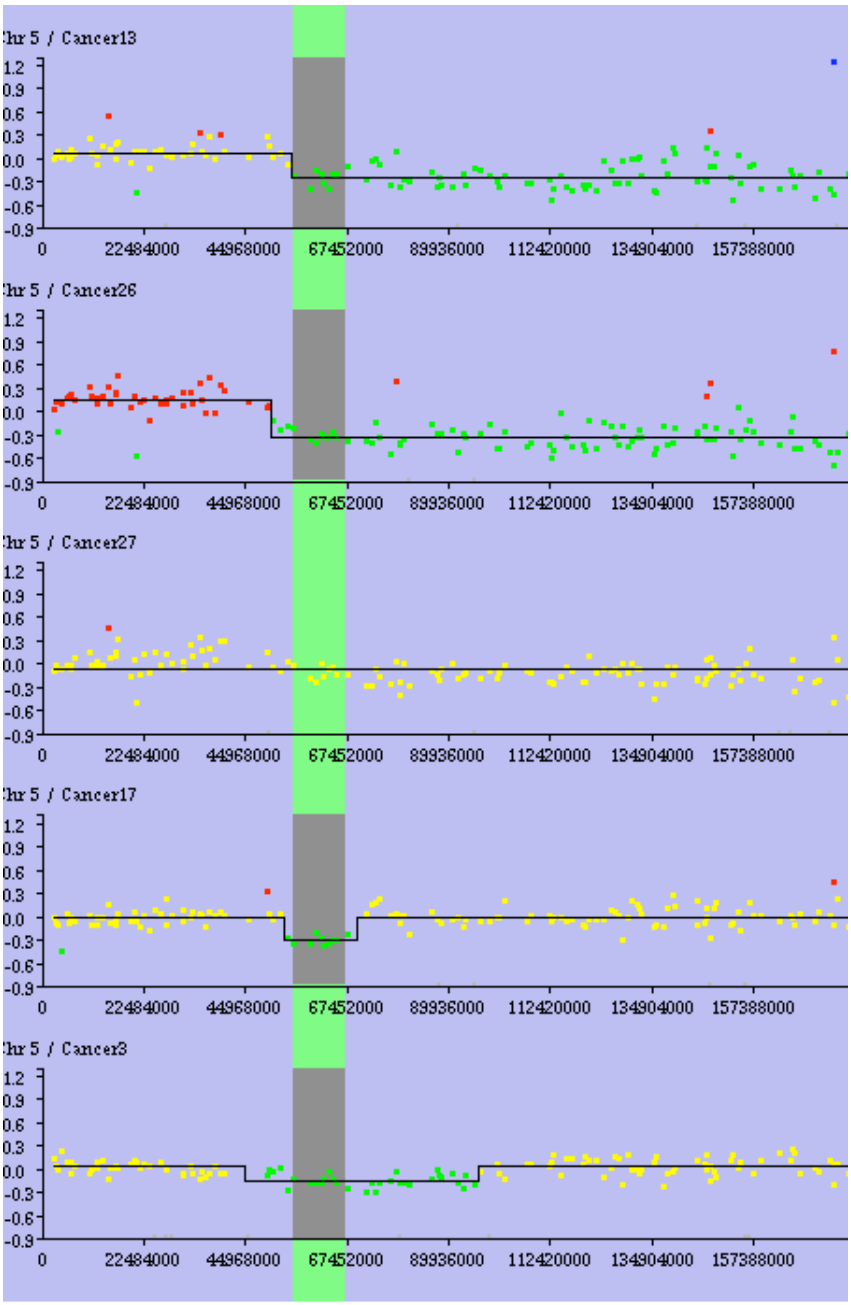

Figure 37: **Minimal Alterations** - Results : the minimal alterations are drawn in red for gain, green for loss.

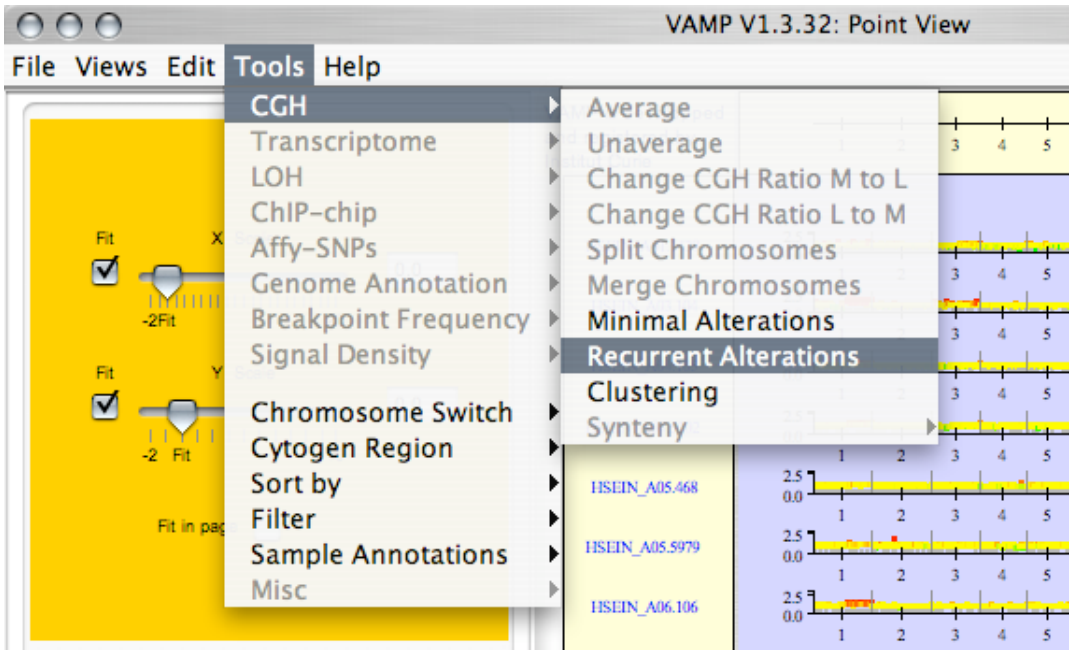

Figure 38: `Tools` → `CGH` → `Recurrent Alterations` - The user opens a new dialog window to set the parameters for Recurrent Alterations.

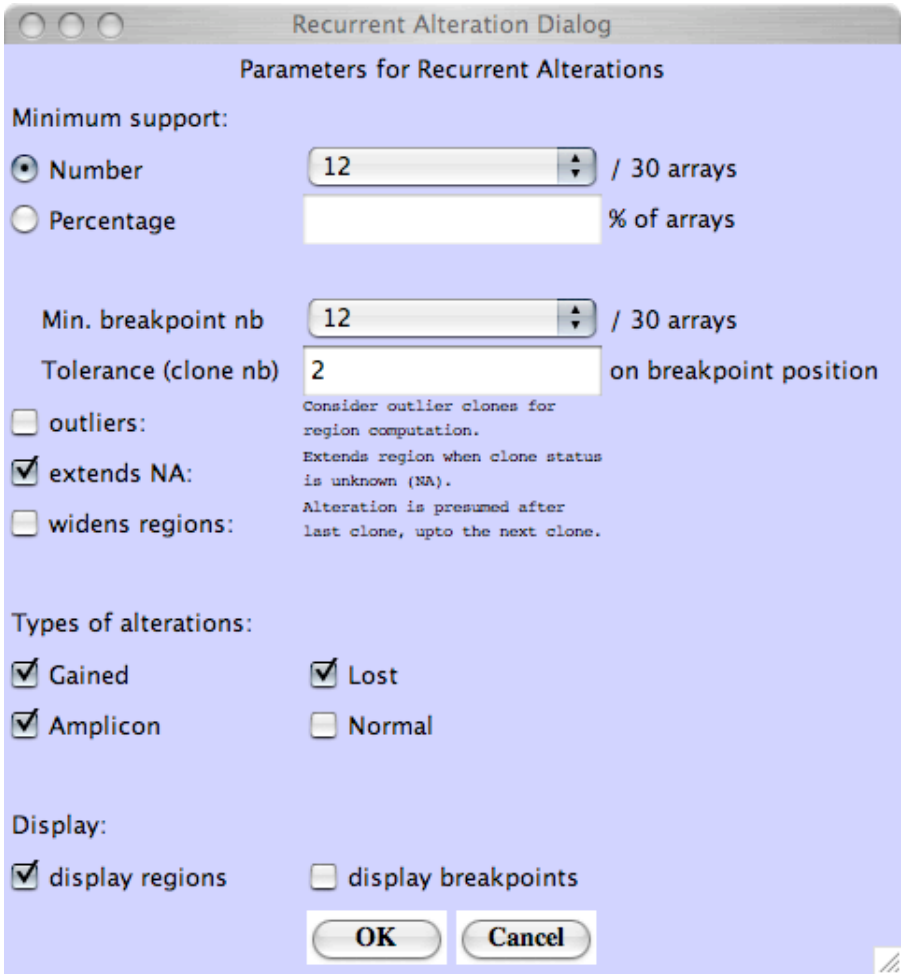

Figure 39: **Recurrent Alterations** - At least 10 out of 27 profiles must share the same alterations.

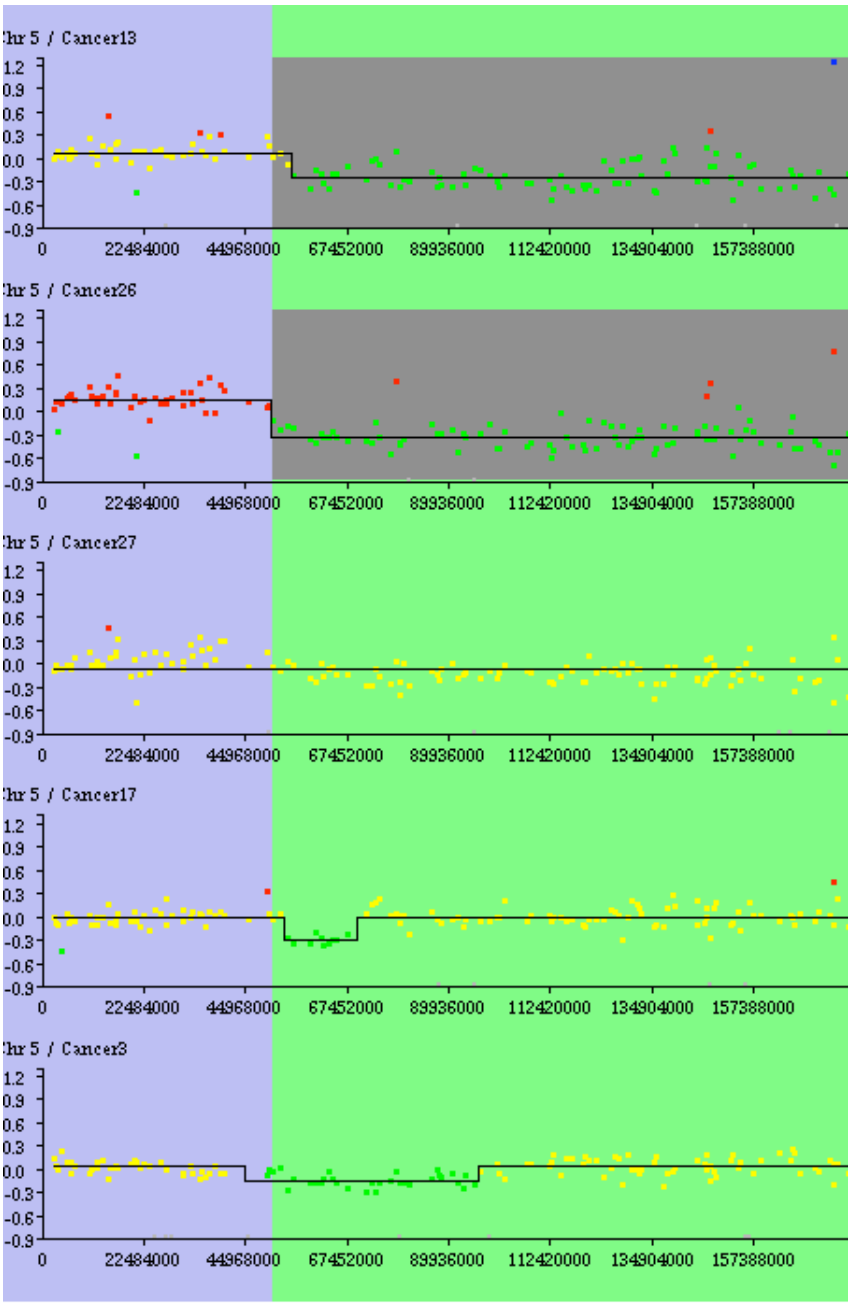

Figure 40: **Recurrent Alterations** - Results : the recurrent alterations are drawn in red for gain, green for loss.

6.3 Clustering profiles

Clustering is a general technique for unsupervised data classification widely used in microarray data analysis. A VAMP function offers the possibility to perform a hierarchical clustering (Kaufman and Rousseuw, 1990) on the array CGH profiles.

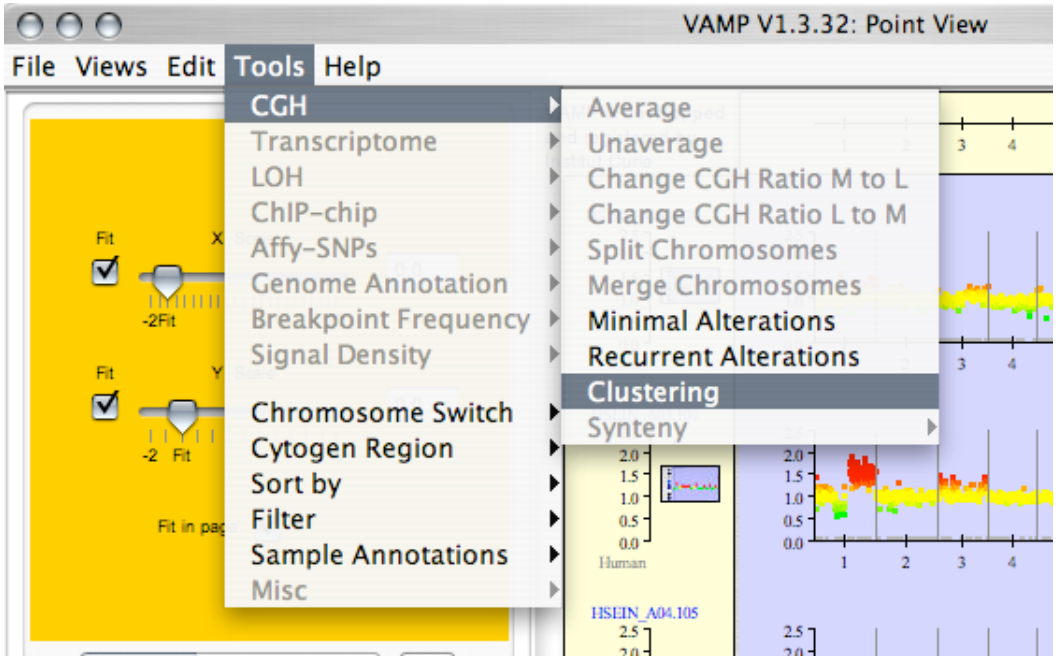

Figure 41: `Tools` → `CGH` → `Clustering` - The user can open a new window of dialog for clustering.

The clustering can be performed on different variables:

- *Clone LogRatio:* The Clone LogRatio values of the whole genomic profile are used
- *Clone Smoothing:* The Clone smoothing values (i.e. the results of a segmentation algorithm) of the whole genomic profile are used
- *Clone Status:* The Clone statuses (i.e. the results of a segmentation algorithm) of the whole genomic profile are used
- *Regions Status:* Regions either selected manually or identified by our algorithm (see **section 6.2**) are used.

Different options are available:

- *Distance metric:* Euclidian, Pearson and Manhattan distance are available
- *Group metric:* Ward, Single linkage and Average linkage are available

VAMP displays the results as a cluster view including a heat map and the trees resulting from the clustering algorithm (**Figure 43**).

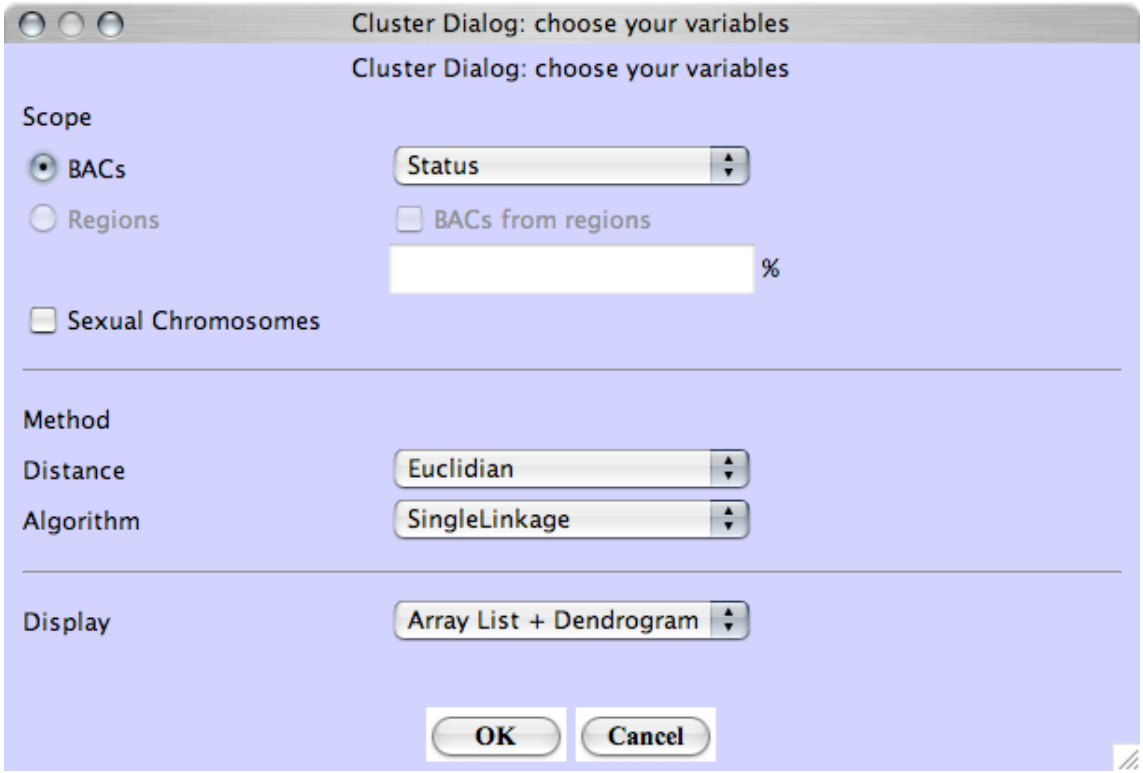

Figure 42: **Clustering profiles** - Different clustering options are available.

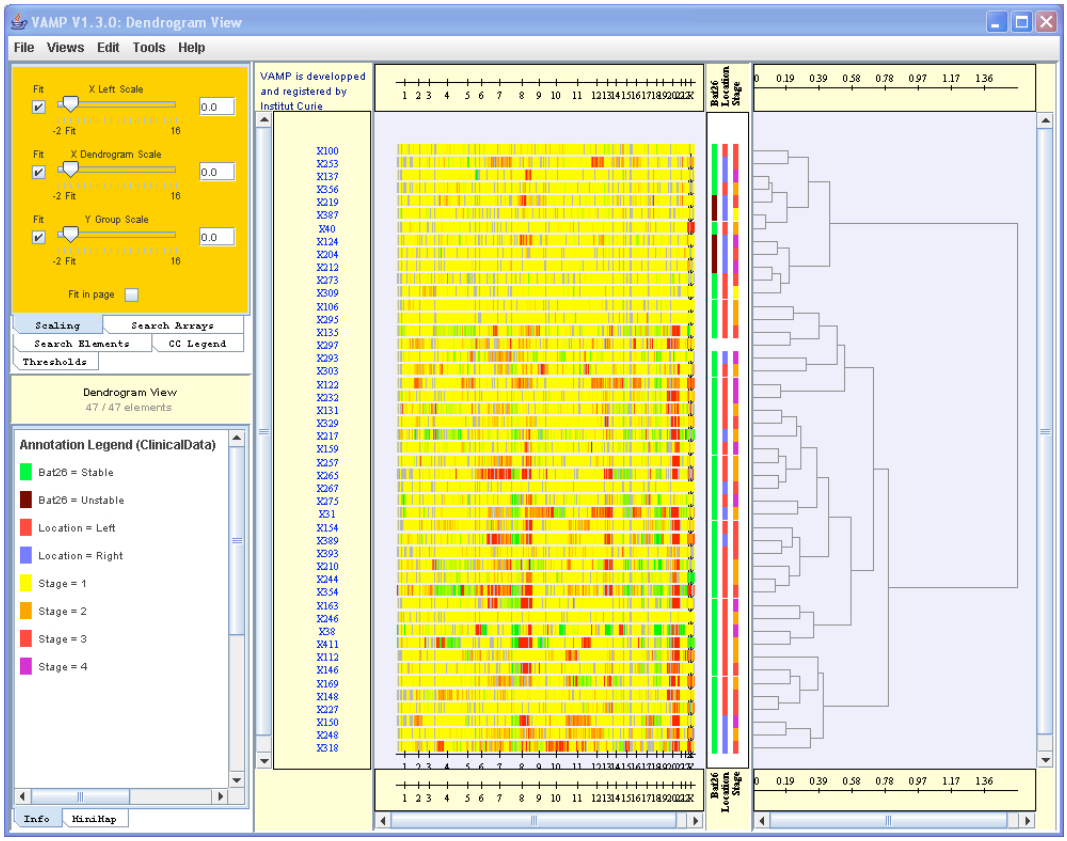

Figure 43: **VAMP interface** - Dotplot view of array-CGH profiles (middle panel), and dendrogram resulting from a hierarchical clustering (right panel). In between, color-coded clinical information about the samples, with a legend (bottom left). Data from Nakao et al. (2004)

6.4 Comparing profiles

VAMP proposes several data manipulation procedures for the profiles such as loading any type of profile (CGH, expression, LOH, ChIP chip) for a given sample. A typical application of VAMP is the simultaneous visualization of the DNA alterations and gene under- and over-expression in a region, **Figure 45**). In the case of one-color microarrays (such as Affymetrix) the user can define a reference profile and compute the ratio with a test profile: he can easily compare the ratio of gene expression with the DNA alteration.

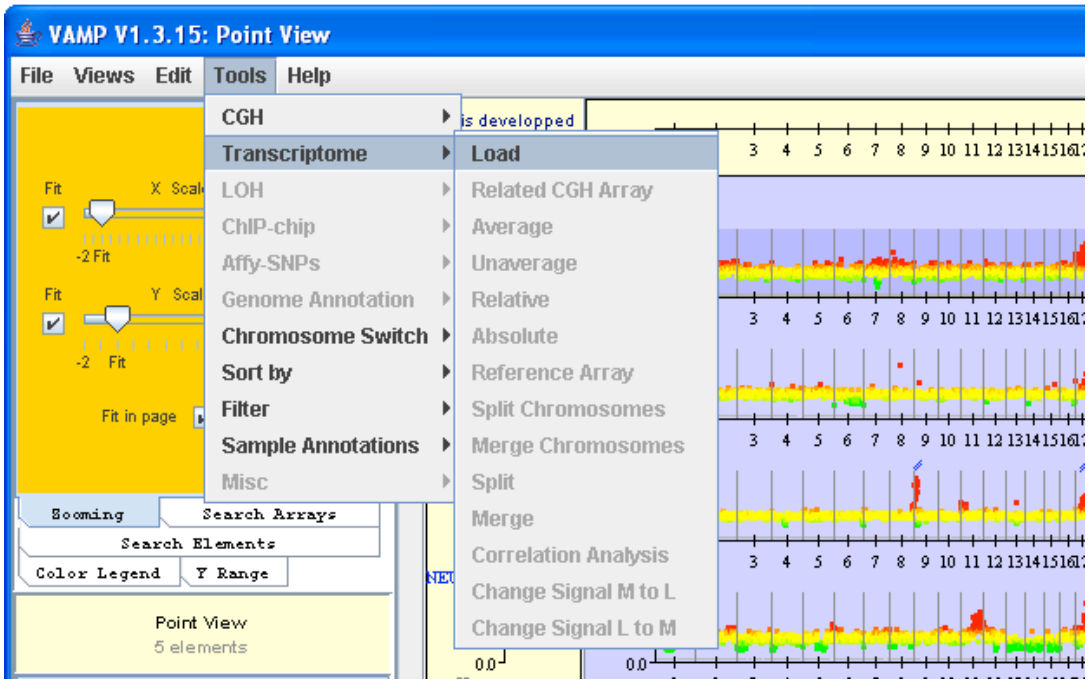

Figure 44: **Transcriptome Load** - The user can load expression profiles The user can load and visualize one expression profiles which is connected to Array-CGH.

In the example below we compare array-CGH (top profile) versus transcriptome ratio (second profile in descending order), computed for Affymetrix U95 array of a bladder tumor sample and of a reference sample. This confrontation pinpoints the probable implication of the oncogene cyclin D1 in this tumor. The third and fourth profiles in descending order correspond to a reference profile (average normal bladder tissue profile) and the profile of the tumor under study, respectively. The second profile is the ratio of the fourth to the reference profile.

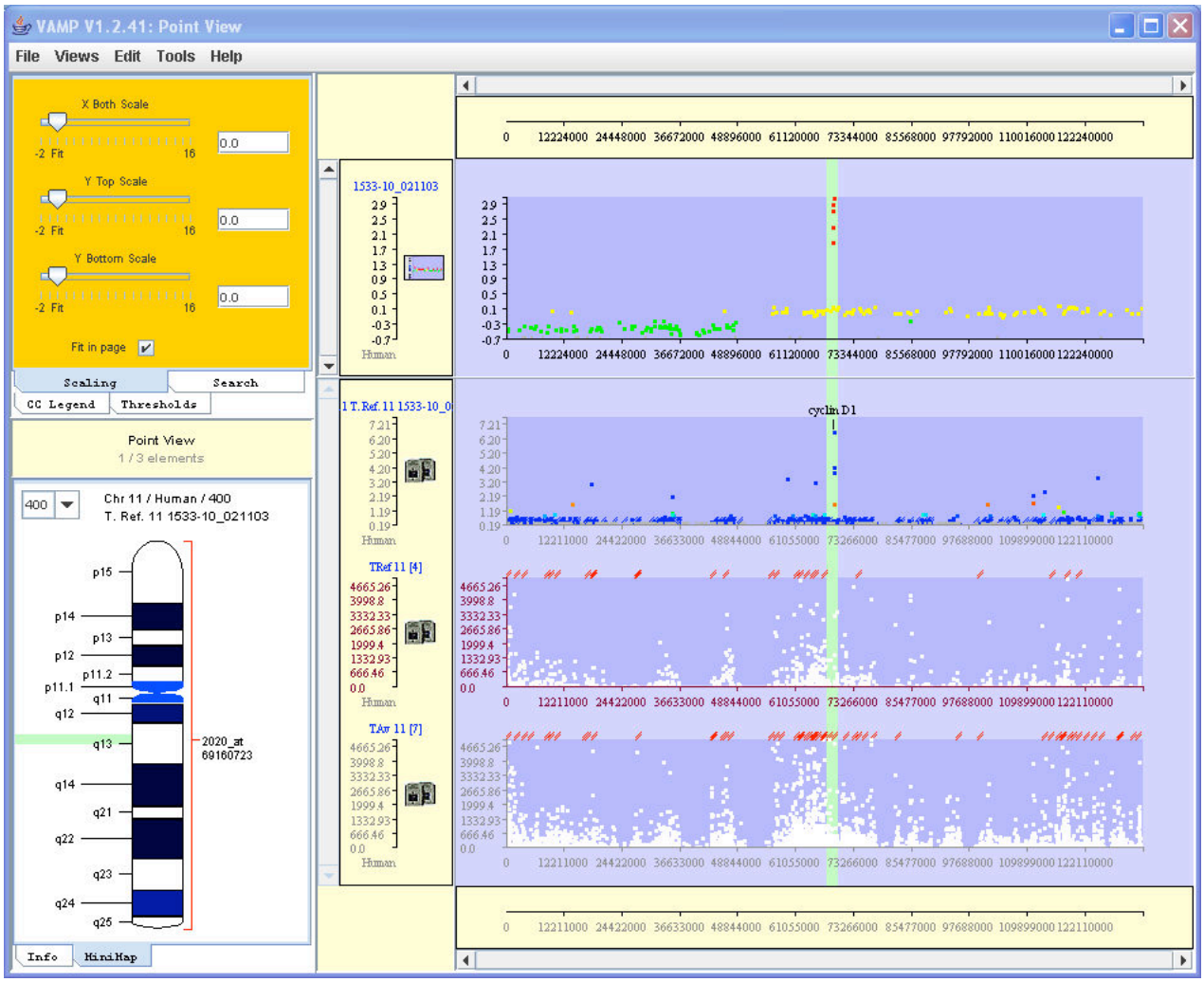

Figure 45: **CGH vs Transcriptome.** An icon at the left of each profile shows the type of loaded profile.

6.5 Confrontation with sample annotation

Clinical data, or any other sample annotations present in the additional XML files (see **section 3.4.3**) can be visualized in the interface or used for filtering tumors or for sorting them (the link between additional data and molecular profile is based on the XML tag `<NumHisto>` representing a unique patient ID (see **section 3.4.2**). To add any clinical properties in the current view, just do the following:

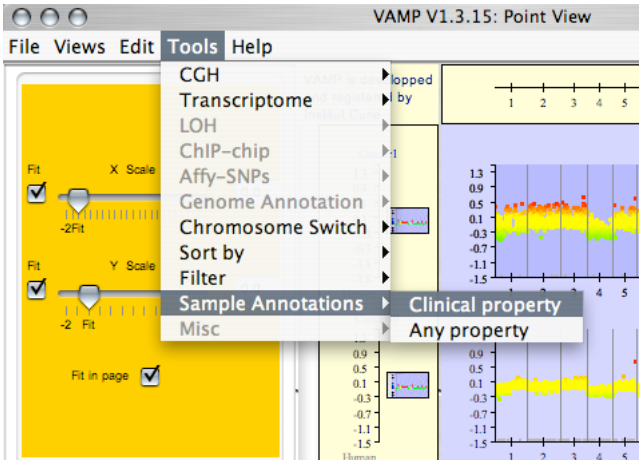

Figure 46: `Tools` →`Sample Annotations` →`Clinical property` - The user opens a dialog box where he can easily choose the available clinical properties.

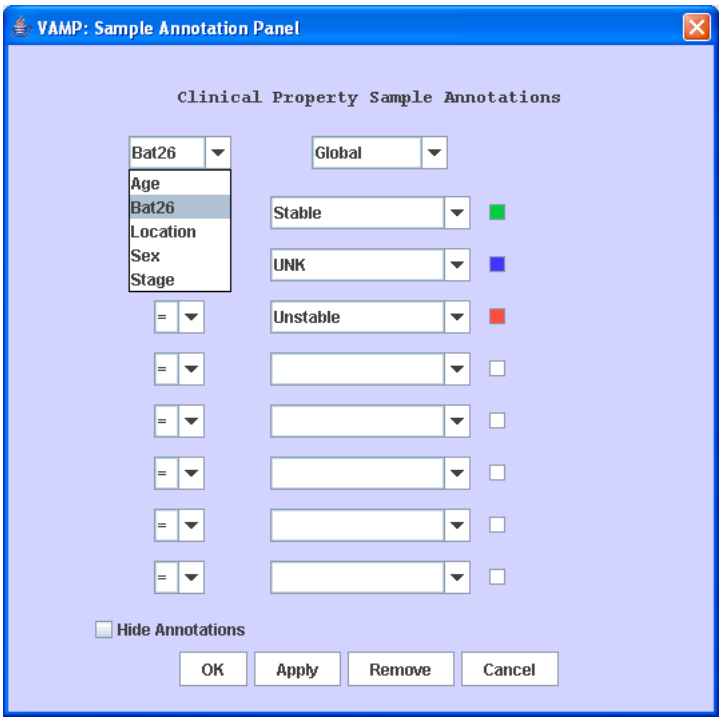

Figure 47: **Sample annotations** - The user chooses clinical properties to be visualized.

This data can be visualized as color-coded bars in an annotation frame on the left of the profiles, and can be easily compared with a clustering result (**Figure 48**).

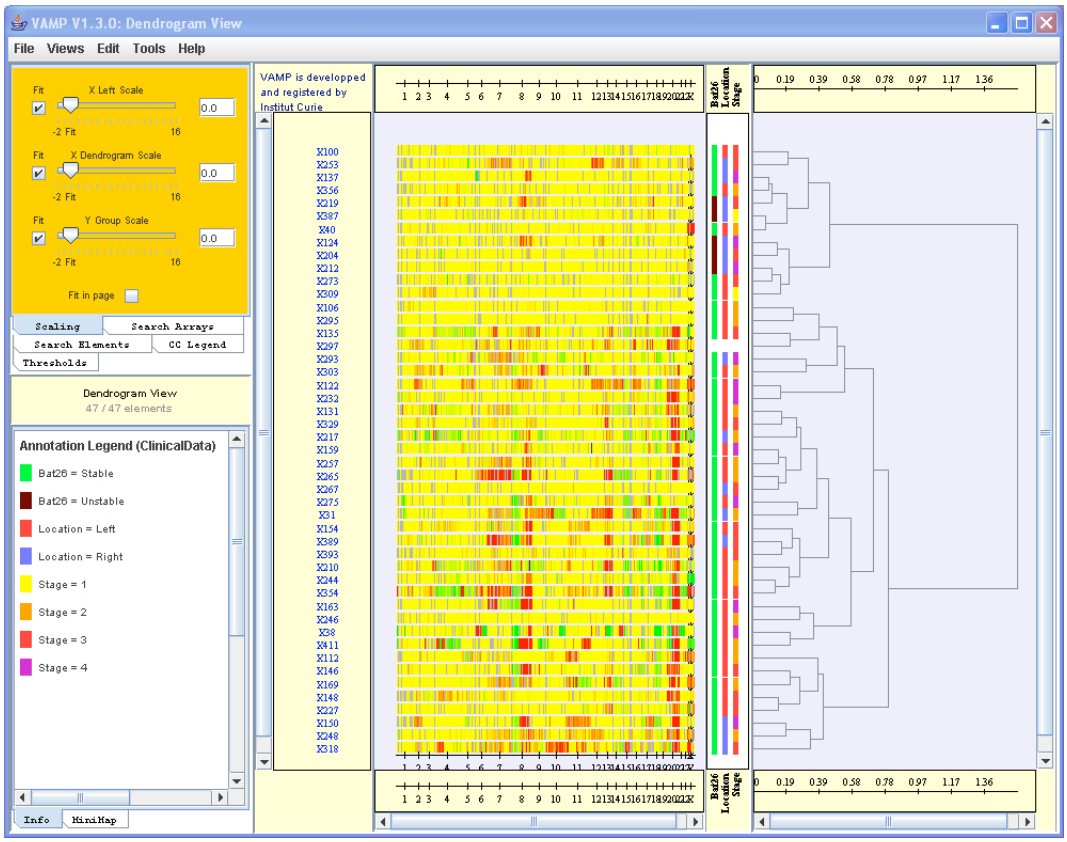

Figure 48: **VAMP interface** - Dotplot view of array-CGH profiles (middle panel), and dendrogram resulting from a hierarchical clustering (right panel). In between, color-coded clinical information about the samples, with a legend (bottom left). Data from Nakao et al. (2004)

6.6 Synteny analysis

VAMP can display the syntenic projection of a profile onto the genome of another species, which serves as a reference; a typical application is the projection of a mouse array-CGH profile onto the human genome (**Figure 52**). In such a case, the mouse clones are ordered according to their mapping onto the human genome. VAMP uses pre-computed information mapping each clone of the sample array onto the reference genome. The synteny relationships can be shown, for a selection of regions of the genome, as links from each clone of the profile to the location of the most similar sequence of the reference genome.

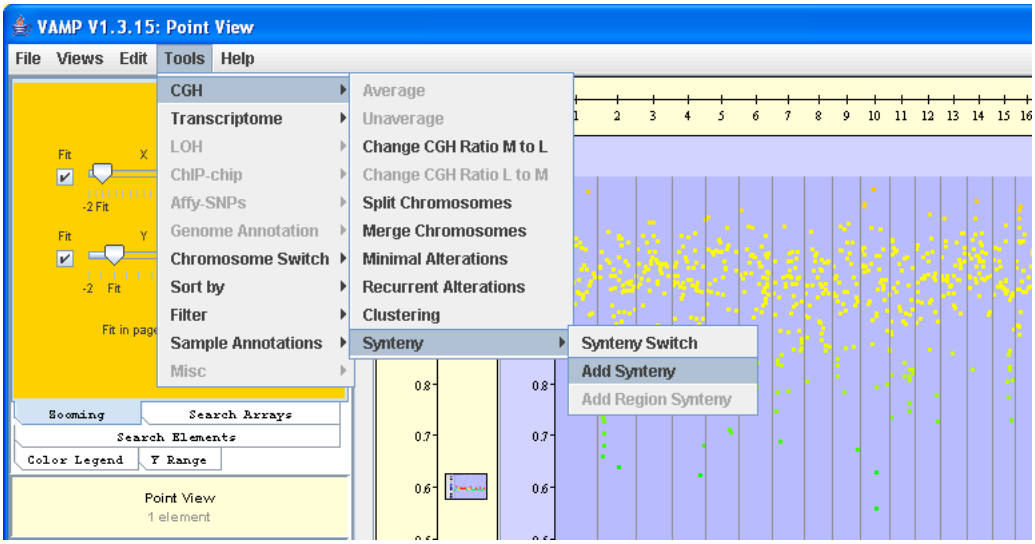

Figure 49: `Switch` **or** `Add` - The user can replace a profile by its projection onto another species genome by using `Synteny` → `Switch` or he can add the projection below the original profile in the same window by using `Synteny` → `Add`.

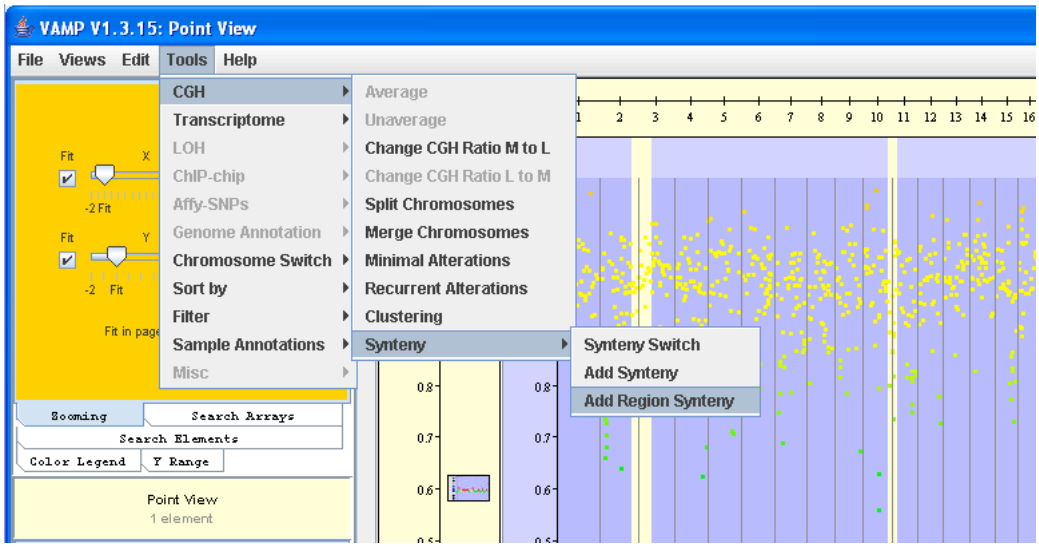

Figure 50: `Add Region Synteny` - The user can projects only those clones that are comprised in the regions he has defined by using `Synteny` → `Add Region Synteny`. Results are shown in the next figure.

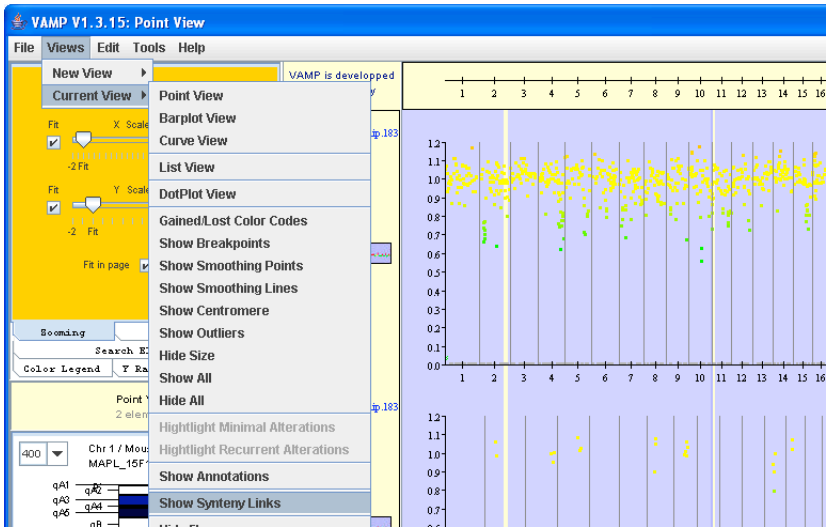

Figure 51: `Show Synteny Links` - The user can visualize the links of synteny using `Current View` → `Show Synteny Links`. Results are shown in the next figure.

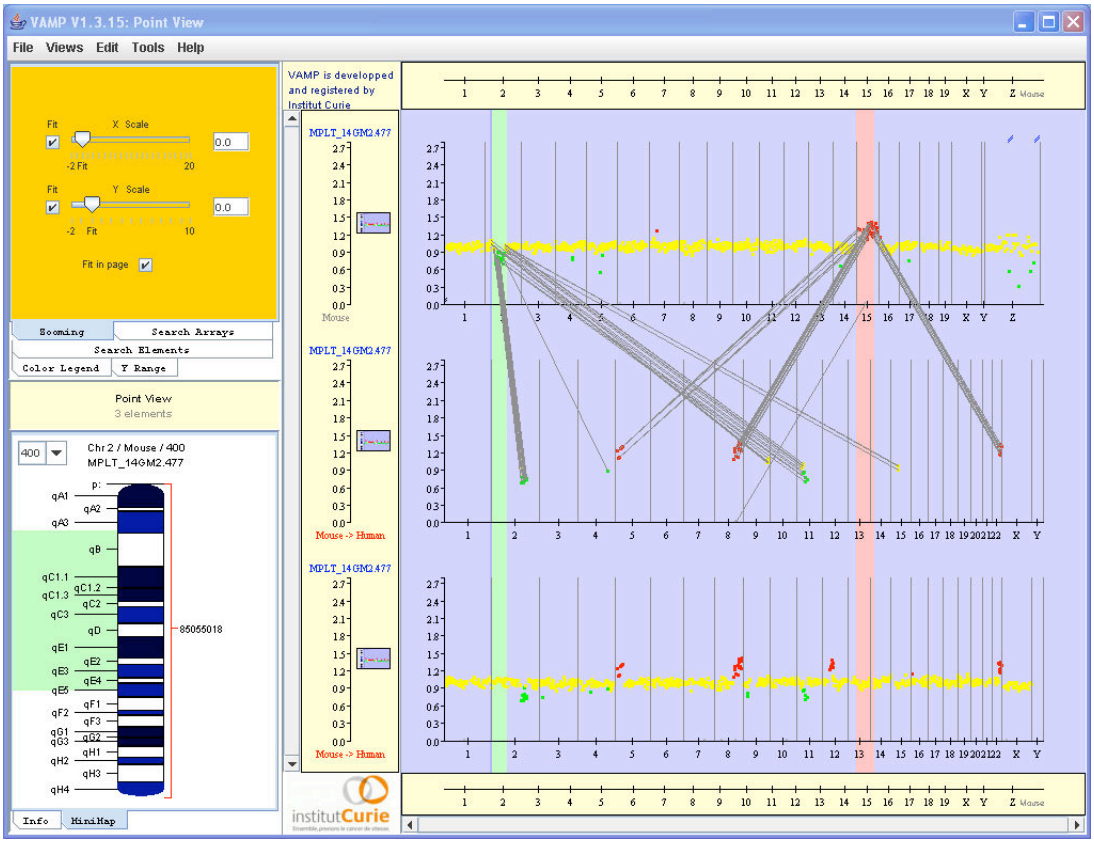

Figure 52: **Synteny visualization** - A complete example of visualization of synteny. The original mouse profile is on the top, its projection onto the human genome at the bottom, and the projection of the clones from the two highlighted regions in the middle, with links from the mouse-ordered profile to the human projection.

6.7 Confrontation with Genome Annotation

With VAMP it is possible to compare a molecular profile with genome annotations, for example gene structure: the user can load a pseudo-profile with the structure of all known genes (introns, exons, splicing variants). Several functions are offered. Below are some examples:

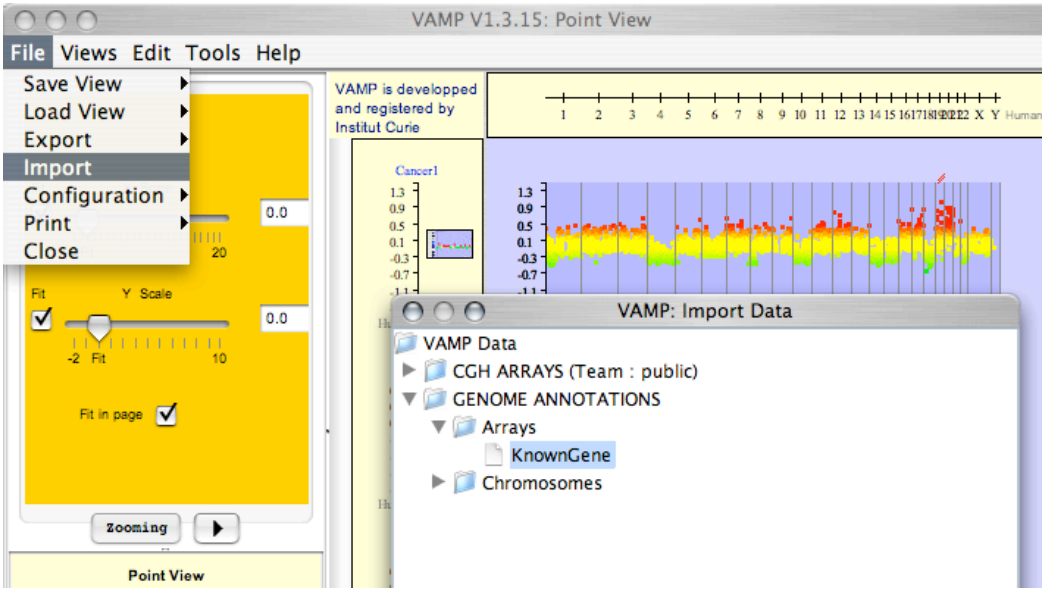

Figure 53: `File` → `Import` - Start by importing genome annotation data.

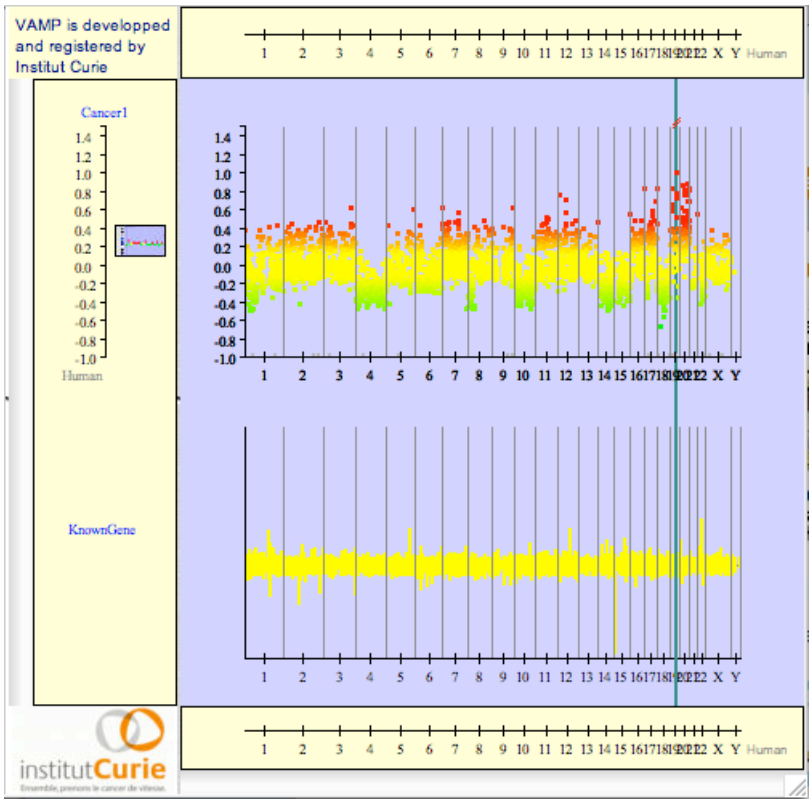

Figure 54: **Genome Annotation** - Then you can put a landmark on your favorite gene and center on it.

Then you can zoom on the landmark until the gene structure is visible.

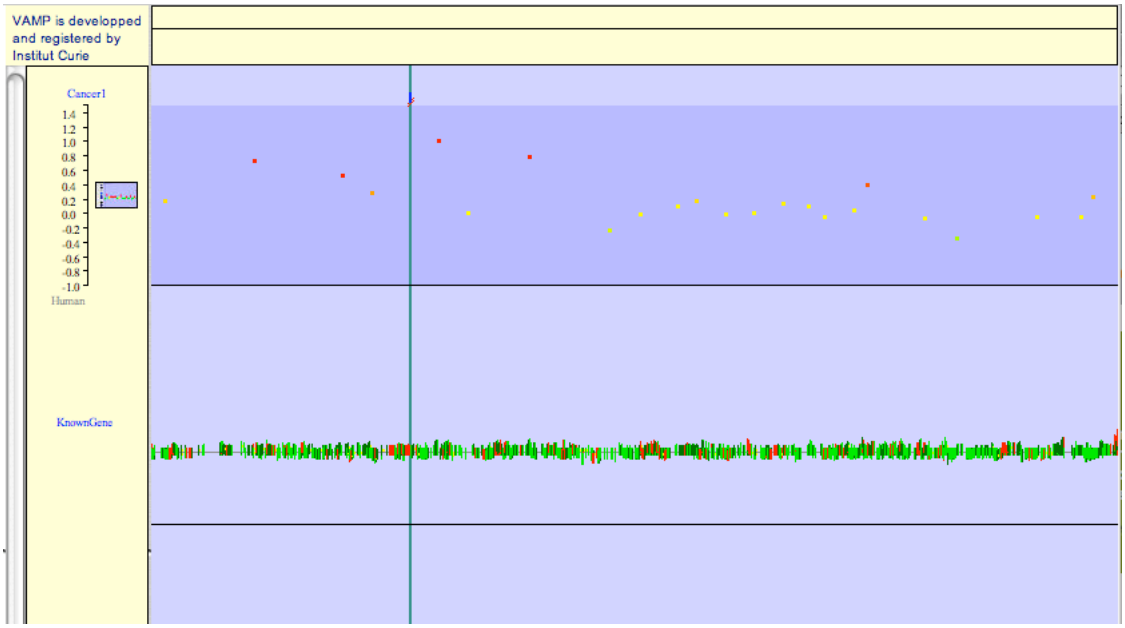

Figure 55: **Genome Annotation (zoom 3x)**

Right-clicking brings up a menu which allows to open web pages from NCBI gene, UCSC Genome Browser or Ensembl ContigView.

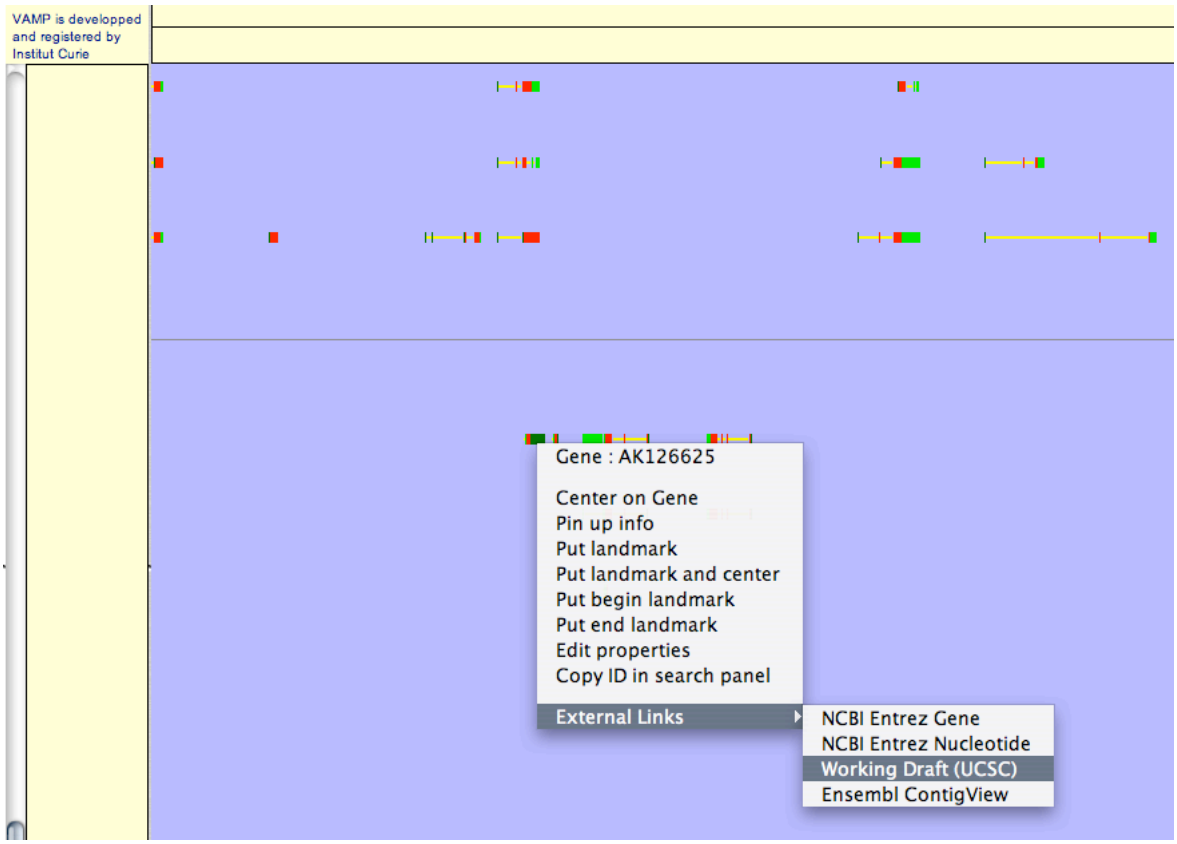

Figure 56: **Genome Annotation (zoom 30x)** (exons in red, introns in yellow, UTR 3' in light green and UTR 5' in dark green)

7 Availability

VAMP is available upon request to vamp@curie.fr. It can be tested on public data sets at the following url: http://bioinfo.curie.fr/vamp.

VAMP is also integrated into CAPweb, a complete bioinformatics pipeline for array-CGH data analysis. CAPweb can be tested at the following url: http://bioinfo.curie.fr/CAPweb

Note that three movies present as well the main functions of VAMP:

- http://bioinfo-out.curie.fr/vamp/doc/tutorial/vamp-demo1.html (Basic functions)
- http://bioinfo-out.curie.fr/vamp/doc/tutorial/vamp-demo2.html (Data Analysis)
- http://bioinfo-out.curie.fr/vamp/doc/tutorial/vamp-demo3.html (Synteny analysis)

8 How to cite us?

If you used VAMP please cite us as follows:
VAMP: Visualization and Analysis of CGH arrays, transcriptome and other Molecular Profiles. La Rosa et al, 2006, submitted to Bioinformatics

References

de Leeuw, R. J., Davies, J. J., Rosenwald, A., Bebb, G., Gascoyne, R. D., Dyer, M. J. S., Staudt, L. M., Martinez-Climent, J. A., and Lam, W. L. (2004). Comprehensive whole genome array CGH profiling of mantle cell lymphoma model genomes. *Hum Mol Genet*, 13:1827–1837.

Douglas, E. J., Fiegler, H., Rowan, A., Halford, S., Bicknell, D. C., Bodmer, W., Tomlinson, I. P. M., and Carter, N. P. (2004). Array comparative genomic hybridization analysis of colorectal cancer cell lines and primary carcinomas. *Cancer Res*, 64:4817–4825.

Gysin, S., Rickert, P., Kastury, K., and McMahon, M. (2005). Analysis of genomic DNA alterations and mRNA expression patterns in a panel of human pancreatic cancer cell lines. *Genes Chromosomes Cancer*, 44(1):37–51.

Hupé, P., Stransky, N., Thiery, J. P., Radvanyi, F., and Barillot, E. (2004). Analysis of array CGH data: from signal ratio to gain and loss of DNA regions. *Bioinformatics*, 20:3413–3422.

Kaufman, L. and Rousseuw, P. (1990). *Finding Groups in Data- An Introduction to Cluster Analysis*. Wiley Series in Probability and Mathematical Sciences.

Nakao, K., Mehta, K. R., Fridlyand, J., Moore, D. H., Jain, A. N., Lafuente, A., Wiencke, J. W., Terdiman, J. P., and Waldman, F. M. (2004). High-resolution analysis of DNA copy number alterations in colorectal cancer by array-based comparative genomic hybridization. *Carcinogenesis*, 25:1345–1357.

Pollack, J. R., Sorlie, T., Perou, C. M., Rees, C. A., Jeffrey, S. S., Lonning, P. E., Tibshirani, R., Botstein, D., Borresen-Dale, A.-L., and Brown, P. O. (2002). Microarray analysis reveals a major direct role of DNA copy number alteration in the transcriptional program of human breast tumors. *Proc Natl Acad Sci U S A*, 99:12963–12968.

Snijders, A. M., Nowak, N., Segraves, R., Blackwood, S., Brown, N., Conroy, J., Hamilton, G., Hindle, A. K., Huey, B., Kimura, K., S, S. L., Myambo, K., Palmer, J., Ylstra, B., Yue, J. P., Gray, J. W., Jain, A. N., Pinkel, D., and Albertson, D. G. (2001). Assembly of microarrays for genome-wide measurement of DNA copy number. *Nat. Genet.*, 29:263–4.

Snijders, A. M., Schmidt, B. L., Fridlyand, J., Dekker, N., Pinkel, D., Jordan, R. C. K., and Albertson, D. G. (2005). Rare amplicons implicate frequent deregulation of cell fate specification pathways in oral squamous cell carcinoma. *Oncogene*, 24:4232–4242.

Veltman, J. A., Fridlyand, J., Pejavar, S., Olshen, A. B., Korkola, J. E., DeVries, S., Carroll, P., Kuo, W.-L., Pinkel, D., Albertson, D., Cordon-Cardo, C., Jain, A. N., and Waldman, F. M. (2003). Array-based comparative genomic hybridization for genome-wide screening of DNA copy number in bladder tumors. *Cancer Res*, 63:2872–2880.