



Crop Information Portal Training

Release 1.0.x

GeoSolutions

February 16, 2016

1	Training Configurations	3
2	Overview	5
3	Using Crop Information Portal	9
4	Administration of Crop Information Portal	51
5	Maintenance	135
6	Glossary	149
	Python Module Index	151

Crop Information Portal is a web portal with reliable data and information on historical and current crop data and agro-metereological condition in Pakistan. It allows users to extract statistics and detect conditions affecting production of major Rabi and Kharif crops.

Training Configurations

In this section will be described how to setup, run and access to the Crop Informational Portal.

1.1 Setup

1. Download and Install VirtualBox

<https://www.virtualbox.org/wiki/Downloads>

2. Extract VM package

3. Add VM to Virtual Box

Machine -> Add..

4. Start VM

When it complains about a missing network card, select “configure VM” and choose your network card from the spinner.

As this is a self-contained environment network access is required only to display on-line backgrounds (Google, OpenStreetMap, MapQuest).

1. Log In

The login credentials are:

Username: trainee

Password: trainee

1.2 Access to the application

Access to the application using the Browser inside the VM at the following URL:

1. <http://localhost/MapStore/>

Or from the Host machine browser at the following URL:

1. <http://localhost:8888/MapStore/>

Crop Information Portal makes available to WEB users the historical archive and the latest produced crop and agro-meteorological data aggregated at district/province scale and integrated with satellite based information into a GIS like environment.

It provides also a SITS(*Satellite Image Time Series*) of the the NDVI(*Normalized Difference Vegetation Index*) at 10 days interval (*dekad*).

5

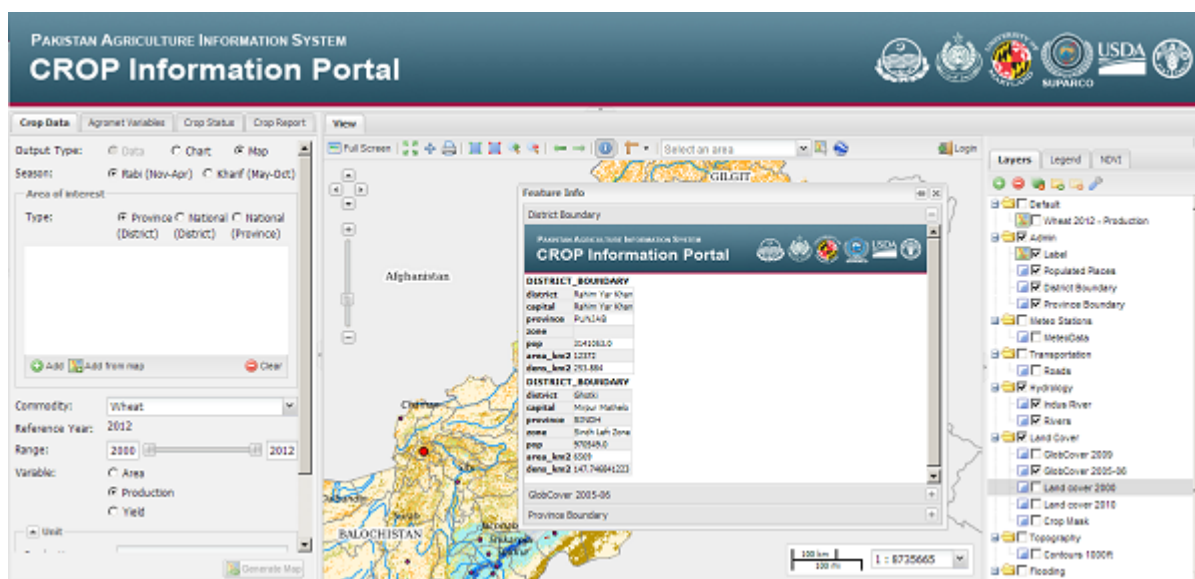


Fig. 2.2: Usage of the GIS like environment.

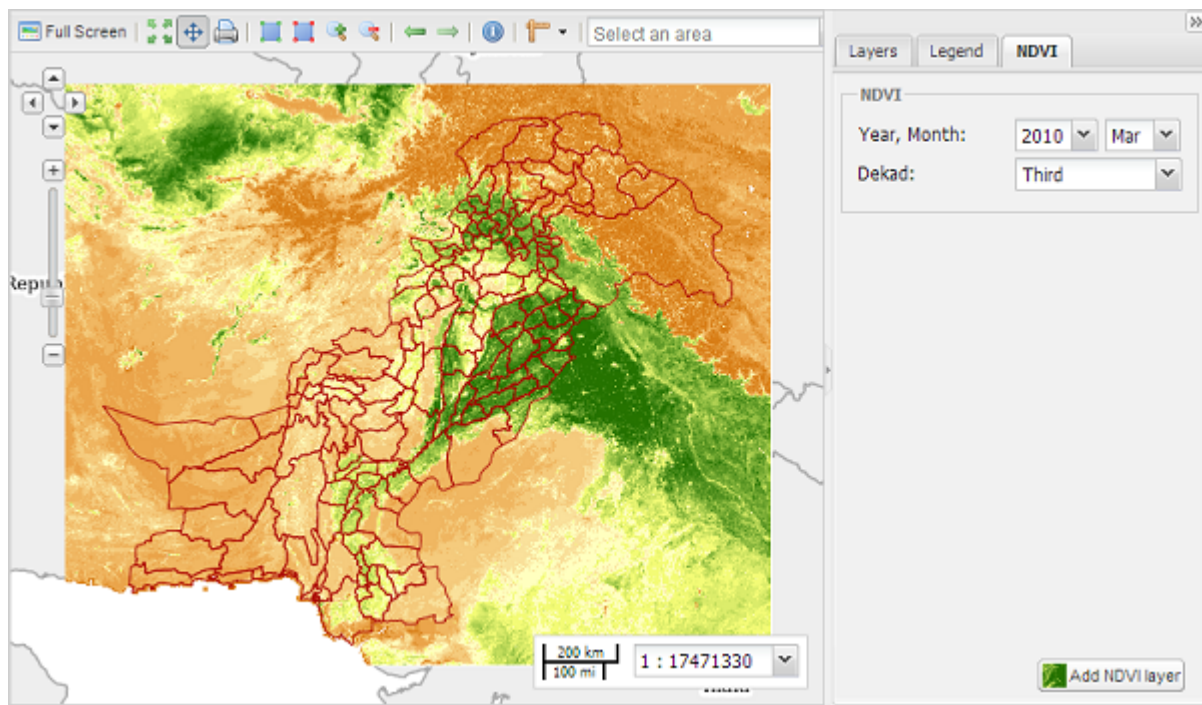


Fig. 2.4: The NDVI image from the SITS.

Portal View Geonetwork Administration Page

Workflows User Manager File Manager Entities

USER LOGGED: admin Logout

Workflows File Manager User Manager Entities

Flows

Active Archived

Refresh Auto Refresh

Title	Description	Status	StartDate	File	Task	Progress		
Create or update Layer	Updates or creates a layer from a given shap...	SUCCESS	2015-12-17 16:27:28.572	20151217144813-suparco_vihe...	Completed	100%	-	i
CSV file ingestion	Ingest CSV files (cropdata and agromet)	SUCCESS	2015-12-17 16:26:47.324	20151217144732-suparco_suga...	Completed	100%	-	i
NDVI file ingestion	Ingest geotiff file and add them to a NDVI mos...	SUCCESS	2015-12-17 16:26:16.390	20151217144701-suparco_rice...	Completed	100%	-	i
NDVI stats generation	Generate a CSV file with a geotiff mask and a ...	SUCCESS	2015-12-17 16:25:42.886	20151217144627-suparco_cotto...	Completed	100%	-	i
		SUCCESS	2015-11-05 12:50:22.025	20151105111219-sindh_wheat...	Completed	100%	-	i
		SUCCESS	2015-11-05 12:07:00.875	20151105102858-kp_wheat_20...	Completed	100%	-	i
		SUCCESS	2015-10-09 11:12:24.877	20151009103502-ADD_104_iri...	Completed	100%	-	i
		SUCCESS	2015-10-09 11:10:48.050	20151009103325-ADD_104_iri...	Completed	100%	-	i

Credits

Fig. 2.5: The administration interface.

Using Crop Information Portal

The Crop Information Portal provides almost all of its functionalities using the framework called MapStore.

In this section you can learn how to use all the available tools.

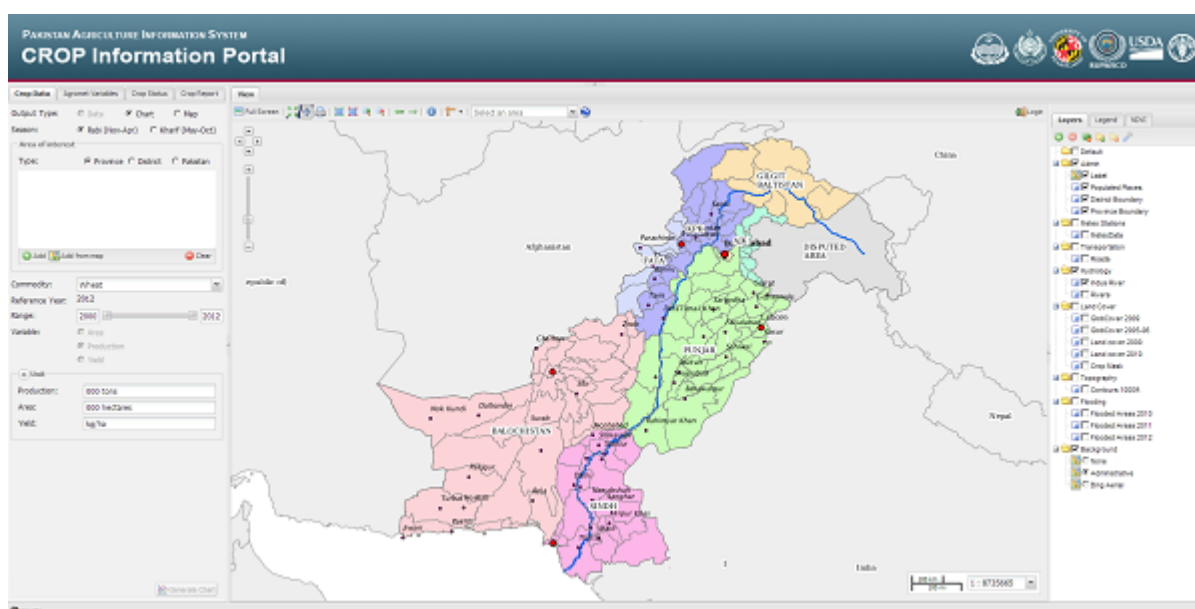


Fig. 3.1: MapStore version for the Crop Information Portal.

3.1 Generic Tools

This section illustrates the main tools to interact with the layers shown on the map.

3.1.1 Main Toolbar

	Enables the full screen display
	Allows you to zoom to the full extent of your map
	Allows you to pan the map
	Allows you to zoom in/zoom out to/from a geographic window by dragging a box
	Allows you to zoom in/zoom out on the center of your map
	Allows you to go back/forward to the previous/next extent
	Identifies the geographic feature on which you click
Length Area Bearing	Measures distance/area/bearing on the map
	Allows you to zoom in an area of interest
	Allow you to submit login/logout

3.2 Layer Switcher

Layer Switcher lists all the layers on the map and shows what the features in each layer represent. The layers at the top of the table of contents draw on top of those below them. In CROP Information Portal's Layer Switcher there are two types of layers: base layers and overlays.

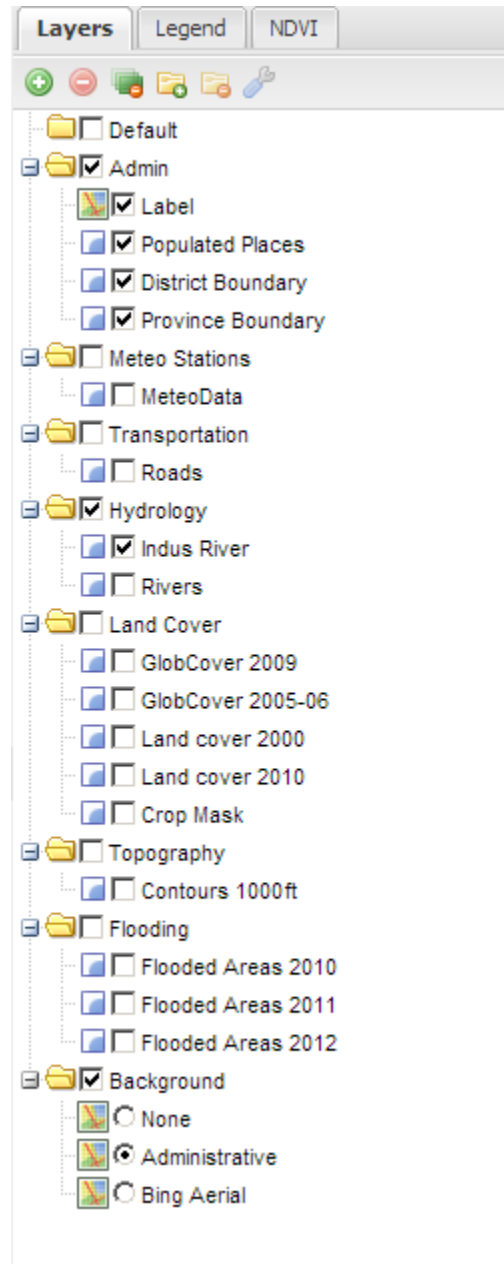


Fig. 3.2: the Layer Switcher.

3.2.1 Base Layers

Base Layers are mutually exclusive layers, meaning only one can be enabled at any given time. Base Layers always display below overlay layers.

In CROP Information Portal this layers are grouped in a GROUP called:

Background

Name	Description
None	no background layer
Administrative	only administrative boundaries
Google Roadmap	only Google Roadmap
Google Hybrid	only Google Hybrid
Google Terrain	only Google Terrain
Open Street Map	only Open Street Map
MapQuest OpenStreetMap	only MapQuest OpenStreetMap

3.2.2 Non Base Layers (Overlays)

Non base layers, sometimes called overlays, are the alternative to Base Layers. Multiple non-base layer can be enabled at a time.

In CROP Information Portal this layers are grouped in several GROUPS

Admin group

Name	Description
Label	Administrative boundaries label
Populated Places	Populated Places boundaries
District Boundary	Pakistan District Boundaries
Province Boundary	Pakistan Province Boundaries

Meteo Stations group

Name	Description
MeteoData	Meteorological Stations of Pakistan

Transportation group

Name	Description
Roads	Main Streets of the Pakistan

Hydrology group

Name	Description
Indus River	Indus River
Rivers	Rivers

Land Cover group

Name	Description
GlobalCover 2009	Land cover of 2009
GlobalCover 2005-06	Land cover of 2005-06
Land cover 2000	Land cover of 2000
Land cover 2010	Land cover of 2010
Crop Mask	Mask of Crop

Topography group

Name	Description
Contours 1000ft	Contours with equidistance of 1000 feet

Flooding group

Name	Description
Flooded Areas 2010	Flooded areas related to 2010
Flooded Areas 2011	Flooded areas related to 2011
Flooded Areas 2012	Flooded areas related to 2012

3.3 Legend Panel

This section illustrates **Legend Panel**.

3.3.1 Tab to view Legend Panel

Legend Panel is located in the east of the layout and shows you legend associated with layers. It only shows the legend of the active layers.

3.4 NDVI Tool

This section illustrates how you can use **NDVI Tool**.

3.4.1 Tab to interact with NDVI tool

NDVI tool is located in the east of the layout and allows you to add an image of **Normalized Difference Vegetation Index (NDVI)** to the map.

The images are produced for dekad and the choice can be made by means of three drop-down menu.

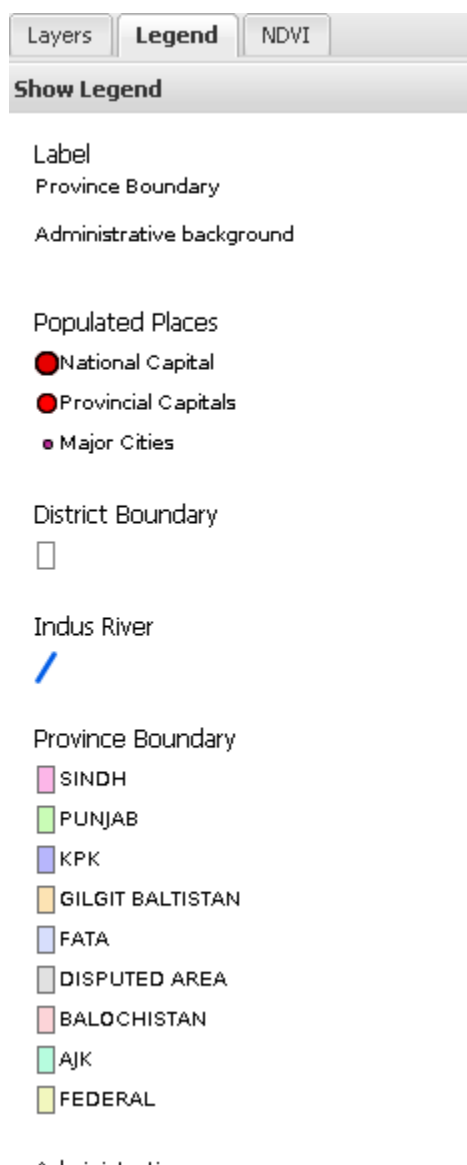


Fig. 3.3: Legend Panel.

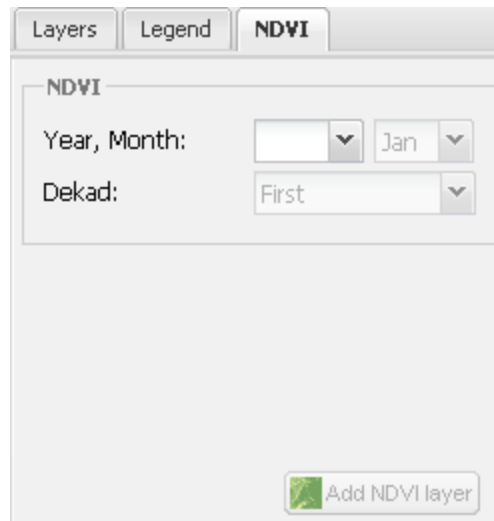


Fig. 3.4: NDVI tool.

3.4.2 Select Dekad

The first drop-down allows you to choose the year, the second allows you to choose the month and the third allows you to choose the decade

Select Year	Select Month	Select Dekad
1998		
1999		
2000		
2001	Jan	
2002	Feb	
2003	Mar	
2004	Apr	
2005	May	
2006	Jun	
2007	Jul	
2008	Aug	
2009	Sep	First
2010	Oct	First
2011	Nov	Second
2012	Dec	Third

3.4.3 Add NDVI layer to the map

Once you have selected a dekad you can add the image to the map by clicking on the button “Add NDVI layer”. Now you can also query the layer.

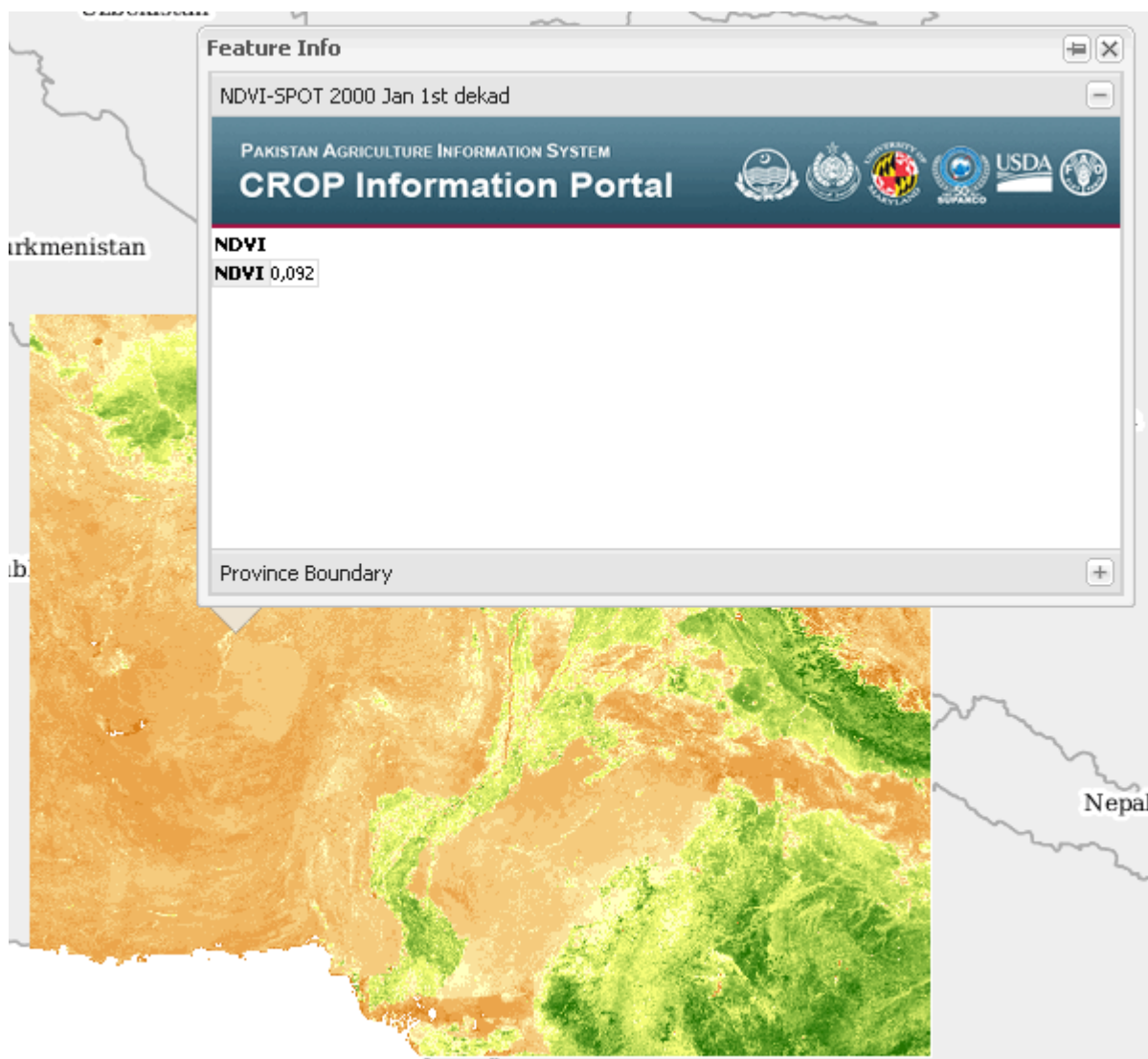


Fig. 3.5: NDVI layer and GetFeatureInfo example.

3.5 Crop Data Tool

This section illustrates how you can use **Crop Data Tool**.

3.5.1 Tab to interact with Crop Data Tool

Crop Data Tool is located in the west of the layout and allows you to generate charts and maps dynamically aggregating yearly data about production, cultivated areas and yield of each crop.

Generate a Chart

To generate charts that describes the trend of the wheat production choose “Chart” as output type and select the Season, the Area of interest and the range of years in which you are interested.

Choose Output Type Chart:

Choose Season:

Choose Area Of Interest:

The screenshot displays the 'Crop Data Tool' interface. At the top, there are tabs for 'Crop Data', 'Agromet Variables', 'Fertilizers', and 'Water Resources'. Below these are sub-tabs for 'Market Prices', 'Crop Status', and 'Crop Report'. The 'Crop Data' sub-tab is active.

Configuration options include:

- Output Type:** Radio buttons for 'Data', 'Chart' (selected), and 'Map'.
- Season:** Radio buttons for 'Rabi (Nov-Apr)' (selected) and 'Kharif (May-Oct)'.
- Area of interest:** A section with a 'Type:' label and radio buttons for 'Province' (selected), 'District', and 'Pakistan'. Below this is a large empty rectangular box. At the bottom of this section are three buttons: '+ Add', '+ Add from map', and '- Clear'.
- Mode:** Radio buttons for 'Composite' (selected), 'Comparison by Region', 'Comparison by Commodity', and 'Comparison by Source'.
- Commodity:** A dropdown menu currently showing 'Wheat'.
- Chart Type:** Radio buttons for 'Values' (selected), 'Anomalies', and 'Anomalies (%)'. Below these is a 'Reference Year' label with the value '2012'.
- Range:** A slider control with '2000' on the left and '2012' on the right.
- Unit:** A section with a 'Unit' label and three dropdown menus: 'Production' (showing '000 tons'), 'Area' (showing '000 hectares'), and 'Yield' (showing 'kg/ha').

At the bottom of the interface, there is a 'Help' button, a 'Generate Chart' button with a chart icon, and a small dropdown arrow next to it.

Fig. 3.6: Crop Data Tool.

Output Type: ☐ Data ☒ Chart ☐ Map



Fig. 3.7: Choose Output Type Chart.

Season: ☒ Rabi (Nov-Apr) ☐ Kharif (May-Oct)

Fig. 3.8: Choose Season.

Area of interest

Type: ☒ Province ☐ District ☐ Pakistan

BALUCHISTAN	
SINDH	




 Add  Add from map  Clear

Fig. 3.9: Choose Area Of Interest.

Area Of Interest selection:

Choose “Composite” Mode:

Choose Commodity:

Choose Chart Type:

Choose Years Range:

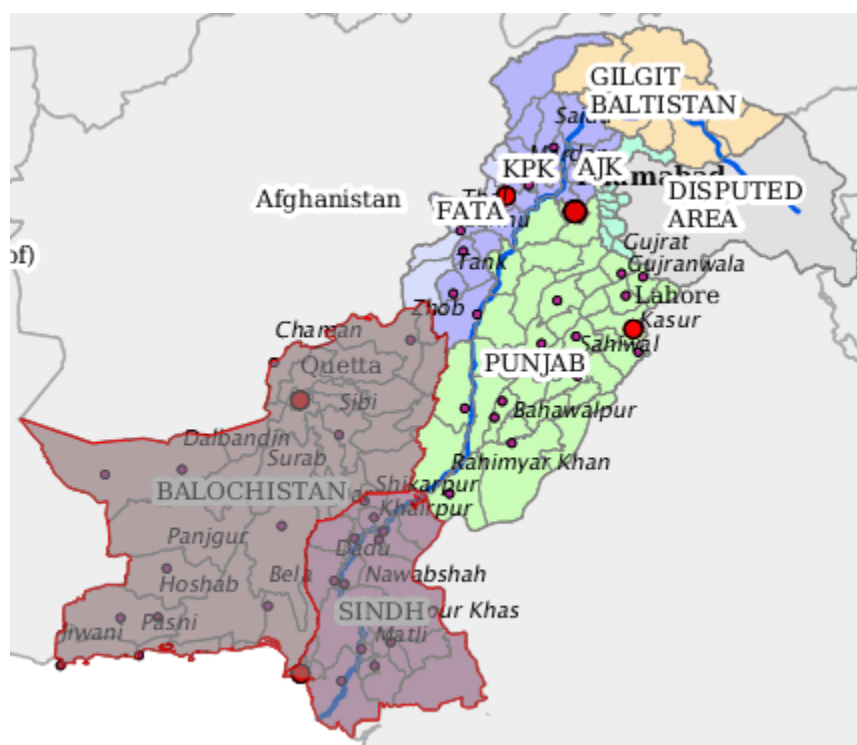


Fig. 3.10: Area Of Interest selection.

Mode:

- ☒ Composite
- ☐ Comparison by Region
- ☐ Comparison by Commodity
- ☐ Comparison by Source

Fig. 3.11: Choose Mode.

Commodity:

Fig. 3.12: Choose Commodity.

Chart Type:

- ☒ Values
- ☐ Anomalies
- ☐ Anomalies (%)

Fig. 3.13: Choose Chart Type.

Reference Year: 2012

Range:

Fig. 3.14: Choose Years Range.

Generate Chart Button:

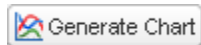


Fig. 3.15: Generate Chart Button.

Generated Charts:

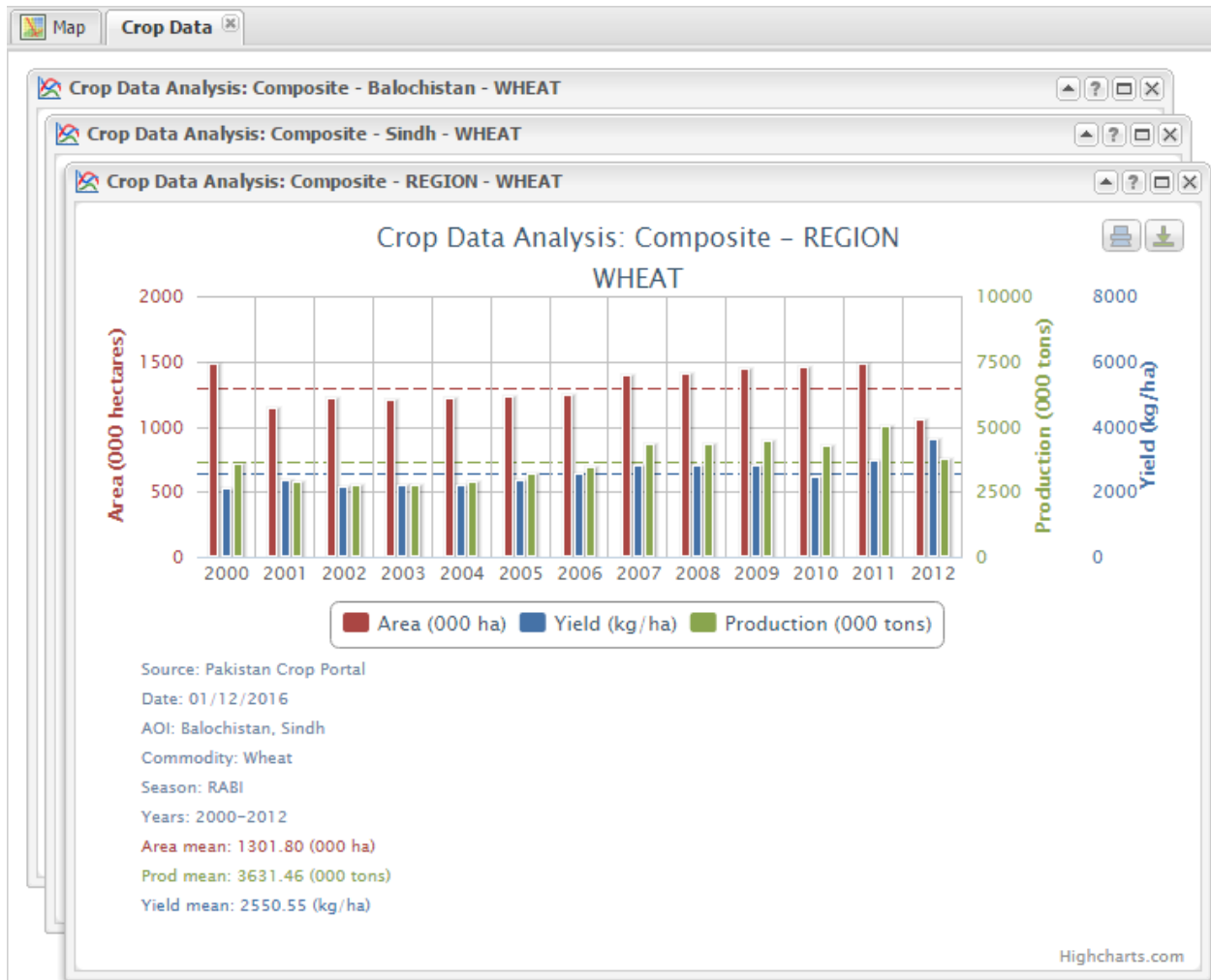


Fig. 3.16: Generated Charts.

Charts Legend:



Fig. 3.17: Charts Legend.

Generate a Map

To generate a map that describes the trend of the rice production choose “Map” as output type and select the Season, the Area of interest and the range of years in which you are interested.

Choose Output Type Map:

Output Type: ☐ Data ☐ Chart ☒ Map

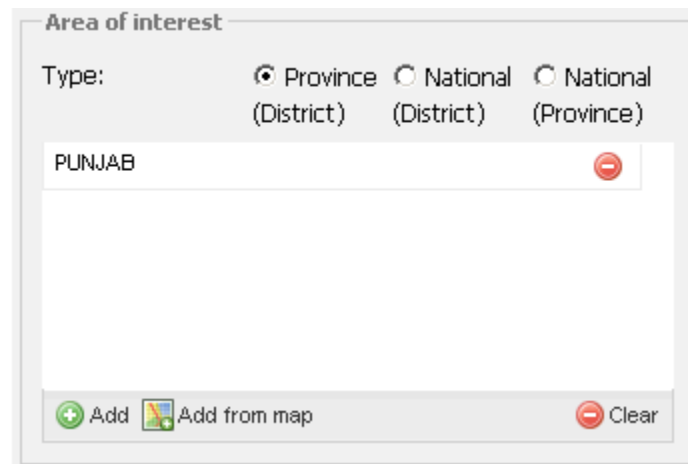
Fig. 3.18: Choose Output Type Map.

Choose Season:

Season: ☐ Rabi (Nov-Apr) ☒ Kharif (May-Oct)

Fig. 3.19: Choose Season.

Choose Area Of Interest:



Area of interest

Type: ☒ Province (District) ☐ National (District) ☐ National (Province)

PUNJAB ⊖

⊕ Add 🗺️ Add from map ⊖ Clear

Fig. 3.20: Choose Area Of Interest.

Area Of Interest selection:

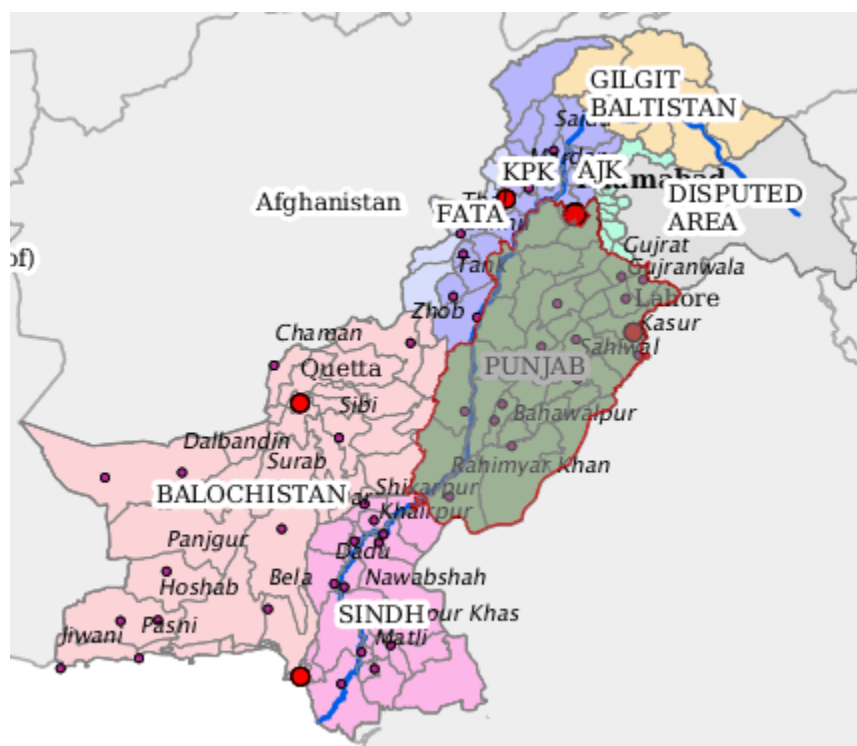


Fig. 3.21: Area Of Interest selection.

Choose Commodity:

Commodity:

Fig. 3.22: Choose Commodity.

Choose Years Range:

Reference Year: 2012
Range:

Fig. 3.23: Choose Years Range.

Choose Variable:

Variable:

<input type="radio"/> Area	<input checked="" type="radio"/> Reference Year
<input checked="" type="radio"/> Production	<input type="radio"/> Difference
<input type="radio"/> Yield	<input type="radio"/> Average

Fig. 3.24: Choose Variable.

Generate Map Button:



Fig. 3.25: Generate Map Button.

Generated Map Layer:



Fig. 3.26: Generated Map Layer.

Generated Map:

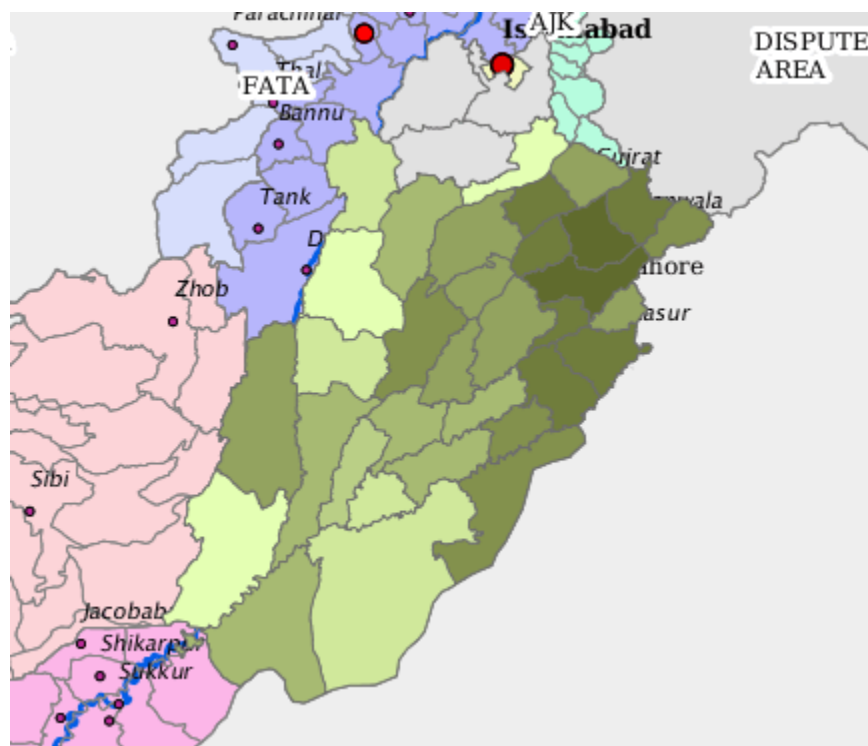


Fig. 3.27: Generated Map.

Generated Map Legend:

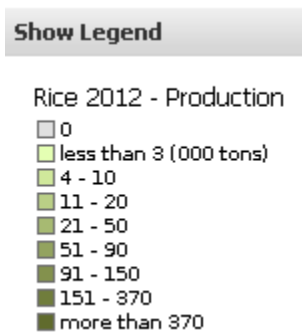


Fig. 3.28: Generated Map Legend.

Smart Identification tool

This tool allows you to query the map generated by Crop Data Tool.
You must first select the generated layer from the Layer Switcher panel.



Fig. 3.29: Smart Map Identification Tool.

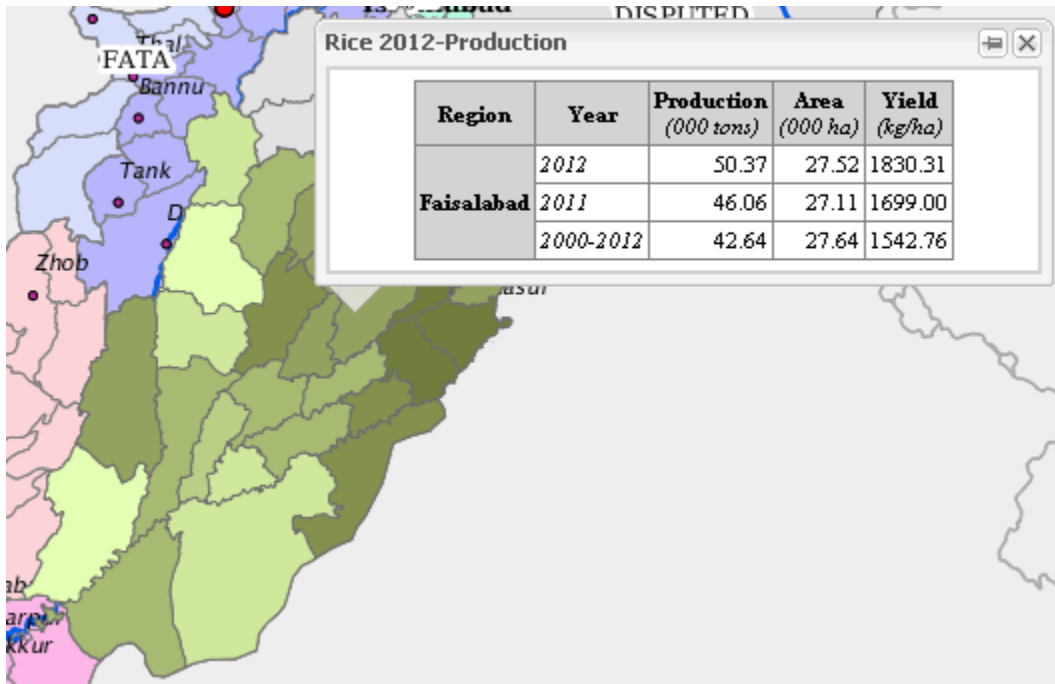


Fig. 3.30: Smart Map Identification.

Generate Tabular data

This option allows you to generate tabular data and download it as Comma Separated Value (CSV). This option is enabled only for logged in users and it is also available for **Crop Status Tool** and **Agromet Tool**

Click on the login button.



Fig. 3.31: Login button.

Enter your credentials.

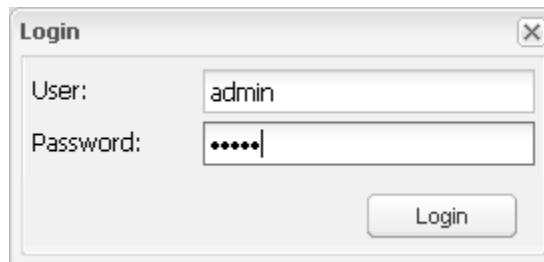
A login form window titled "Login" with a close button (X) in the top right corner. It contains two input fields: "User:" with the text "admin" and "Password:" with masked characters ".....". A "Login" button is located at the bottom right of the form.

Fig. 3.32: Login Form.

If the credentials are valid the option Data is enabled in the Output Type radio button.

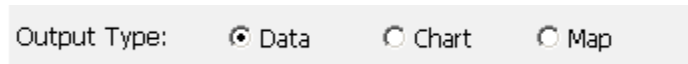
A horizontal row of three radio buttons labeled "Data", "Chart", and "Map". The "Data" radio button is selected, indicated by a filled circle.

Fig. 3.33: Output type Data.

To generate tabular data and download it as Comma Separated Value (CSV) of the rice production choose “Data” as output type and select the Season, the Area of interest and the range of years in which you are interested.

Select the season: Rabi or Kharif:

Season: ☐ Rabi (Nov-Apr) ☒ Kharif (May-Oct)

Fig. 3.34: Choose Season.

Select the Area Of Interest:

Area of interest

Type: ☒ Province ☐ District ☐ Pakistan

PUNJAB ✖

+ Add 🗺️ Add from map ✖ Clear

Fig. 3.35: Choose Area Of Interest.

the selected zones are highlighted on the map:

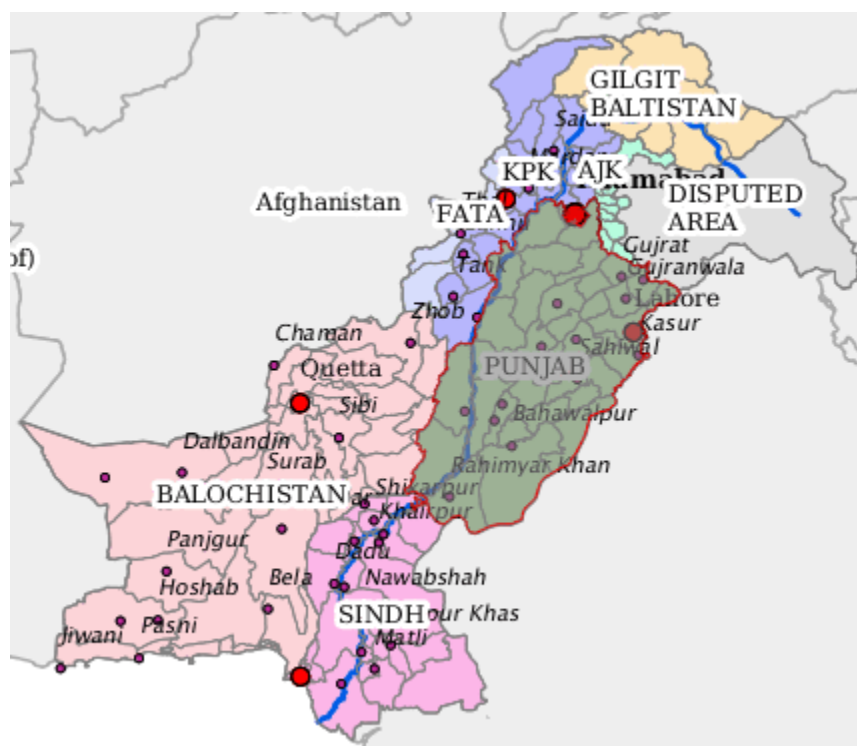


Fig. 3.36: Area Of Interest selection.

Note that the red disk icon allows to remove item(s) from the list.

Also re-clicking on a shape makes the same shape to be deselected.

Additional note, if you want a specific area you do not know where it is but you know its name, you can use the Add (+) button: an input box for searching the DB pops up.

Select at least one data source

Sources	
<input type="checkbox"/>	Official
<input type="checkbox"/>	SUPARCO

Fig. 3.37: Sources list.

Select a commodity:

Select the time interval by using the dedicated slider. It defines the range for averages and the reference year (highest). The default range is the max available.

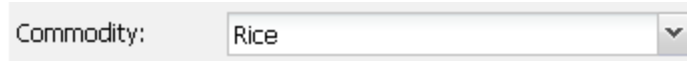


Fig. 3.38: Choose Commodity.

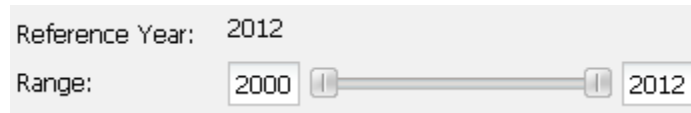


Fig. 3.39: Choose Years Range.

Select the factor to analyze.

Variable:

<input type="radio"/> Area	<input checked="" type="radio"/> Reference Year
<input checked="" type="radio"/> Production	<input type="radio"/> Difference
<input type="radio"/> Yield	<input type="radio"/> Average

Fig. 3.40: Choose Variable.

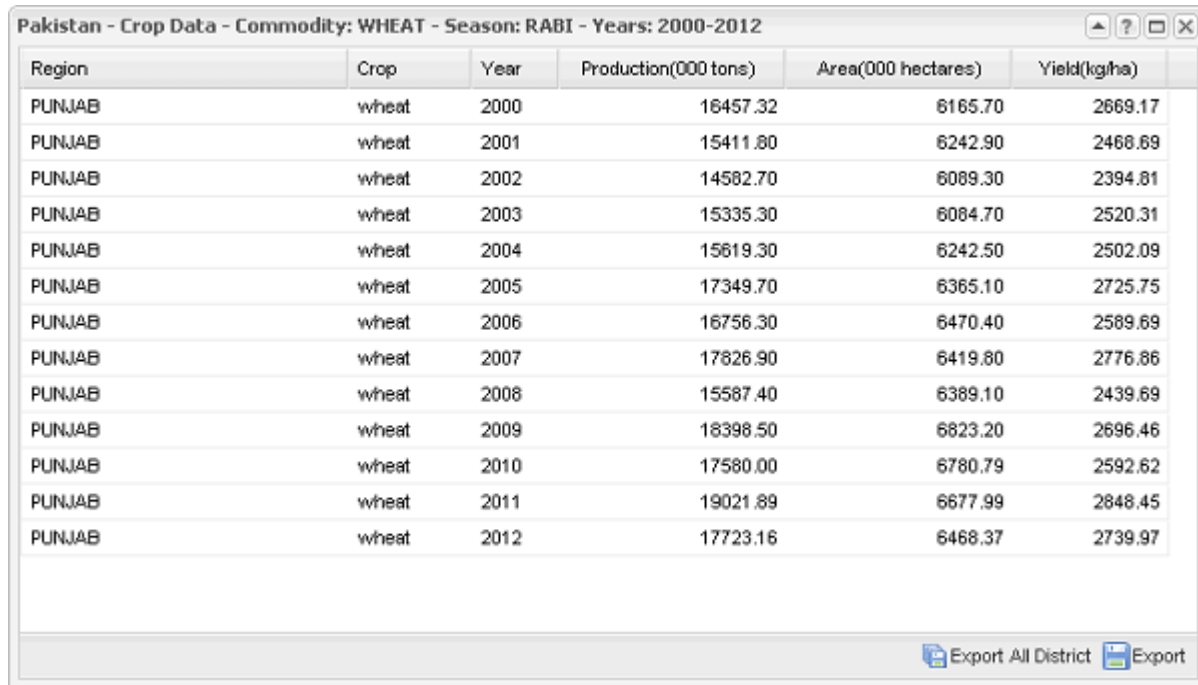
Click on the Generate Table button



Fig. 3.41: Generate Table Button.

The result is a formatted table ("Crop Data" near the View tab) of source values.

After clicking on the button will open a grid where the information will be listed in tabular form.



Region	Crop	Year	Production(000 tons)	Area(000 hectares)	Yield(kg/ha)
PUNJAB	wheat	2000	16457.32	6165.70	2669.17
PUNJAB	wheat	2001	15411.80	6242.90	2468.69
PUNJAB	wheat	2002	14582.70	6089.30	2394.81
PUNJAB	wheat	2003	15335.30	6084.70	2520.31
PUNJAB	wheat	2004	15619.30	6242.50	2502.09
PUNJAB	wheat	2005	17349.70	6365.10	2725.75
PUNJAB	wheat	2006	16756.30	6470.40	2589.69
PUNJAB	wheat	2007	17826.90	6419.80	2776.86
PUNJAB	wheat	2008	15587.40	6389.10	2439.69
PUNJAB	wheat	2009	18398.50	6823.20	2696.46
PUNJAB	wheat	2010	17580.00	6780.79	2592.62
PUNJAB	wheat	2011	19021.89	6677.99	2848.45
PUNJAB	wheat	2012	17723.16	6468.37	2739.97

Fig. 3.42: Tabular Data Grid.

Factor values (e.g. NDVI) are listed by dekad and month.

Three columns provide data for reference year, previous and selected time interval (average).

All factors are included in the same table.

Now you can export the data in CSV format and save them on your PC.

If you have selected one or more provinces, you can export the data for only one province or districts belonging to the province.

If you have selected a district can export only the data for the district.

The icon at the bottom right corner allows the data to be exported as CSV file.

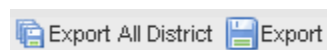


Fig. 3.43: Export Data Buttons.

3.6 Crop Status Tool

This section illustrates how you can use **Crop Status Tool**.

3.6.1 Tab to interact with Crop Status Tool

Crop Status Tool is located in the west of the layout and allows you to generate chart dynamically.

The screenshot shows the 'Crop Status' tab selected in a multi-tabbed interface. The tabs are 'Crop Data', 'Agromet Variables', 'Crop Status', and 'Crop Report'. The 'Crop Status' tab contains the following elements:

- Output Type:** Two radio buttons, 'Data' (unselected) and 'Chart' (selected).
- Season:** Two radio buttons, 'Rabi (Nov-Apr)' (selected) and 'Kharif (May-Oct)' (unselected).
- Area of interest:** A section with a 'Type:' label and two radio buttons, 'Province' (selected) and 'District' (unselected). Below this is a 'Province:' label followed by a search input field with the placeholder text 'Search' and a small map icon to its right.
- Year:** A dropdown menu currently showing '2014'.
- Commodity:** A dropdown menu currently showing 'Wheat'.
- Factors:** A table with five rows, each featuring a checkbox, a factor name, and a unit.

Factors		
<input type="checkbox"/>	Max Temperature	°C
<input type="checkbox"/>	NDVI	
<input type="checkbox"/>	Precipitation	mm
<input type="checkbox"/>	Day length	hr
<input type="checkbox"/>	Min Temperature	°C

Fig. 3.44: Crop Status Tool.

Output Type Chart

This close-up shows the 'Output Type:' label and the two radio buttons: 'Data' (unselected) and 'Chart' (selected).

Fig. 3.45: Choose Output Type Chart.

This close-up shows the 'Season:' label and the two radio buttons: 'Rabi (Nov-Apr)' (selected) and 'Kharif (May-Oct)' (unselected).

Fig. 3.46: Choose Season.

Area of interest

Type: ☒ Province ☐ District


Province: 

Fig. 3.47: Choose Area Of Interest.

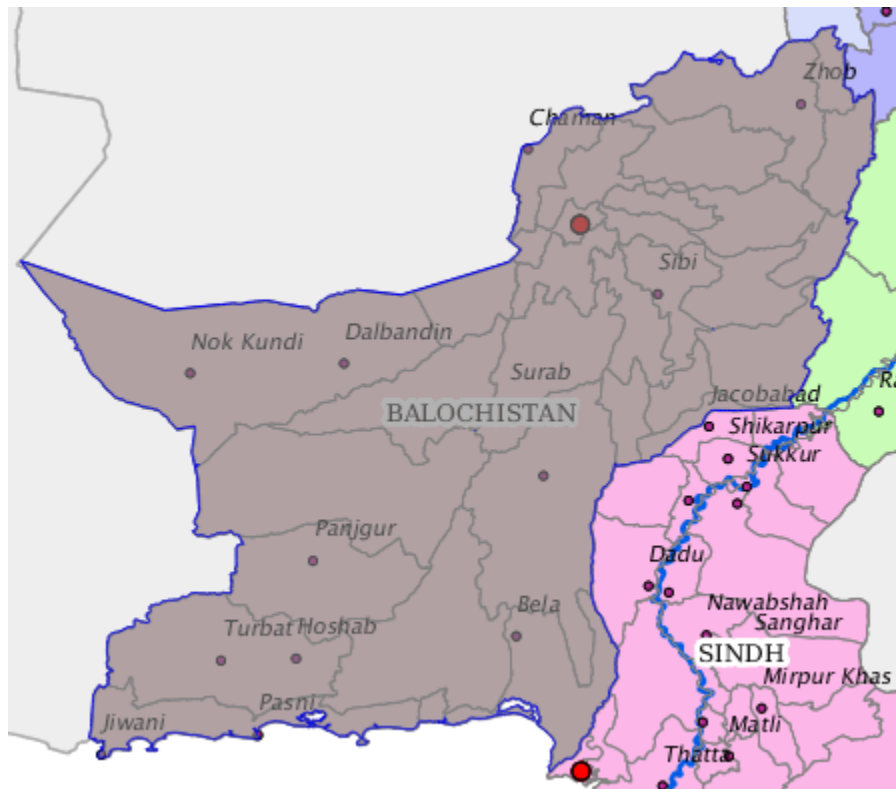


Fig. 3.48: Area Of Interest selection.


Year: 

Fig. 3.49: Choose Year.


Commodity: 

Fig. 3.50: Choose Commodity.

Factors		
<input checked="" type="checkbox"/>	Max Temperature	°C
<input type="checkbox"/>	NDVI	
<input type="checkbox"/>	Precipitation	mm
<input type="checkbox"/>	Day length	hr
<input type="checkbox"/>	Min Temperature	°C

Fig. 3.51: Choose Factor.

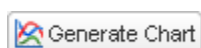


Fig. 3.52: Generate Chart Button.

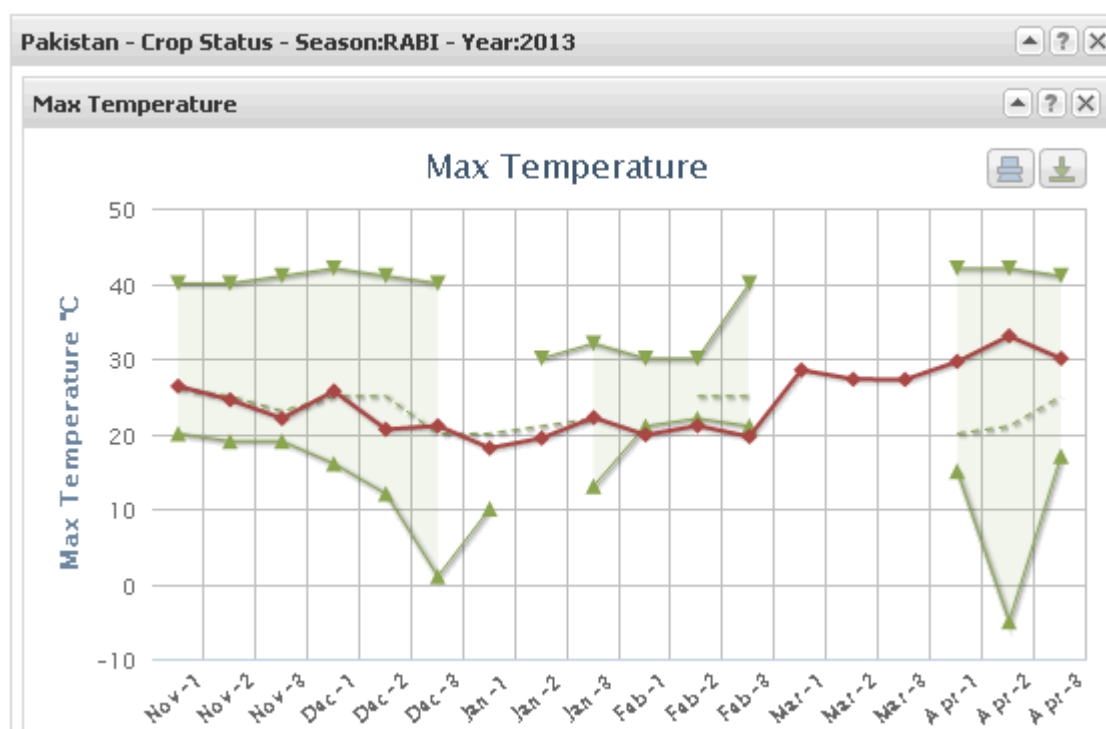


Fig. 3.53: Generated Chart.

Source: Pakistan Crop Portal
 Date: 01/12/2014
 AOI: 'balochistan'
 Commodity: wheat
 Season: RABI
 Year: 2013

--- Optimal ▼ Max ▲ Min ◆ Max Temperature

Fig. 3.54: Generated Chart Legend.

3.7 Agromet Tool

This section illustrates how you can use **Agromet Tool**.

3.7.1 Tab to interact with Agromet Tool

Agromet Tool is located in the west of the layout and allows you to generate chart dynamically.

The agromet tool aggregates agro-meteorological data from different areas of interest and display these data in dekadal charts.

The charts contains 3 charts:

- one chart for the reference year values
- one chart for the previous year values
- one chart for the average of the values of the previous years on the selected interval.

The screenshot shows the 'Agromet Variables' tab of a web application. The interface includes several sections for configuring data output:

- Navigation:** Tabs for 'Crop Data', 'Agromet Variables' (active), 'Crop Status', and 'Crop'.
- Output Type:** Radio buttons for 'Data' and 'Chart' (selected).
- Season:** Radio buttons for 'Rabi (Nov-Apr)' (selected) and 'Kharif (May-Oct)'.
- Area of interest:**
 - Type: Radio buttons for 'Province' (selected), 'District', and 'Pakistan'.
 - A large empty rectangular box for visualization.
 - Buttons: '+ Add', 'Add from map' (with a map icon), and '- Clear'.
- Reference Year:** A text input field containing '2012'.
- Range:** A slider control with input fields for '2000' and '2014'.
- Factors:** A table with checkboxes for selecting data factors.

Factors	
<input type="checkbox"/>	Max Temperature °C
<input type="checkbox"/>	NDVI
<input type="checkbox"/>	Precipitation mm
<input type="checkbox"/>	Day length hr
<input type="checkbox"/>	Min Temperature °C

Fig. 3.55: Agromet Tool.

Output Type Chart

To generate charts that describes the trend of factor **Max Temperature** choose **Chart** as output type and select the **Season**, the **Area of interest** and the **Range of years** in which you are interested.

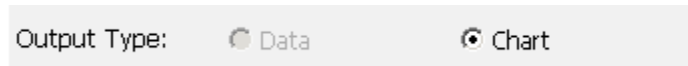


Fig. 3.56: Choose Output Type Chart.

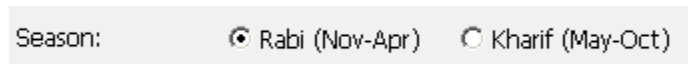


Fig. 3.57: Choose Season.

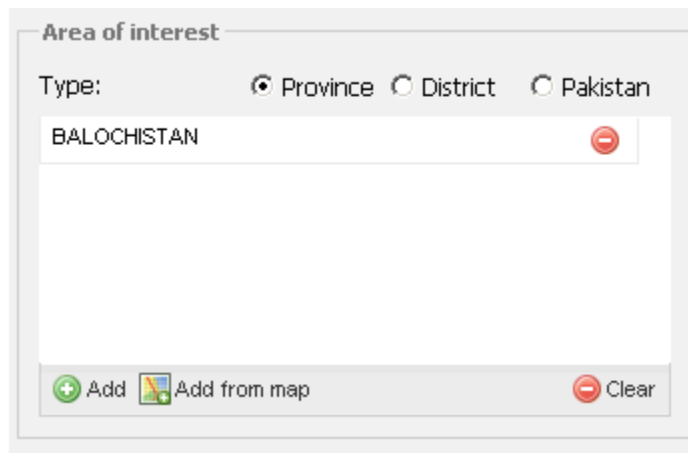


Fig. 3.58: Choose Area Of Interest.

3.8 Reporting Tool

This module is responsible of the PDF report generation that includes data charts and specific maps with the aggregated data.

To generate a report you need:

- Season
- **Area of interest (AoI)**
 - Type
 - Zones
- Commodity
- Range
- Meteorological factors
- Layout configuration

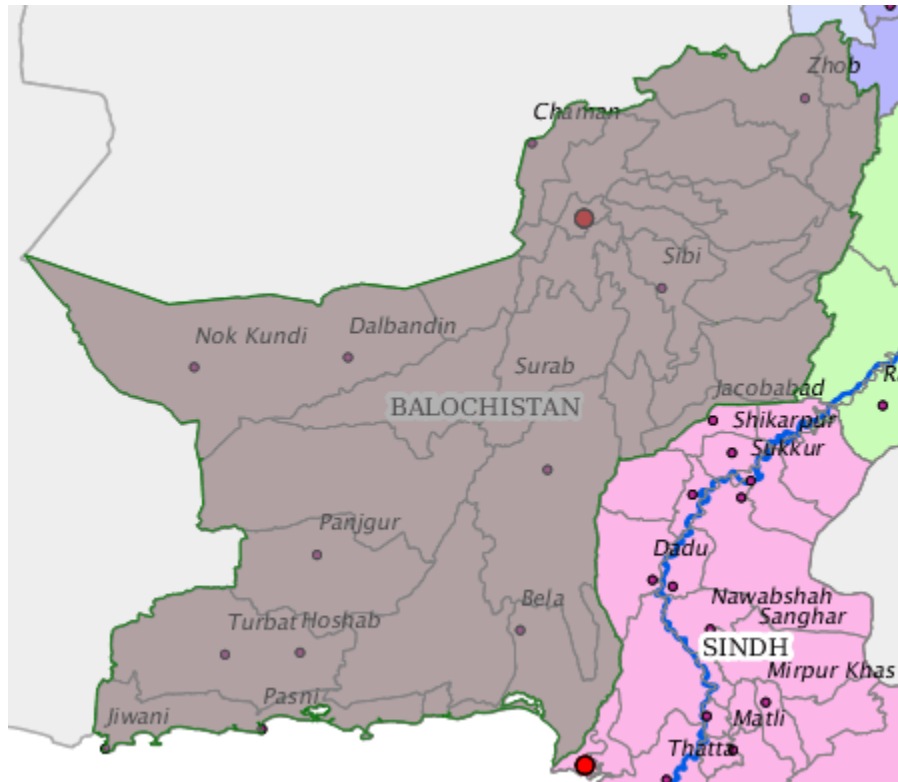


Fig. 3.59: Area Of Interest selection.

Reference Year: 2012

Range: 2000 2012

Fig. 3.60: Choose Years Range.

Factors		
<input checked="" type="checkbox"/>	Max Temperature	°C
<input type="checkbox"/>	NDVI	
<input type="checkbox"/>	Precipitation	mm
<input type="checkbox"/>	Day length	hr
<input type="checkbox"/>	Min Temperature	°C

Fig. 3.61: Choose Factor.

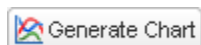


Fig. 3.62: Generate Chart Button.

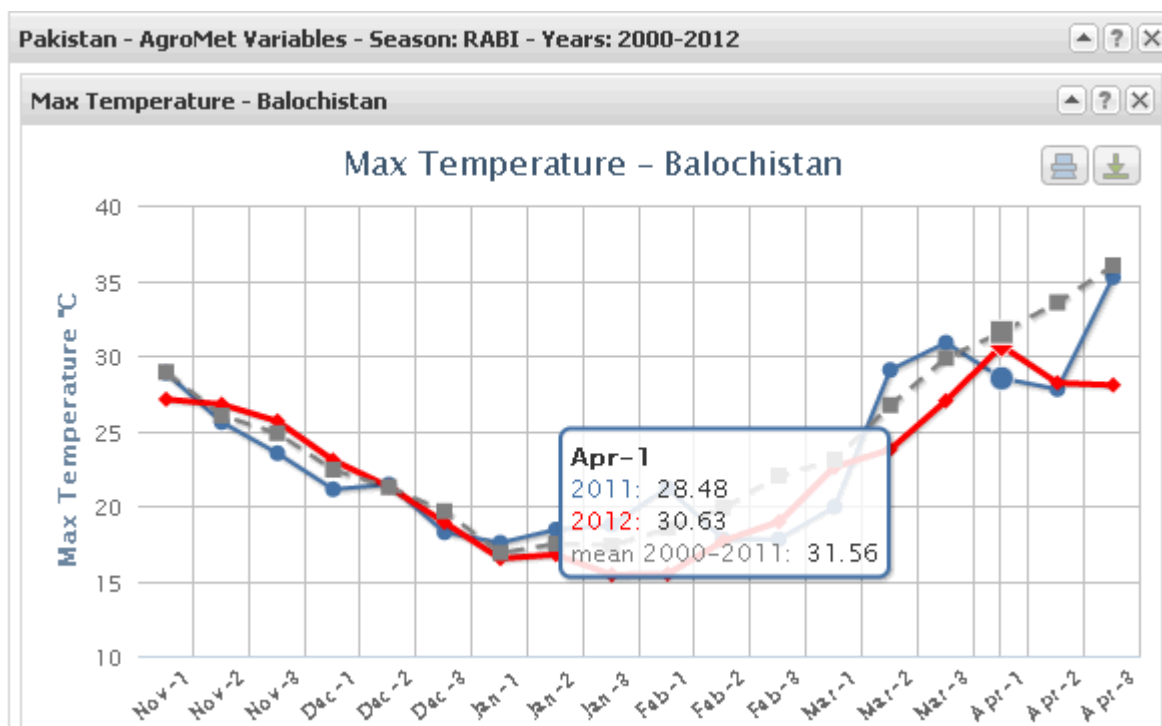


Fig. 3.63: Generated Chart.

Source: Pakistan Crop Portal
 Date: 01/12/2014
 AOI: Balochistan
 Season: RABI
 Years: 2000-2012



Fig. 3.64: Generated Chart Legend.

To access to this module, please press on Crop Report button on the user interface:



Fig. 3.65: Button for the Crop Report module.

3.8.1 Season

The image shows a 'Season:' label followed by two radio buttons. The first radio button is selected and is labeled 'Rabi (Nov-Apr)'. The second radio button is unselected and is labeled 'Kharif (May-Oct)'.

Fig. 3.66: Season selection.

3.8.2 Area of interest

You can select province, district or Pakistan.

For province and district selection you need to select the selected zones with the Add button

The image shows a form titled 'Area of interest'. It has a 'Type:' label followed by three radio buttons: 'Province' (selected), 'District', and 'Pakistan'. Below the radio buttons is a large empty rectangular area. At the bottom of the form, there are three buttons: a green '+ Add' button (highlighted with a red rectangle), a map icon followed by 'Add from map', and a red '- Clear' button.

Fig. 3.67: Add button.

and search the zone to add:

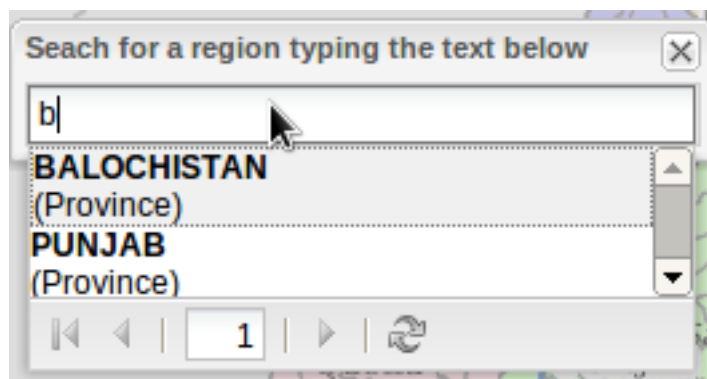
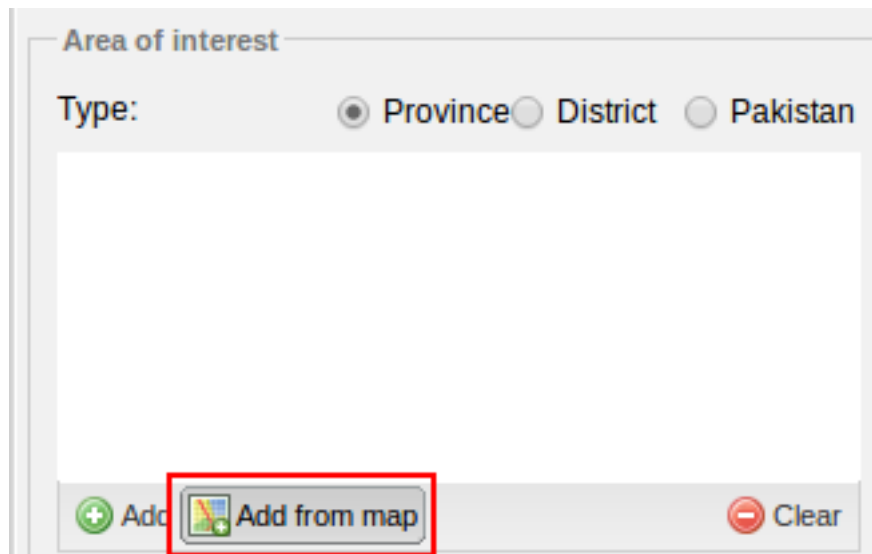


Fig. 3.68: Search autocomplete.

or Add from the map



The screenshot shows a web form titled "Area of interest". Below the title, there is a "Type:" label followed by three radio buttons: "Province" (which is selected), "District", and "Pakistan". Below the radio buttons is a large, empty rectangular box. At the bottom of the form, there are three buttons: a green button with a plus sign and the text "Add", a button with a map icon and the text "Add from map" (which is highlighted with a red rectangular box), and a red button with a minus sign and the text "Clear".

Fig. 3.69: Add from the map.

and click on the map on the zone to be added.

In both options, the zone will be added into the AoI grid

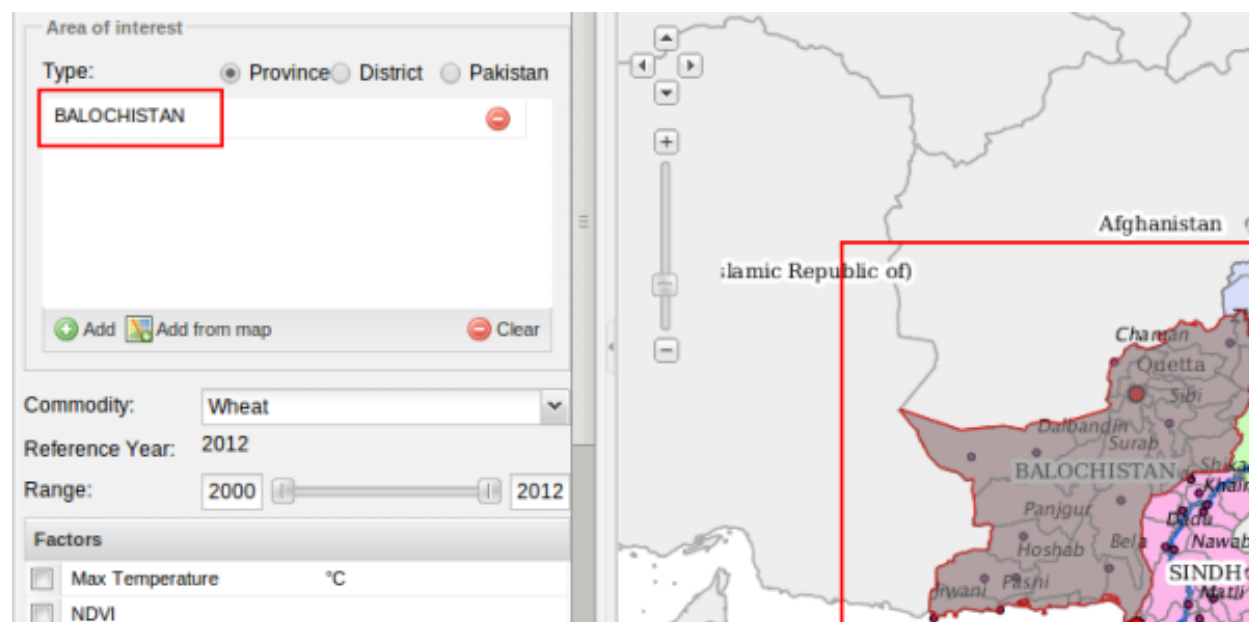


Fig. 3.70: Balochistan province selected as AoI.

If you select 'Pakistan', you don't have to select any AoI:

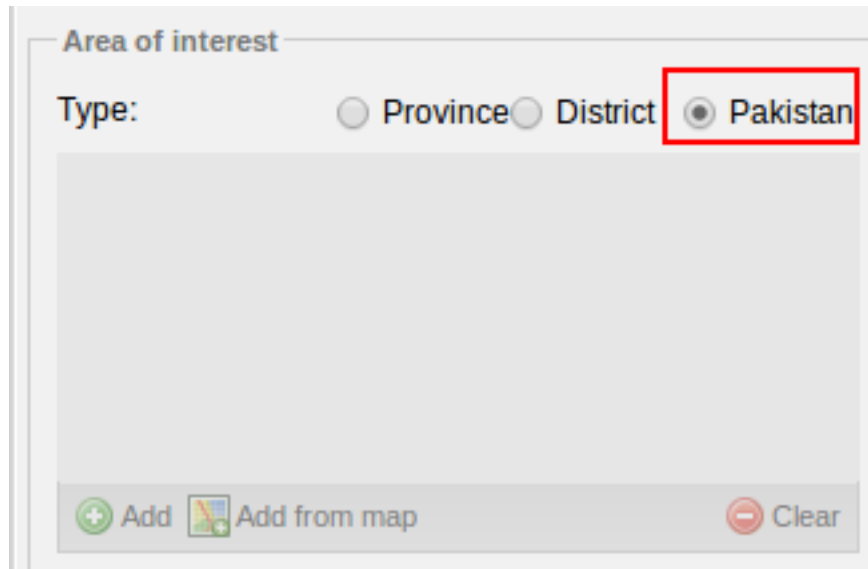


Fig. 3.71: Pakistan as AoI.

3.8.3 Commodity

Select the commodity for the report.

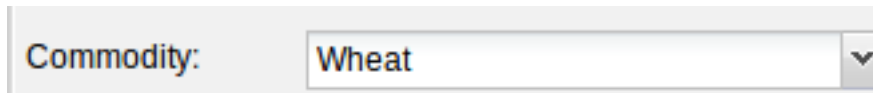


Fig. 3.72: Wheat as commodity.

3.8.4 Range

Select the range for the report. The last year of the range will be selected as Reference year for the map.

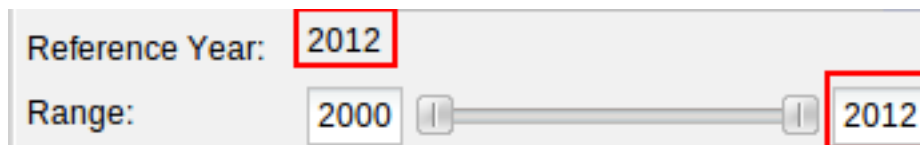
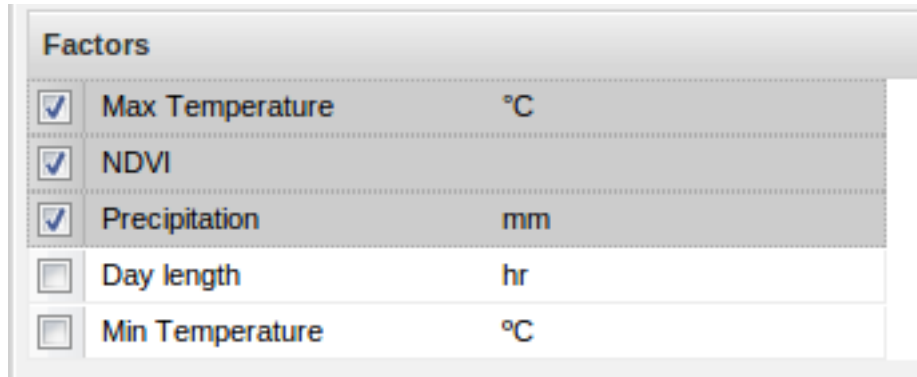


Fig. 3.73: Range and reference year selection.

3.8.5 Meteorological factors

Select the factors for the report. You can select one or more of this factors

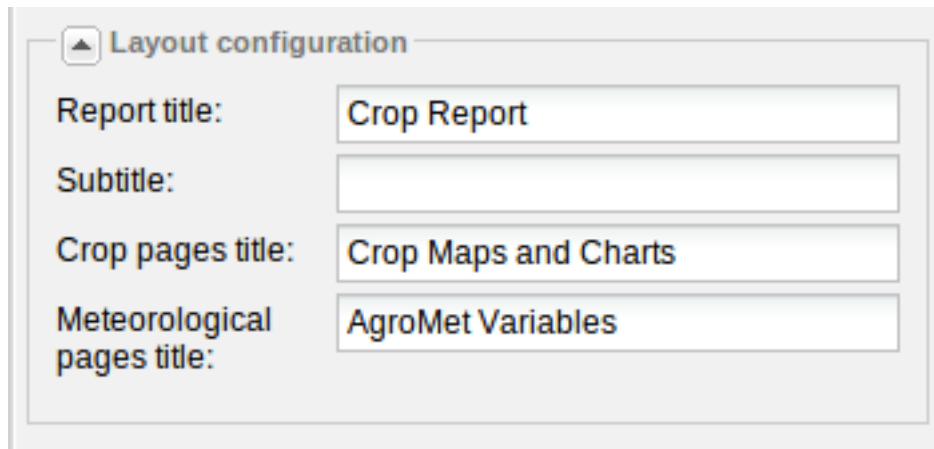


Factors		
<input checked="" type="checkbox"/>	Max Temperature	°C
<input checked="" type="checkbox"/>	NDVI	
<input checked="" type="checkbox"/>	Precipitation	mm
<input type="checkbox"/>	Day length	hr
<input type="checkbox"/>	Min Temperature	°C

Fig. 3.74: Range and reference year selection.

3.8.6 Layout configuration

In this section you can customize a little bit the generated report.



Layout configuration

Report title:

Subtitle:

Crop pages title:

Meteorological pages title:

Fig. 3.75: Default layout configuration.

- **Report title:** Title for the report (appear on the first page of the report)
- **Subtitle:** Subtitle for the report (appear on the first page of the report)
- **Crop pages title:** Title for the pages with maps and charts with the aggregated data
- **Meteorological pages title:** Title for pages with meteorological variables charts (the number of charts and pages depends on your factor selection)

3.8.7 Keep calm

Then you must press on Generate Report

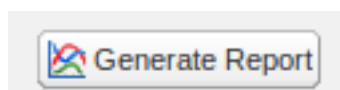
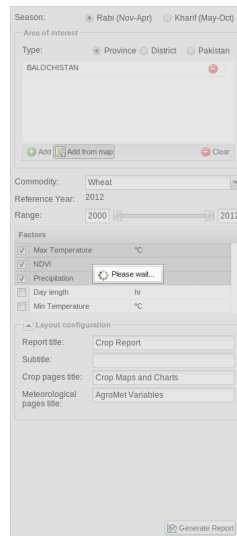


Fig. 3.76: Generate button.

and wait for the report generation.



The screenshot shows a web-based form for generating a crop report. The form is organized into several sections:

- Season:** Radio buttons for "Rabi (Nov-Apr)" (selected) and "Kharif (May-Oct)".
- Area of Interest:** A dropdown menu showing "BALUCHISTAN" with a red "X" icon.
- Type:** Radio buttons for "Province" (selected), "District", and "Pakistan".
- Buttons:** "Add" (green), "Add from map" (blue), and "Clear" (red).
- Commodity:** A dropdown menu showing "Wheat".
- Reference Year:** A text input field showing "2012".
- Range:** A range selector showing "2000" and "2012".
- Factors:** A list of checkboxes for "Max Temperature" (checked), "NDVI" (checked), "Precipitation" (checked), "Day length" (unchecked), and "Min Temperature" (unchecked). A "Please wait..." loading spinner is visible next to the "Precipitation" checkbox.
- Layout configuration:** A section with text input fields for "Report title" (filled with "Crop Report"), "Subtitle", "Crop pages title" (filled with "Crop Maps and Charts"), and "Meteorological pages title" (filled with "AgroMet Variables").
- Generate Report:** A button at the bottom right with a circular arrow icon.

Fig. 3.77: Wait for the report generation.

We process a lot of data and render it on a PDF document. When the report will be finished, your browser will ask you if you want to download the PDF file.

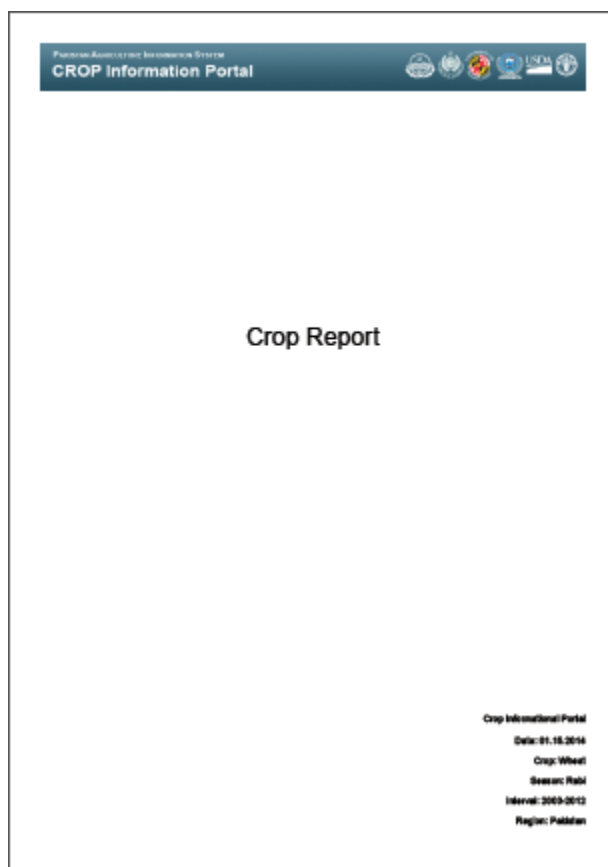


Fig. 3.78: The PDF cover

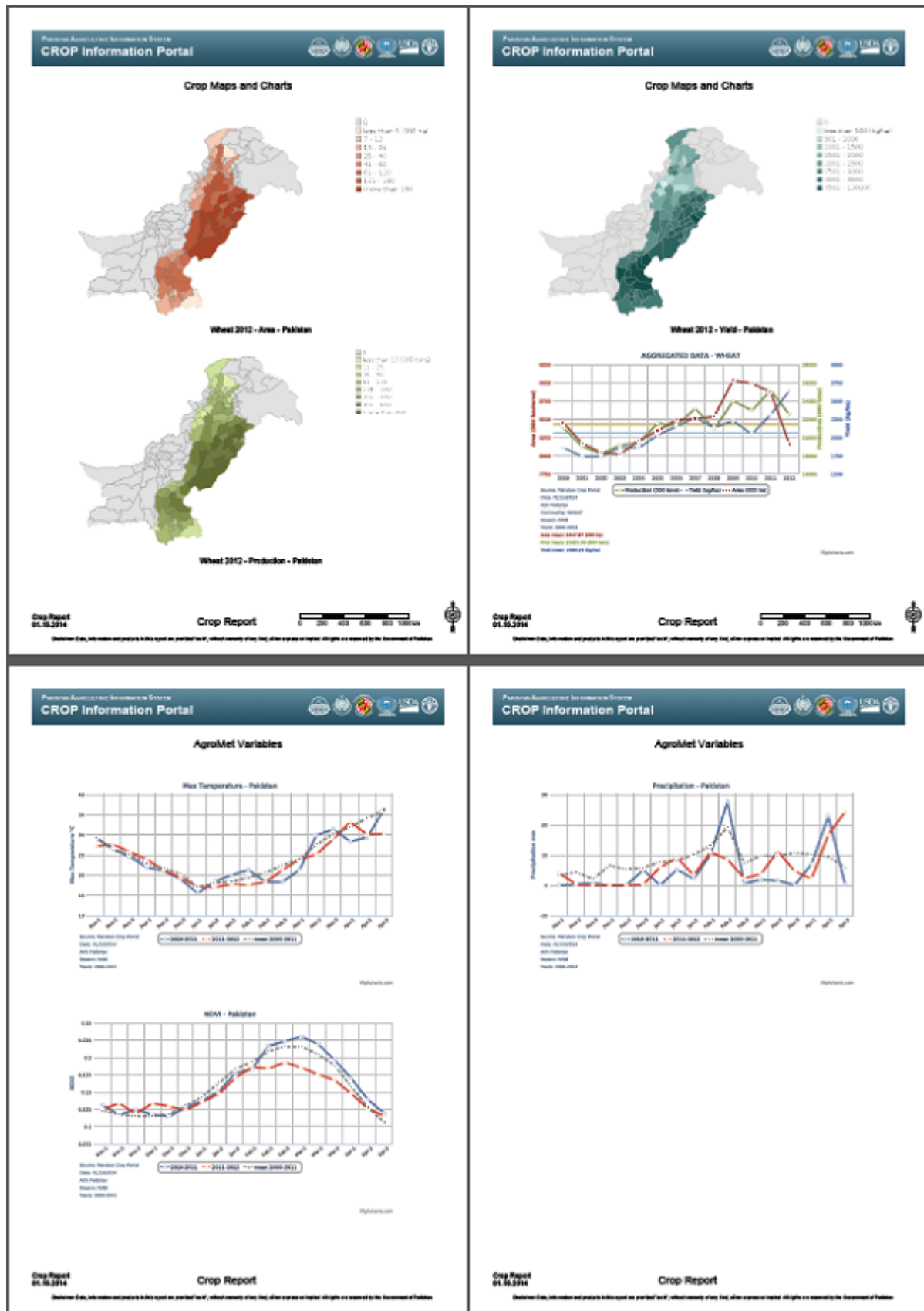


Fig. 3.79: The PDF content

Administation of Crop Information Portal

This section illustrates how manage users and keep updated the portal.

4.1 The Administration Page

The portal provides a lot of data to facilitate the analysis of the state of the crops in Pakistan.

These data must be continuously updated, and, where possible, automatically.

To allow mantainence operations, the portal has an administration page accessible from the User drop-down menu.

The tasks that can be performed are:

- administration of registered users for data download
- creation of new crops/agromet factors
- ingestion of new data or update existing values
- insertion of new NDVI images
- extraction of mean NDVI values per district/province using (optional) a crop mask.

4.1.1 Login

To access the administration area, you need to have an account of administrator.

Click on the Login button in the toolbar () and insert the account parameters:

Note: Only the users with the ADMIN role can access to the administration interface.

The default admin credentials are

- **user:**admin
- **password:**admin

Now you will have the option to access the reserved Administration Page.

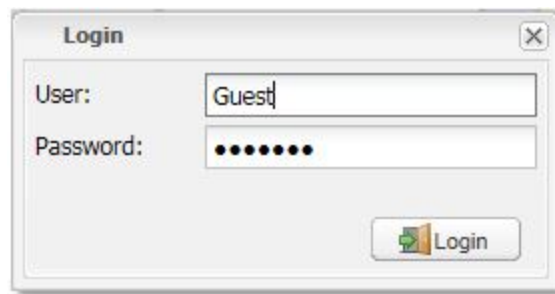


Fig. 4.1: Login Form.

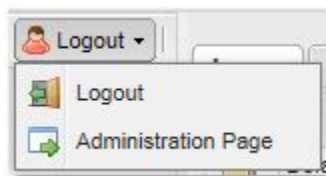


Fig. 4.2: Administration Page option.

4.1.2 User interface and tasks

For details on the tasks performed, procedures and practical examples, refer to the Administration manual. Here is an overview of look and feel and functionalities.

All tasks are distributed in four main groups:



Fig. 4.3: Administration Tools.

Workflows

Interface to manage data.

Database updates are conducted running tasks in this section.

Any tasks performed in the workflows group are logged and can be retrieved for analysis of results.



Fig. 4.4: Workflows tool.

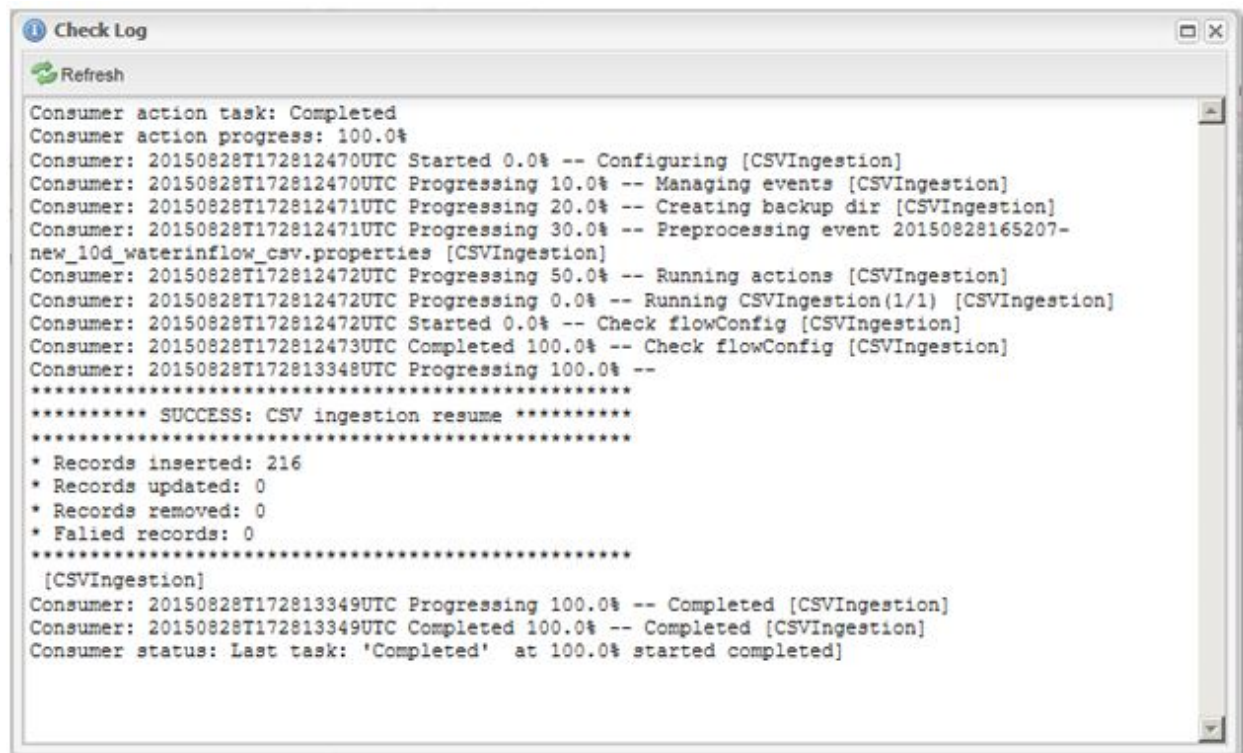


Fig. 4.5: Workflows tool.

User manager

Interface to manage users.

An user can have roles of guest (data download) and administration.

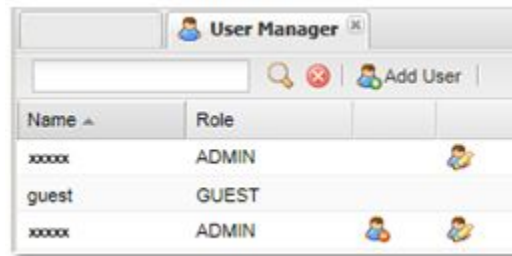


Fig. 4.6: User Manager.

File manager

Data and service files are uploaded and managed; for example the database is updated starting from CSV files uploaded here.

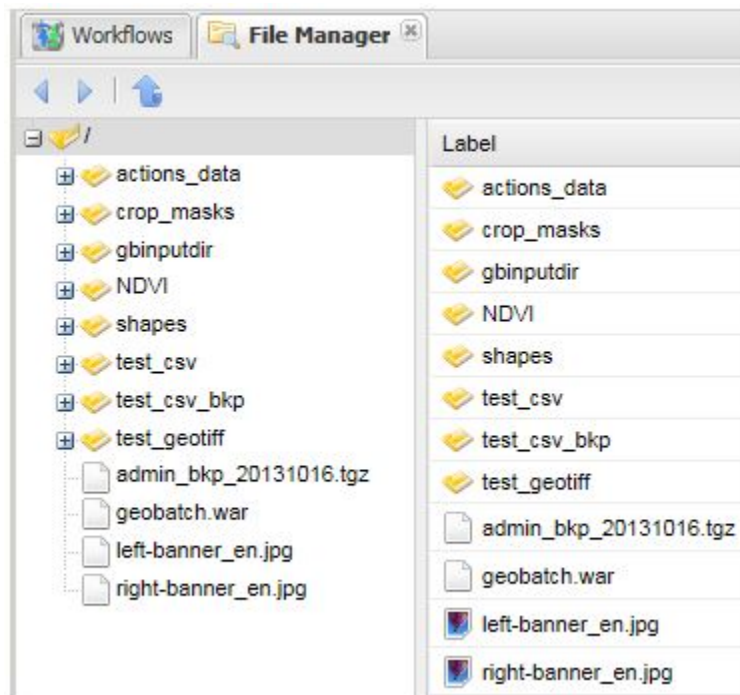
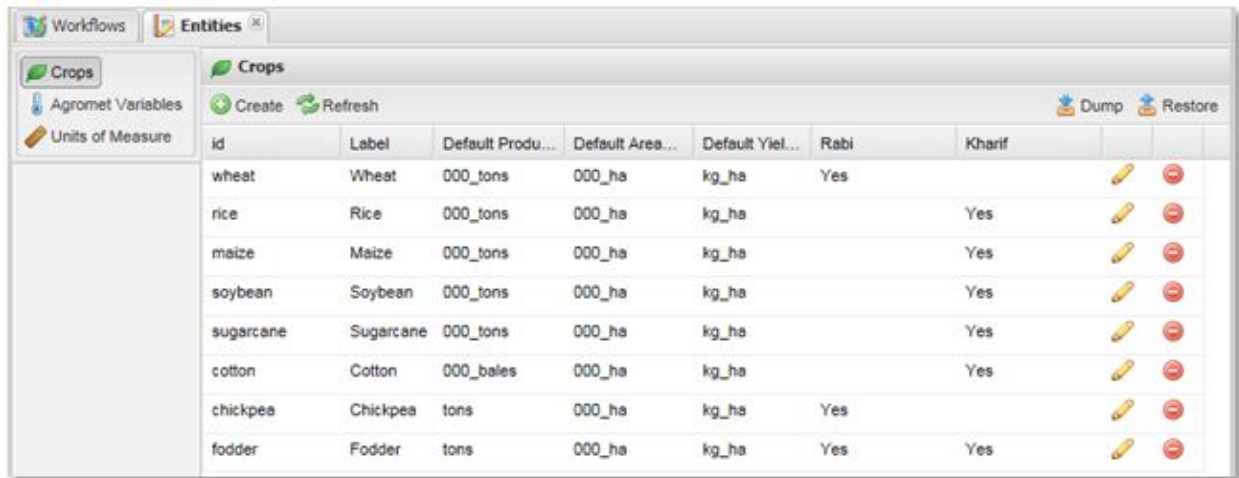


Fig. 4.7: File manager.

Entities manager

Crop and agromet variables are configured; units of measures are introduced and assigned to commodities or variables.

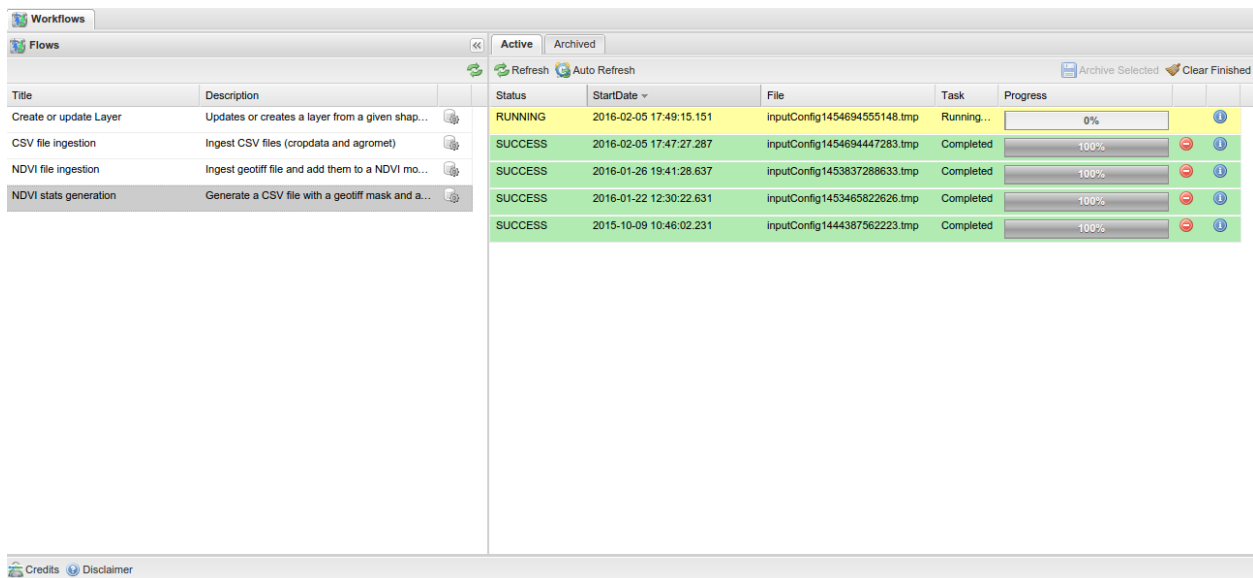


The screenshot shows the 'Entities' tab in the application. On the left, there is a sidebar with 'Crops', 'Agromet Variables', and 'Units of Measure'. The main area is titled 'Crops' and contains a table with columns: id, Label, Default Produ..., Default Area..., Default Yiel..., Rabi, and Kharif. There are also 'Create', 'Refresh', 'Dump', and 'Restore' buttons. The table lists eight crops: wheat, rice, maize, soybean, sugarcane, cotton, chickpea, and fodder, each with its respective units and seasonal availability.

id	Label	Default Produ...	Default Area...	Default Yiel...	Rabi	Kharif
wheat	Wheat	000_tons	000_ha	kg_ha	Yes	
rice	Rice	000_tons	000_ha	kg_ha		Yes
maize	Maize	000_tons	000_ha	kg_ha		Yes
soybean	Soybean	000_tons	000_ha	kg_ha		Yes
sugarcane	Sugarcane	000_tons	000_ha	kg_ha		Yes
cotton	Cotton	000_bales	000_ha	kg_ha		Yes
chickpea	Chickpea	tons	000_ha	kg_ha	Yes	
fodder	Fodder	tons	000_ha	kg_ha	Yes	Yes

Fig. 4.8: File manager.

4.2 The Workflows Tool



The screenshot shows the 'Workflows' tab in the application. It displays a list of workflow tasks with columns: Title, Description, Status, StartDate, File, Task, and Progress. The tasks include 'Create or update Layer', 'CSV file ingestion', 'NDVI file ingestion', and 'NDVI stats generation'. The first task is currently 'RUNNING', while the others are 'SUCCESS'.





Title	Description	Status	StartDate	File	Task	Progress
Create or update Layer	Updates or creates a layer from a given shap...	RUNNING	2016-02-05 17:49:15.151	inputConfig1454694555148.tmp	Running...	0%
CSV file ingestion	Ingest CSV files (cropland and agromet)	SUCCESS	2016-02-05 17:47:27.287	inputConfig1454694447283.tmp	Completed	100%
NDVI file ingestion	Ingest geotiff file and add them to a NDVI mo...	SUCCESS	2016-01-26 19:41:28.637	inputConfig1453837288633.tmp	Completed	100%
NDVI stats generation	Generate a CSV file with a geotiff mask and a...	SUCCESS	2016-01-22 12:30:22.631	inputConfig1453465822626.tmp	Completed	100%
		SUCCESS	2015-10-09 10:46:02.231	inputConfig1444387562223.tmp	Completed	100%

The “Workflows” tool provide functionalities to run common tasks on the server.

The available flows are:

- Create or Update Layer
- CSV file ingestion
- NDVI ingestion

- NDVI stats generation

Create or update Layer	Updates or creates a layer from a given shap...	
CSV file ingestion	Ingest CSV files (cropdata and agromet)	
NDVI file ingestion	Ingest geotiff file and add them to a NDVI mo...	
NDVI stats generation	Generate a CSV file with a geotiff mask and a...	

4.2.1 Create or update Layer

This flow allows to insert or update an existing vector layer in the database using a *shape file* as input. The system use the name of the shape file to publish a new spatial table on the database, or to update an existing one.

4.2.2 CSV file ingestion

This flow allows to update the data of the crop information portal using the csv file. This is the main way to update:

- Crop Data
- Agromet Variables Data
- Fertilizers Data
- Irrigation Data
- Market prices Data
- Crop Status Limits

Each CSV type has its own format and its own options.

4.2.3 NDVI ingestion

This flow allows to update the NDVI images published on the server. The system parse the file name to identify the *dekad* for the image.

for instance the dv16013_pak.tif is related to 2016, Jan, 3rd dekad.

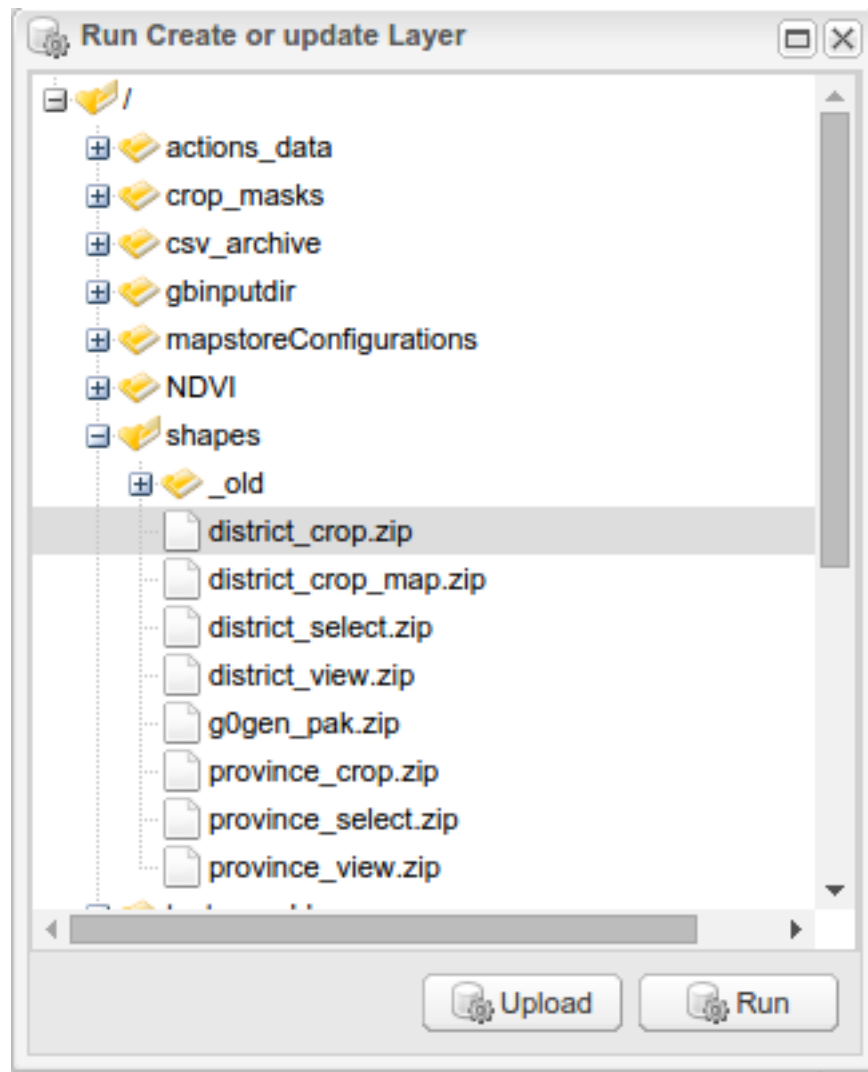
4.2.4 NDVI stats generation

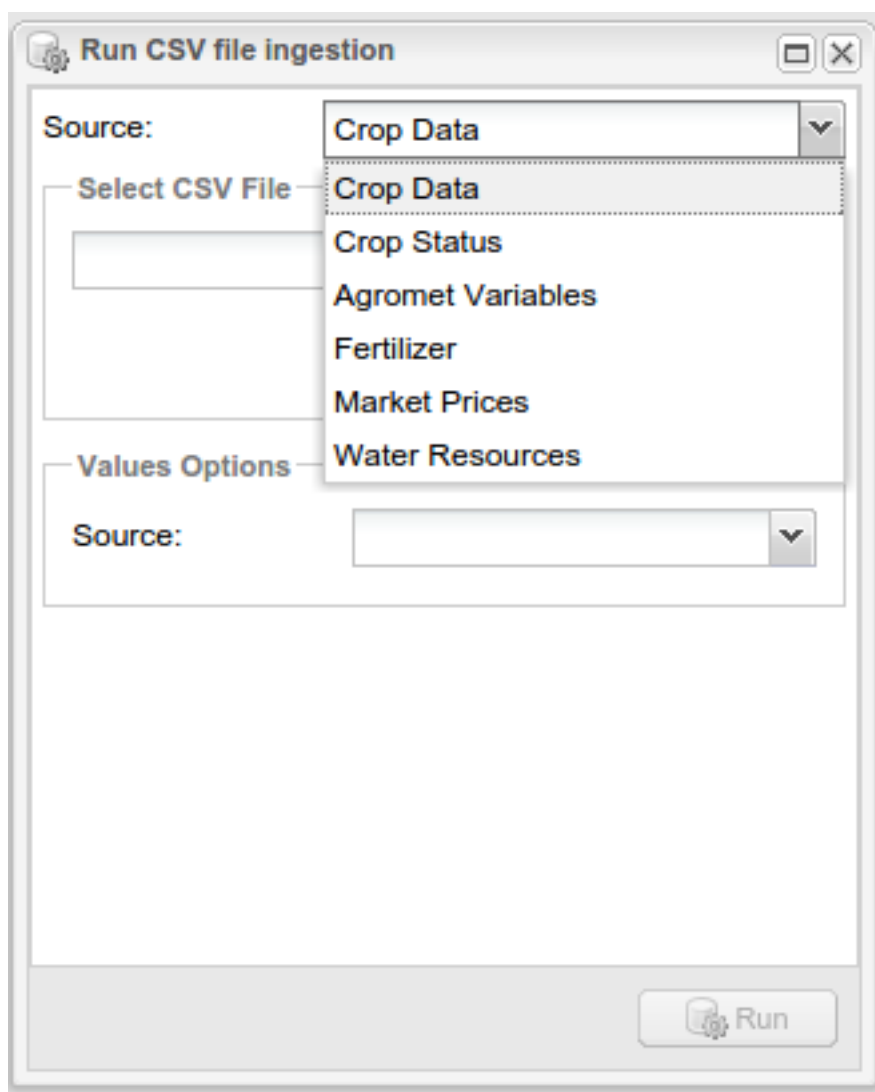
Generates statistics from the NDVI images published on the server, in CSV format. The generated CSV file can also be used in the *CSV file ingestion* flow to update the values of the NDVI Agromet Variable.

4.3 User Management Page

This section illustrates how manage users from the User management page.

By clicking on the section **User Manager** the administrators of the system can access to the User management page





The image shows a software dialog box titled "Run CSV file ingestion". It contains several input fields and a dropdown menu. The "Source:" label is followed by a dropdown menu currently showing "Crop Data". Below this, the "Select CSV File" label is followed by an empty text input field. The "Values Options" label is followed by another dropdown menu, also currently showing "Crop Data". A "Run" button with a gear icon is located at the bottom right of the dialog box.

Run CSV file ingestion

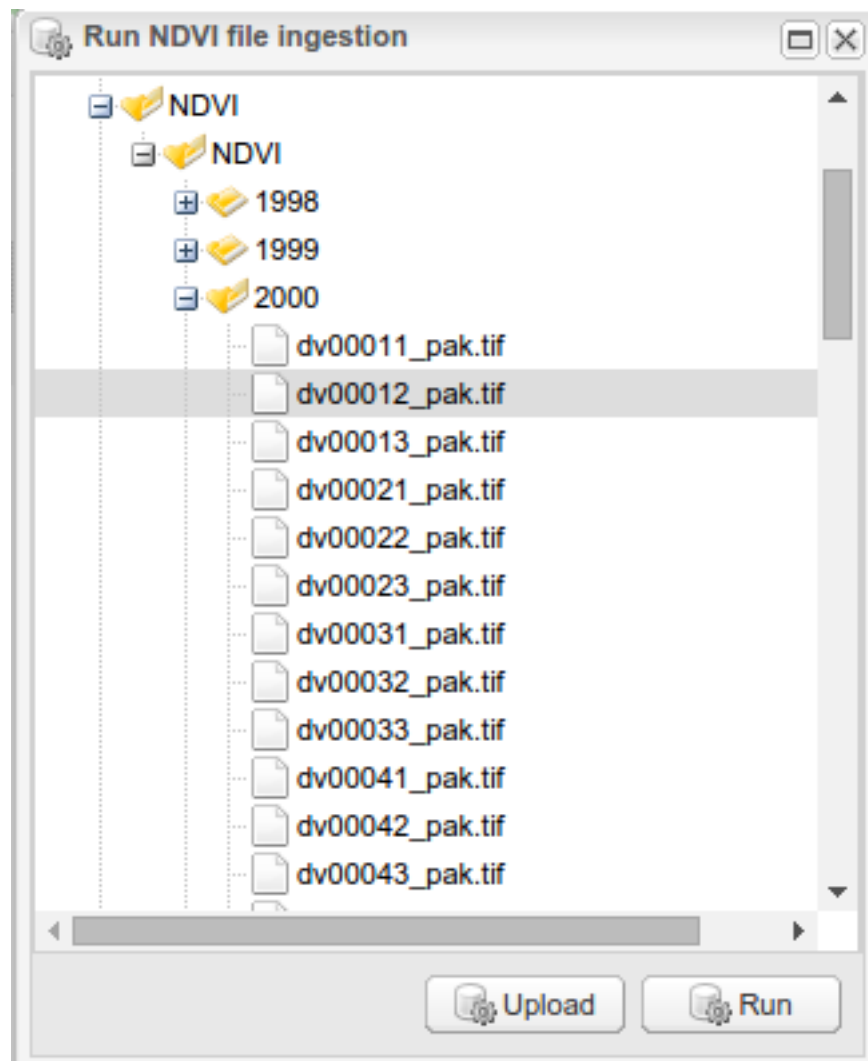
Source: Crop Data

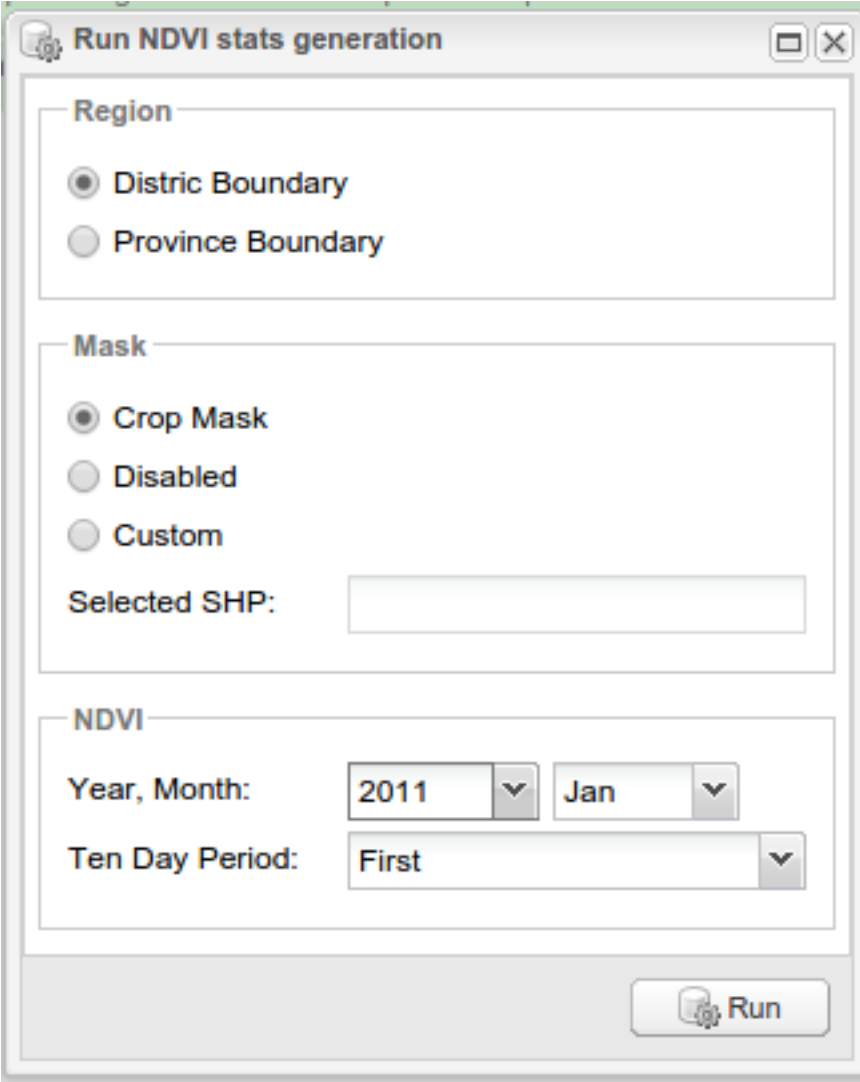
Select CSV File

Values Options

Source:

Run





The dialog box titled "Run NDVI stats generation" contains three sections: "Region", "Mask", and "NDVI".

Region:

- ☒ Distric Boundary
- ☐ Province Boundary

Mask:

- ☒ Crop Mask
- ☐ Disabled
- ☐ Custom

Selected SHP:

NDVI:

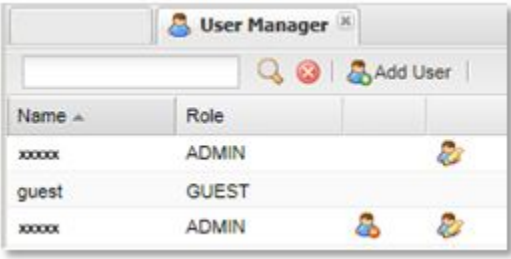
Year, Month: 2011 Jan

Ten Day Period: First

Run

Workflows User Manager File Manager Entities

Fig. 4.9: Access to User Management page.



The User Manager window displays a table of users and an "Add User" button.

Name	Role	
xxxxxx	ADMIN	
guest	GUEST	
xxxxxx	ADMIN	

Add User

Fig. 4.10: User Management page.

From this page you can see the list of users and their **ROLE**.

The users with **ROLE ADMIN** can access to the administration interface.

The users with **ROLE USER** are only allowed to use Data output type in the tools in mapstore.

By clicking on **Add User** button you can create new users.

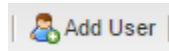


Fig. 4.11: Add User button.

From the form shown below you must insert an User Name, its **ROLE** (USER or ADMIN) and a password for the created user.

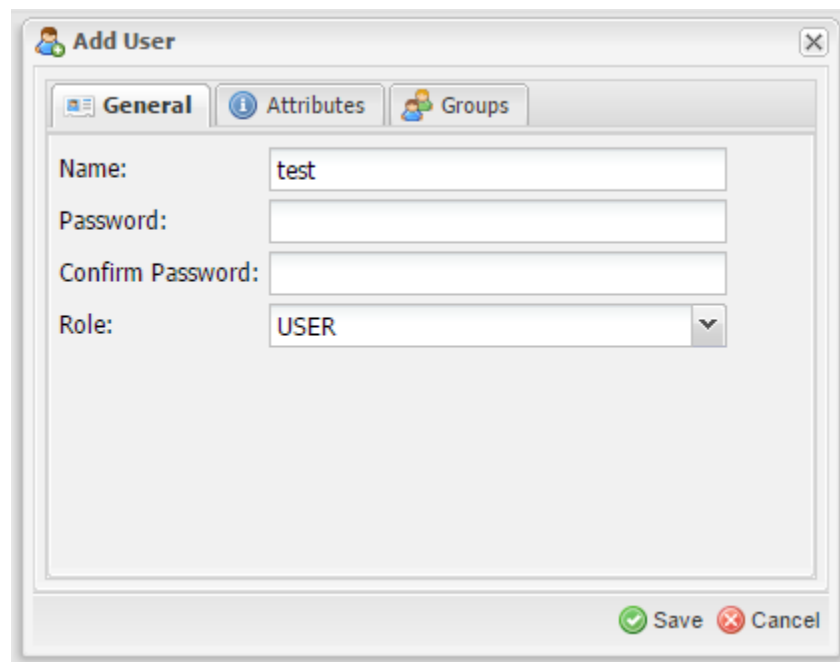
A screenshot of a web application window titled "Add User". The window has a close button (X) in the top right corner. Below the title bar are three tabs: "General" (selected), "Attributes", and "Groups". The "General" tab contains four form fields: "Name:" with the value "test", "Password:", "Confirm Password:", and "Role:" with a dropdown menu showing "USER". At the bottom right of the window are two buttons: "Save" with a green checkmark icon and "Cancel" with a red X icon.

Fig. 4.12: Create User form, General tab.

You can optionally fill other user details, as e-mail address, company name and other notes.

Each user can be part of different groups.

By clicking on **Save** button you submit the creation of new user.

Now the list of users shows the new user.

The screenshot shows a window titled "Add User" with a close button (X) in the top right corner. Below the title bar are three tabs: "General", "Attributes" (which is selected and highlighted with an information icon), and "Groups". The "Attributes" tab contains three input fields: "email:" with a single-line text box, "Company:" with a single-line text box, and "Notes:" with a multi-line text area. At the bottom right of the window are two buttons: a green "Save" button with a checkmark icon and a red "Cancel" button with an X icon.

Fig. 4.13: Create User form, Attributes tab.

The screenshot shows the same "Add User" window, but with the "Groups" tab selected. The "Groups" tab is divided into two main sections: "Available" on the left and "Selected" on the right. The "Available" section contains a list box with the item "test" selected. The "Selected" section is currently empty. Between these two sections is a vertical column of six buttons: an up arrow, a down arrow, a right arrow, a left arrow, an up arrow, and a down arrow. At the bottom right of the window are the same "Save" and "Cancel" buttons as in the previous screenshot.

Fig. 4.14: Create User form, Groups tab.

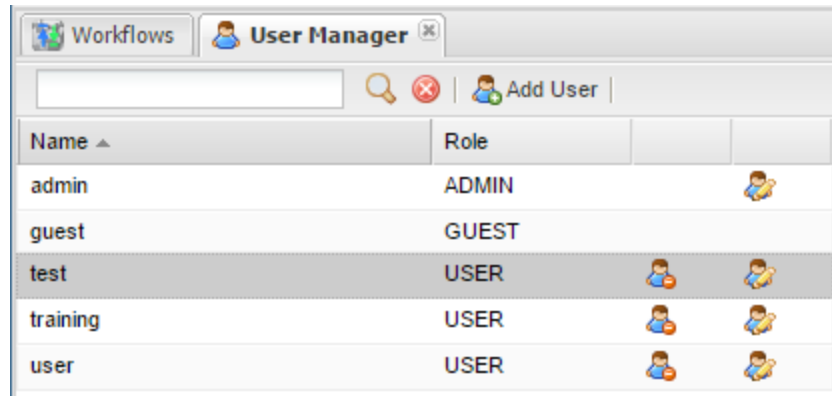


Fig. 4.15: User added.

Using the icons on the right of each user row you can edit or delete the users as shown by the forms below. From the Edit User form you can change every user attribute (password, role and other attributes), but the username cannot be changed.

The 'Edit user data' window has three tabs: 'General', 'Attributes', and 'Groups'. The 'General' tab is active, showing the following fields:

- Name: test
- New Password:
- Confirm Password:
- Role: USER (dropdown menu)

At the bottom right, there are 'Save' and 'Cancel' buttons.

Fig. 4.16: User edit.

4.4 Crops, Agromet Variables and Units of Measure

To manage Crops, Agromet Variables and Units of Measure you can use the **Entities manager** tool in the Administration Page

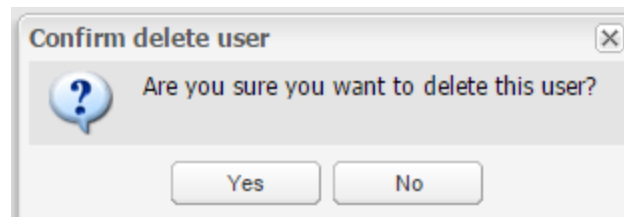


Fig. 4.17: User delete.

id	Label	Default Produ...	Default Area...	Default Yiel...	Rabi	Kharif
wheat	Wheat	000_tons	000_ha	kg_ha	Yes	
rice	Rice	000_tons	000_ha	kg_ha		Yes
maize	Maize	000_tons	000_ha	kg_ha		Yes
soybean	Soybean	000_tons	000_ha	kg_ha		Yes
sugarcane	Sugarcane	000_tons	000_ha	kg_ha		Yes
cotton	Cotton	000_bales	000_ha	kg_ha		Yes
chickpea	Chickpea	tons	000_ha	kg_ha	Yes	
fodder	Fodder	tons	000_ha	kg_ha	Yes	Yes

Fig. 4.18: Entities Manager tool

4.4.1 Manage Crops

Create a new Crop

To create a new crop, click on the **Create** button. A form with the information to set will be displayed. Complete the form as follows:

The new crop will be shown in the Crops list.

Anyway the new crop will not be available in MapStore until some data about the crop is ingested.

Id: soybean

Label: Soybean

Season: ☐ Rabi ☒ Kharif

Production Uom: 000 tons

Area Uom: ha

Yield Uom: kg/ha

Save

Fig. 4.19: Sample Soybean crop

Workflows

Entities

Crops

Agromet Variables

Units of Measure

Crops

Create Refresh

Dump Restore

id	Label	Default Production Uom	Default Area Uom	Default Yield Uom	Rabi	Kharif		
wheat	Wheat	000_tons	000_ha	kg_ha	Yes			
rice	Rice	000_tons	000_ha	kg_ha		Yes		
maize	Maize	000_tons	000_ha	kg_ha		Yes		
soybean	Soybean	000_tons	000_ha	kg_ha		Yes		
sugarcane	Sugarcane	000_tons	000_ha	kg_ha		Yes		
cotton	Cotton	000_bales	000_ha	kg_ha		Yes		
chickpea	Chickpea	tons	000_ha	kg_ha	Yes			
fodder	Fodder	tons	000_ha	kg_ha	Yes	Yes		

Fig. 4.20: The new crop is added to the list

How to allow map generation for new Crops

To generate maps you **must** add 6 new styles to geoserver with a specific name:

`<province|district>_<crop_id_lower_case>_<area|prod|yield>_style`

In the case above the styles to add will have the following names:

- province_soybean_area_style
- province_soybean_prod_style
- province_soybean_yield_style
- district_soybean_area_style
- district_soybean_prod_style
- district_soybean_yield_style

Note: These styles are needed to generate maps in the *Crop Data* module of *MapStore*.

- **Area Type: National(Province)**

- province_soybean_area_style.xml: style to generate maps for area.
 - province_soybean_prod_style.xml: style to generate maps for production.
 - province_soybean_yield_style.xml: style to generate maps for yield.

- **Area Type: National(District) or Province(District)**

- district_soybean_area_style.xml: style to generate maps for area .
 - district_soybean_prod_style.xml: style to generate maps for production.
 - district_soybean_yield_style.xml: style to generate maps for yield.
-

Add a SLD style You can access to the style layer descriptor (SLD) definition page on GeoServer with [this link](#) or selecting *Styles* module on GeoServer:

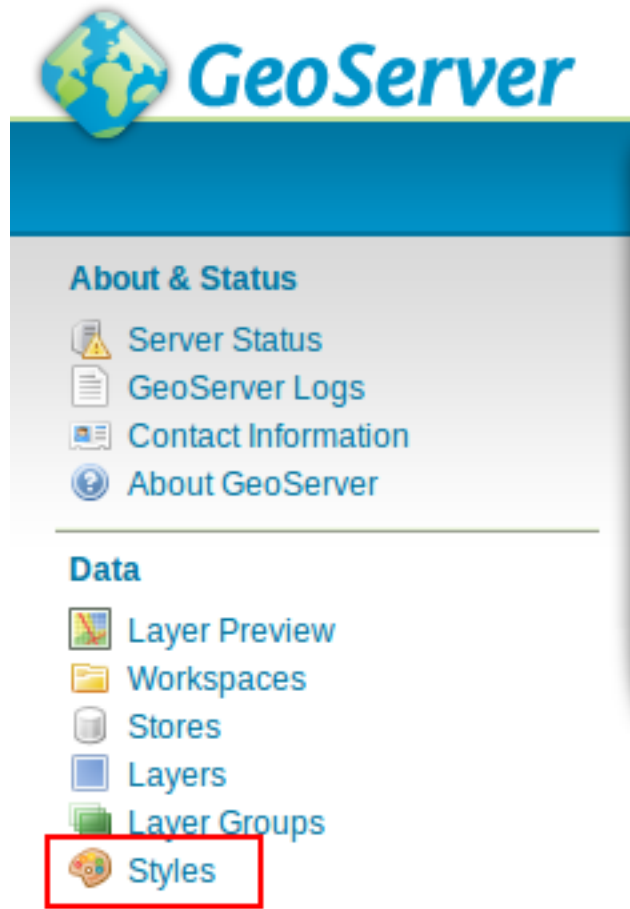


Fig. 4.21: Styles module on GeoServer

and press on *add a new style*

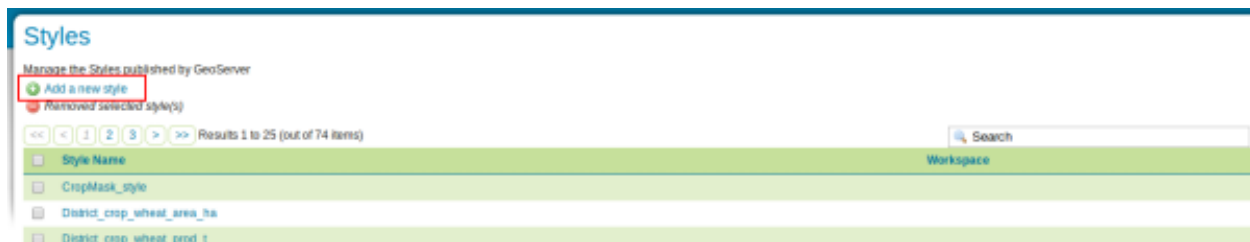


Fig. 4.22: Add a new style button

Now, you have to fill the form with the new name, for this exercise you can copy a style from the existing ones.

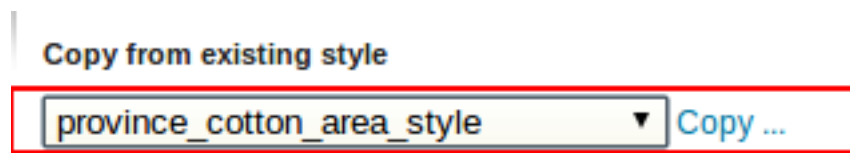


Fig. 4.23: Copy *province_cotton_area_style*

change the style with the new name and modify filters as you want:

copy from one of the present one:



Fig. 4.24: Edit the style

validate and submit the new style:

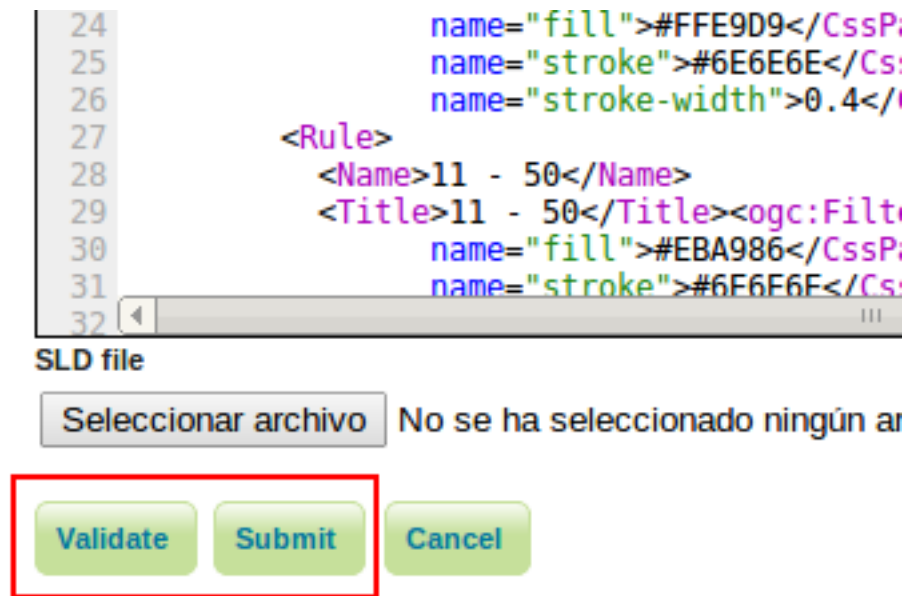


Fig. 4.25: Styles submit

To see a complete example of how to add a crop see [Complete Example](#).

Edit/Delete Crops

You can edit/delete the crops clicking on the **Edit** and **Delete** buttons on the right of each row.

4.4.2 Manage Agrometeorological Variables

Create a new Variable

To create a new variable, click on the **Create** button. A form with the information to set will be displayed. Complete the form as follows:

The new variable will be shown in the Crops list.

Anyway the new variable will not be available in MapStore until some data about the variable is ingested.

Edit/Delete Variables

You can edit/delete the variable clicking on the **Edit** and **Delete** buttons on the right of each row.

Note: NDVI_avg is used in a special way from the system, so it should never be deleted. otherwise the ingestion of generated statistics from the NDVI statistics module will not work.

Factor: ET_avg

Label: Evapotranspiration

Unit: mm/day

Aggregation: ☒ avg ☐ sum

Save

Fig. 4.26: Sample Evapotranspiration variable

Workflows		Entities	
Crops		Agromet Variables	
Agromet Variables		Create Refresh Dump Restore	
Units of Measure			
Factor	Label	Aggregation	Unit
NDVI_avg	NDVI	avg	
ppt_sum_mm	Precipitation	avg	mm
Daylen_avg	Day length	avg	hr
Tmin_avg	Min Temperature	avg	°C
ET_avg	Evapotranspiration	avg	mm/day
Tmax_avg	Max Temperature	avg	°C

Fig. 4.27: The new variable is added to the list

4.4.3 Complete example

You can follow a complete example adding the data and styles and testing it on mapstore

Complete example

Note: All resources for this exercise are present on the folder `ROOT\training\data\crop_agro` of the training. Please open the folder and use the files inside to complete the example.

We have attached data and styles for a complete process to add a new crop with data and styles to be used on the portal:

1. **Create the new crop:** follow the instructions in Create a new Crop section and create a new crop with the identifier *soybean*
2. **Create the styles:** Create a new style of each needed type (use xml files inside the training folder):
 - `province_soybean_area_style.xml`
 - `province_soybean_prod_style.xml`
 - `province_soybean_yield_style.xml`
 - `district_soybean_area_style.xml`
 - `district_soybean_prod_style.xml`
 - `district_soybean_yield_style.xml`

After this step you will have the new styles published on GeoServer:

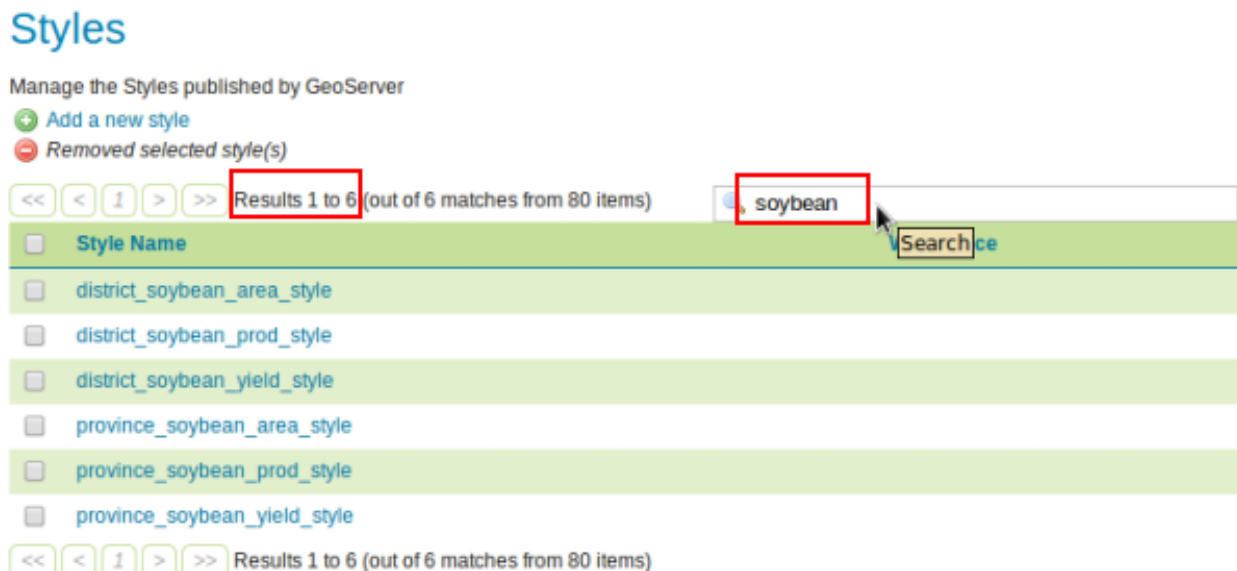


Fig. 4.28: Example's styles

3. **Import the data:** Follow the instructions on CSV ingestion section with the file `soybean_10-12.csv`
4. **Test data:** Follow the instructions on crop data tool section to use the new crop:

As map. You must select a correct year in the range (2010-2012). The map should be something like that:

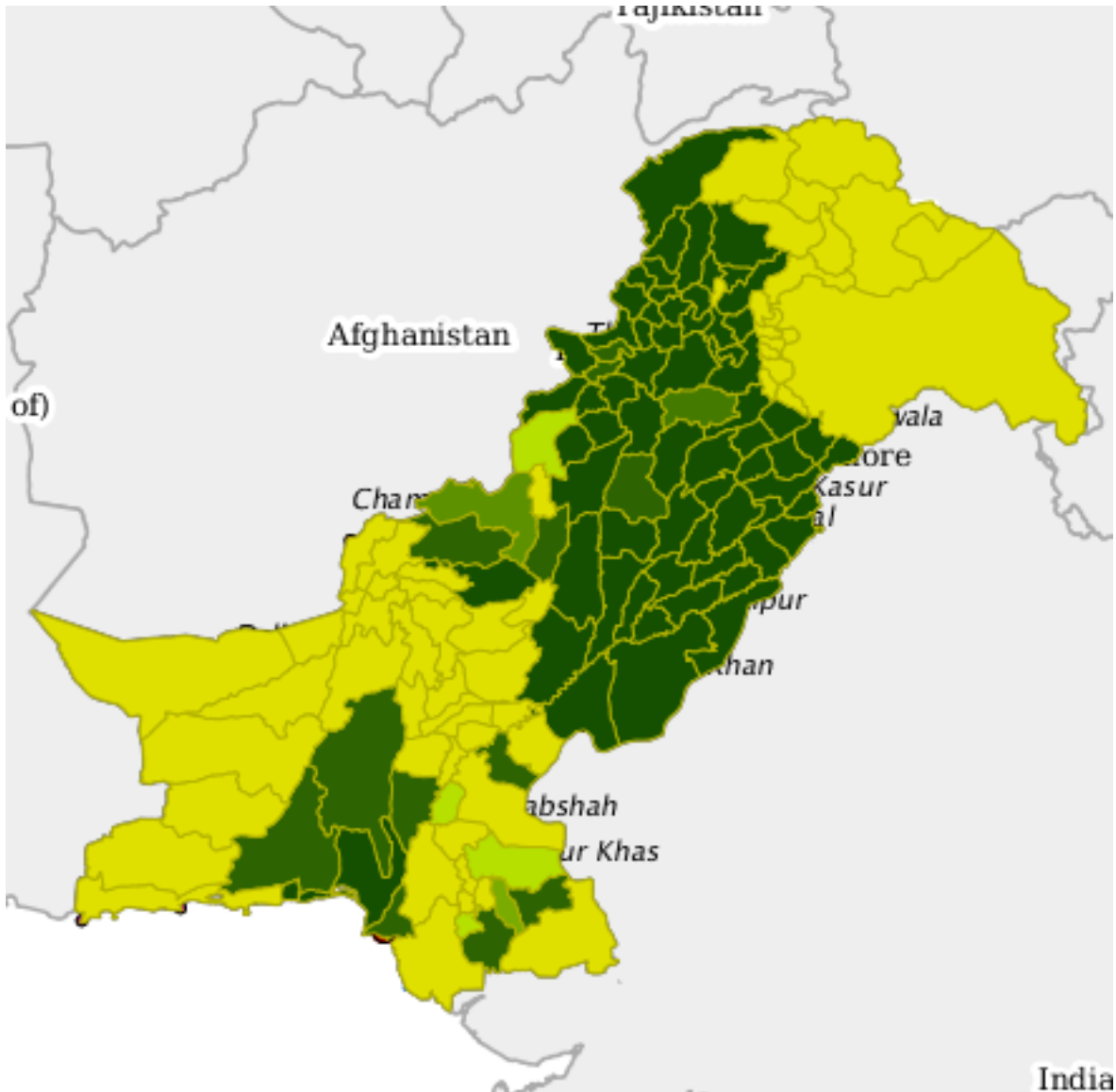


Fig. 4.29: Expected map (Soybean 2010 - Yield)

As chart: You must select the correct interval (2010-2012) and district or provinces with data.

5. **Delete the test data:** Follow the instructions on CSV ingestion section with the file `soybean_10-12_del.csv` and change the zoom level. You should see the same color on all province/districts.

4.4.4 Unit of Measure

The units of measure used in the crop information portal can be configured here.

The attribute of a unit of measure object are:

- **id:** the identifier of the unit of measure

ID	Label	Short Name	Description	Class	factor	filter		
000_ha	000 hectares	000 ha	1000 hectares	area	1			
000_tons	000 tons	000 tons	1000 tons	production	1			
tons_ha	t/ha	t/ha	tons / ha	yield	1			
kg_ha	kg/ha	kg/ha	kg/ha	yield	1000			
kg	Kilograms	kg	Kilograms	production	1000000			
tons	tons	tons	tonnes	production	1000			
ha	ha	ha	hectare = 10000 sqm	area	1000			
000_acres	000 acres	000 a	1000 acres (1 a = 0.404678363 ha)	area	2.471098263			
acres	acres	acres	acres = 404.678363 (000 ha)	area	2471.09823			
currency_1kg	1 Kg	1		denominator	0.01			
000_bales	000 bales	000_bales	1000 bales (1 bale = 170 kg)	production	5.882352941	cotton		
000_maunds	000 maunds	000 m	1000 maunds (1 maund = 37.324kg)	production	26.79241239	cotton		
mds_acr	m/a	m/a	maunds/acres	yield	10.84232395	cotton		
ton_acre	t/a	t/a	Tons / acres	yield	0.404678079			
USD	US Dollars	USD	dollars	currency	1			
currency_1bale	Bale	bale	conversion factor bale/100 kg = 1.7	denominator	1.7			
currency_100kg	100 Kg	100	factor to compute price for a unit of 100 kg weight	denominator	1			

Fig. 4.30: The unit of measure table

- **Label:** the label to use for this unit of measure
- **Short Name:** a short version of the label used in charts when needed.
- **Description:** a description of the unit of measure, is optional.
- **Class:** represent where this unit of measure is used in the system. Can be one of: * area: used for areas in crop data tool * production: used for production in crop data tool * yield: used for yield in crop data tool * denominator: used as a denominator in market prices tool (e.g. USD/40kg) * currency: only one is allowed. USD. * exchangerate: stores the current exchange rate PKR/USD. you can change the value of this with recent
 - waterflow: used for waterflow values in irrigation tool
 - watersupply: used for water supply values in irrigation tool
- **filter:** used only for production in crops. Can be populated with a comma-separated list of crop_id to use the unit of measure only with the crops listed.

As for the other entities you can create a new unit of measure clicking on the “Create” button

Clicking on the pencil in the table row you can edit a unit of measure.

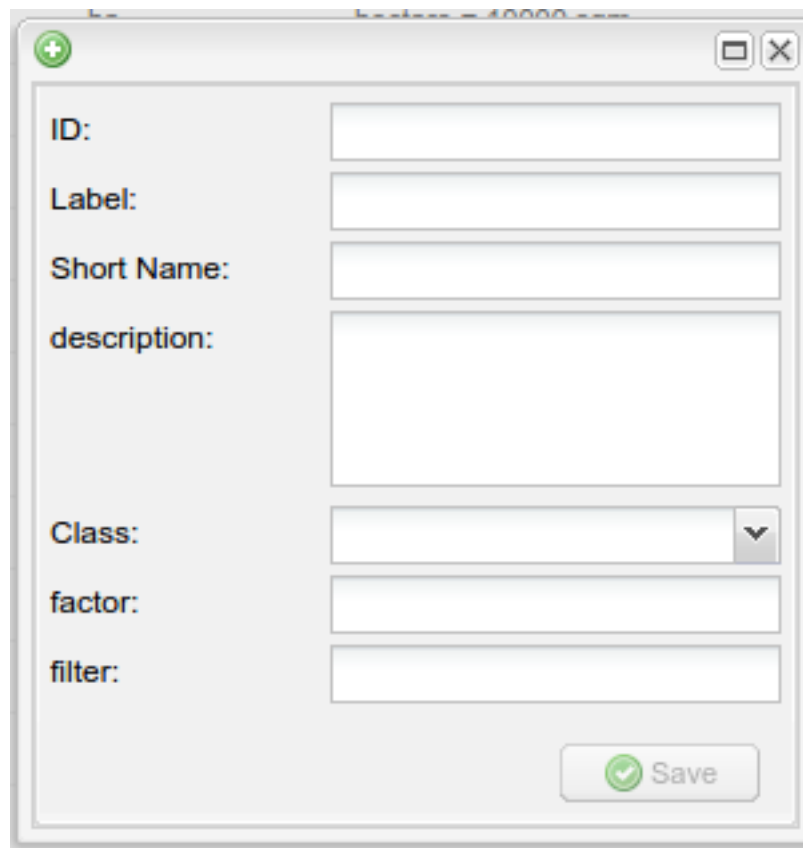
4.5 Architecture

The portal is composed of several web applications that interact with each other. Some of them are used directly by the users (MapStore, Administration Interface).

4.5.1 Components

The following are the components of the architecture of the Crop Information Portal.

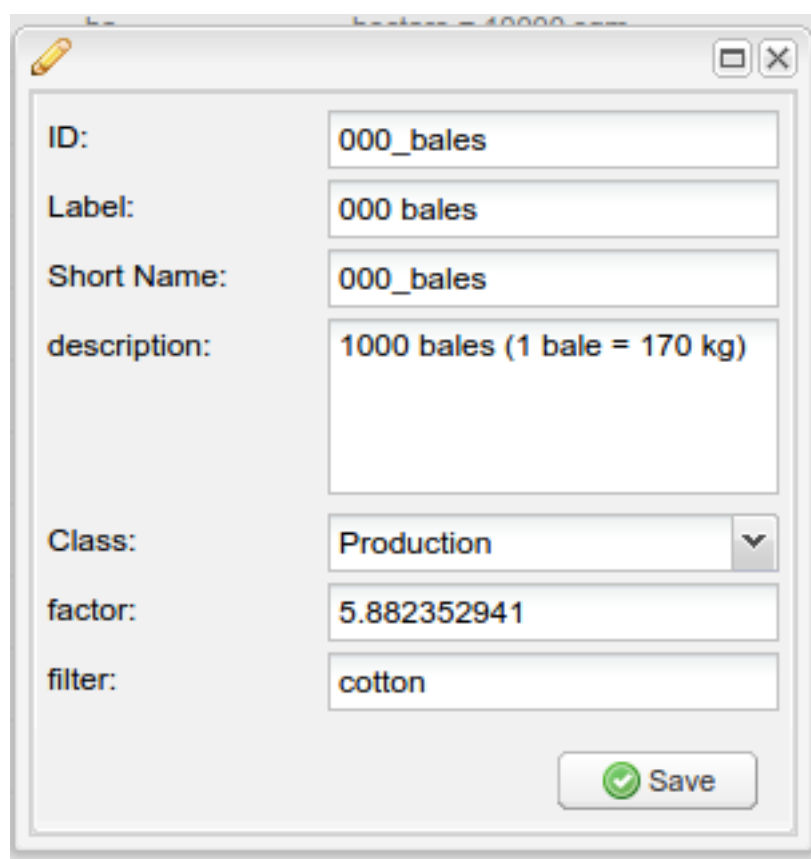
- **MapStore:** The main front-end to generate charts, maps and reports inferring agro-metereological data. Interacts with GeoServer to allow this kind of processing.



The screenshot shows a web form for creating a new unit of measure. The form is contained within a window with a title bar that includes a green plus icon on the left and minimize and close icons on the right. The form fields are arranged in a two-column layout. The left column contains labels: 'ID:', 'Label:', 'Short Name:', 'description:', 'Class:', 'factor:', and 'filter:'. The right column contains corresponding input fields: a text box for 'ID:', a text box for 'Label:', a text box for 'Short Name:', a larger text area for 'description:', a dropdown menu for 'Class:', a text box for 'factor:', and a text box for 'filter:'. At the bottom right of the form is a 'Save' button with a green checkmark icon.

ID:	<input type="text"/>
Label:	<input type="text"/>
Short Name:	<input type="text"/>
description:	<input type="text"/>
Class:	<input type="text"/>
factor:	<input type="text"/>
filter:	<input type="text"/>

Fig. 4.31: Create a new unit of measure



The screenshot shows a dialog box titled "Edit a unit of measure" with a pencil icon in the top-left corner and standard window controls in the top-right. The dialog contains the following fields:

Field	Value
ID:	000_bales
Label:	000 bales
Short Name:	000_bales
description:	1000 bales (1 bale = 170 kg)
Class:	Production
factor:	5.882352941
filter:	cotton

At the bottom right of the dialog is a "Save" button with a green checkmark icon.

Fig. 4.32: Edit a unit of measure.

- **Administration Interface:** The administrator front-end to update published data and generate statistics from raster images interacting with GeoBatch. Provides also user management functionalities.
- **GeoStore:** a non-sql REST access database used to store users data and to log their operations.
- **GeoServer** is an Open Source software server that allows users to share and edit geospatial data. Designed for interoperability, it publishes data from any major spatial data source using open standards.
- **GeoBatch** is an Open Source application for the collection, processing and publication of geospatial data in real time.
- **GeoNetwork** is an Open Source catalog application to manage spatially referenced resources.

4.5.2 General Architecture

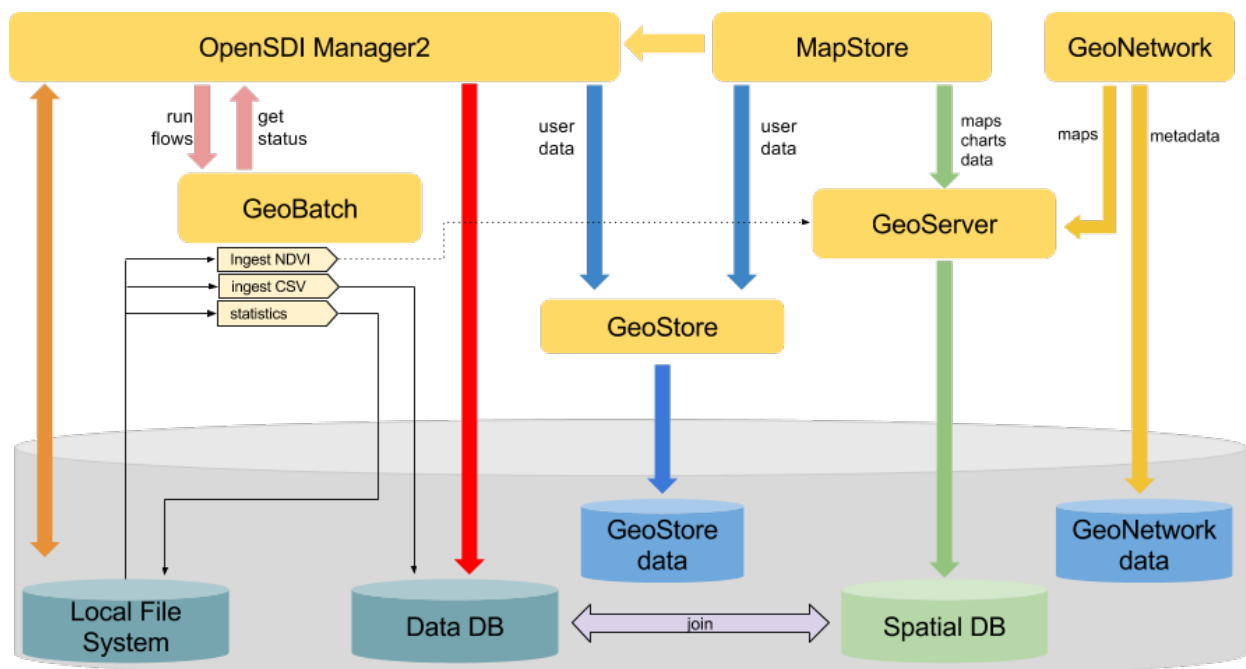


Fig. 4.33: Architecture of Crop Information Portal.

The *administrator interface* interacts directly with the file system, with the database and with *GeoBatch*. Allows to upload and download files and launch some *GeoBatch* flows on them. Provides also a direct interaction with the database to customize the agrometeorological factors and crops to show inside *MapStore*.

GeoBatch runs flows using files from the filesystem. Depending on the type of flow geobatch interacts with the other components (publish data in geoserver, update database, create csv files...).

MapStore uses the capabilities of *GeoServer* to show aggregated data and maps. The filtering and aggregation functionalities are implemented as parametric views on *GeoServer*. All the parametric views in *GeoServer* are optimized to aggregate and filter the data in the fastest way. MapStore requires the proper output format to geoserver in order to show tabular data, maps, charts or download CSV files.

GeoStore is used to manage user data that are not directly connected with the model.

There are other small web applications non present in the schema:

- **highcharts-export:** An application that allows to rasterize charts in various formats (*pdf, svg, png, jpeg*).
- **http_proxy :** A secure proxy that can allow *MapStore* to get maps from other qualified WMS sources.

- **xmlJsonTranslate** : a container of services required by *MapStore* grouped in a separated application.

A more detailed description about the model is available here:

Model

The NRL database contains the main model

This is the list of the tables containing data in the database:

Name	Description
agromet	data about agrometeorological factors
agrometdescriptor	information about specific factors
cropdata	data about crops (area,production,yield)
cropdescriptor	information about specific crop
fertilizer	data about fertilizer usage
cropstatus	thresholds for factor values related to a specific period of the year and crop
marketprice	data about market prices for each district
waterflow	data about river water inflow at rim stations
withdrawal	data about irrigation water supply
measure units	database of the units of measure

On the same database you will find also some GeoSpatial data:

Name	Description
g0gen_pak	world national boundaries. Used as background on the map.
na-tional_boundary	national boundary Pakistan. Used as Pakistan boundary on the map.
province_view	provinces in Pakistan.Used as province boundaries on the map.
province_crop	provinces in Pakistan to generate crop maps.
dis-trict_view	districts in Pakistan. Karachi area in 5 districts and frontier regions split from Districts. Used as district layer in TOC.
district_crop	Karachi one polygon and frontier regions merged to KP districts. Used for selection where data are distributed in this way (e.g. crop data, fertilizer...) .
dis-trict_crop_map	districts in Pakistan to generate crop maps.
dis-trict_select	Karachi area in 5 districts and frontier regions merged to KP districts. Used for selection where data are distributed in this way (agromet data...).

These

List of relations

measure_units Contains the unit of measure to use in the Crop Information Portal

schema	Column	Description
	id	the identifier for the unit of measure
	name	the name to use for this unit of version
	shortname	a short name for the unit of measure
	description	a description of the unit of measure, just mnemonic.
	cls	identify the class (area, production, yield, waterflow, withdrawal, denominator, exchangerate)
	filter	a comma separated list of strings to filter wich units are allowed for which data

Sample Content here some sample rows from the *measure_units* table

id	name	short-name	description	cls	coefficient	filter
000_ha	000 hectares	000 ha	1000 hectares	area	1	
000_tons	000 tons	000 tons	1000 tons	production	1	
tons_ha	t/ha	t/ha	tons / ha	yield	1	
kg_ha	kg/ha	kg/ha	kg/ha	yield	1000	
kg	Kilograms	kg	Kilograms	production	1000000	
tons	tons	tons	tonnes	production	1000	
ha	ha	ha	hectare = 10000 sqm	area	1000	
flow_cubicfeet	000 Cubic Feet per Second	000 Cusec	1 [000 Cusec] = 0.0283168467 [000 cubic meter/sec]	water-flow	1	irrigation
000_acres	000 acres	000 a	1000 acres (1 a = 0.404678363 ha)	area	2.471098263	
acres	acres	acres	acres = 404.678363 (000 ha)	area	2471.09823	

cropdescriptor Contains informations about specific crops.

schema	Column	Description
	id	the identifier for the crop
	label	the label to display for this crop
	seasons	the season(s) of this crop
	prod_default_unit	identify the default unit of measure for production data
	area_default_unit	identify the default unit of measure for area
	yield_default_unit	identify the default unit of measure for yield

Note: The default unit of measure is the expected unit of measure in the input csv, and it is also the unit of measure selected on the UI startup.

Sample Content here some sample rows from the *cropdescriptor* table

id	label	seasons	prod_default_unit	area_default_unit	yield_default_unit
wheat	Wheat	RABI	000_tons	000_ha	kg_ha
rice	Rice	KHARIF	000_tons	000_ha	kg_ha
maize	Maize	KHARIF	000_tons	000_ha	kg_ha
soybean	Soybean	KHARIF	000_tons	000_ha	kg_ha
sugarcane	Sugarcane	KHARIF	000_tons	000_ha	kg_ha
cotton	Cotton	KHARIF	000_bales	000_ha	kg_ha
chickpea	Chickpea	RABI	tons	000_ha	kg_ha
fodder	Fodder	RABI,KHARIF	tons	000_ha	kg_ha

cropdata Contains the data about production, cultivated area and yield of every year. This table can be populated ingesting the csv files.

Sample Content here some sample rows from the **cropdata** table

crop	district	province	year	years	area	production	yield	src
rice	Bahawalnagar	PUNJAB	2010	2010-11	63.13	127.35	2017	Official
rice	Bahawalpur	PUNJAB	2010	2010-11	6.07	10.34	1703	Official
rice	Bhakkar	PUNJAB	2010	2010-11	1.21	1.83	1512	Official
rice	Chiniot	PUNJAB	2010	2010-11	32.37	63.55	1963	Official

The crop column have to be the same of the id in **cropdescriptor** table. *district* and *province* columns must match with district and province fields in the *_**crop** and *_**crop_map** tables. Yield column is redondant for compatibility reasons. the yield is calculated at runtime getting values from *area* and *production* fields.

Note:

Unit of measure of data is now uniform for all the data. CSV data are converted at ingestion time in the database format. During the update the previous data (cotton production data) will be converted from the original format (000 bales) into the db format (000 tons)

- **production** is expressed in thousands of tons.
- **area** is always expressed in thousands of hectares
- **yield** is always expressed in kg/ha.

agrometdescriptor The agrometdescriptor is a table that contains the list of the available factors.

schema	Column	Description
	factor	this is the string that identify the factor.
	label	this is the label that represents the factor in the application
	aggregation	this is the kind of aggregation method to apply. can be avg (average) or sum.
	unit	the unit of measure for the factor

Sample Content here some sample rows from the **agrometdescriptor** table

factor	label	aggregation	unit
Tmax_avg	Max Temperature	avg	°C
NDVI_avg	NDVI	avg	
ppt_sum_mm	Precipitation	avg	mm
Daylen_avg	Day length	avg	hr
Tmin_avg	Min Temperature	avg	°C

agromet Contains agro-metereological data for each district of Pakistan for each *dekad*.

Column	Description
district	the district for this value
province	the province for this value
year	the year for this value
month	the month for this value
dec	the <i>dekad</i> for this value
factor	the factor for this value
value	the factor for this value
s_yr	the year of the start of the Rabi Season
s_dec	the <i>dekad</i> starting from the start of the Rabi Season

Sample Content here some sample rows from the *agromet* table

district	province	year	month	dec	factor	value	s_yr	s_dec	abso- lute_dek	dek_in_year
Malakand	KPK	2006	Oct	2	Tmin_avg	10.9563	2012	35	72238	22
Mansehra	KPK	2006	Oct	2	Tmin_avg	10.999	2012	35	72238	22
Nowshera	KPK	2006	Oct	2	Tmin_avg	11.3808	2012	35	72238	22
Pe-shawar	KPK	2006	Oct	2	Tmin_avg	9.0622	2012	35	72238	22
Mardan	KPK	2006	Oct	2	Tmin_avg	11.3221	2012	35	72238	22
Shangla	KPK	2006	Oct	2	Tmin_avg	11.4006	2012	35	72238	22
Swabi	KPK	2006	Oct	2	Tmin_avg	12.7756	2012	35	72238	22
Swat	KPK	2006	Oct	2	Tmin_avg	10.5381	2012	35	72238	22
Tank	KPK	2006	Oct	2	Tmin_avg	16.1516	2012	35	72238	22

Note:

- The *factor* column have to be the same of the id in **agrometdescriptor** table.
- *district* and *province* columns must match with *district* and *province* columns in the **district_boundary** and **province_boundary** tables.
- s_dec, s_yr and dek_in_year (*dekad in year*) are redondant field used to simplify queries to the database and make them faster.

crop status Contains limits and optimal values of agro-metereological values for each crop and each *dekad* in year.

Column	Description
factor	the agro-metereological variable.
crop	the crop to apply this limit
month	the month
dec	the <i>dekad</i>
max	the upper bound of the value
min	the lower bound of the value
opt	the optimal value

Sample Content here some sample rows from the *cropstatus* table

factor	crop	month	dec	s_dec	max	min	opt
Tmax_avg	wheat	Jan	1	7	36	3	20
Tmax_avg	wheat	Jan	2	8	37	4	21
Tmax_avg	fake_crop_2	Apr	1	16	16	42	15
Tmax_avg	fake_crop_2	Apr	2	17	17	42	-5
Tmax_avg	fake_crop_2	Apr	3	18	18	41	17
Tmax_avg	fake_crop_2	Dec	1	4	4	42	16
Tmax_avg	fake_crop_2	Dec	2	5	5	41	12
Tmax_avg	fake_crop_2	Dec	3	6	6	40	1
Tmax_avg	fake_crop_2	Feb	1	10	10	30	21
Tmax_avg	fake_crop_2	Feb	2	11	11	30	22

fertilizers Contains fertilizers data for each district of Pakistan for each month.

Column	Description
province	the province
district	the district for data
year	the year
month	the month (e.g. Jun)
month_num	the month number (1-12)
nutrient	the fertilizer name
offtake_tons	the quantity (tons)

Note: the district is and province are optional. If the district is empty, the data is related to the whole province. If both province and district are empty, the data is for the whole Pakistan.

Sample Content here some sample rows from the *fertilizers* table

id	district	month	month_num	nutrient	off-take_tons	province	year
3	Charsadda	Jan	1	Nitrogen	653.63	KHYBER PAKHTUNKHWA	2011
4	Dera Ismail Khan	Jan	1	Nitrogen	47.08	KHYBER PAKHTUNKHWA	2011
6	Lakki Marwat	Jan	1	Nitrogen	259.04	KHYBER PAKHTUNKHWA	2011
7	Malakand	Jan	1	Nitrogen	43.41	KHYBER PAKHTUNKHWA	2011
8	Mardan	Jan	1	Nitrogen	2829.47	KHYBER PAKHTUNKHWA	2011
9	Nowshera	Jan	1	Nitrogen	875.56	KHYBER PAKHTUNKHWA	2011
10	Peshawar	Jan	1	Nitrogen	2438.76	KHYBER PAKHTUNKHWA	2011
11	Swabi	Jan	1	Nitrogen	294.57	KHYBER PAKHTUNKHWA	2011
12	Attock	Jan	1	Nitrogen	127.03	PUNJAB	2011
14	Bahawalpur	Jan	1	Nitrogen	5963.26	PUNJAB	2011

market_price Contains market prices data for each district of Pakistan for each *dekad*.

Column	Description
crop	the commodity for the price
decade	the <i>dekad</i> .
decade_absolute	<i>absolute dekad</i>
decade_year	the dec in year
month	the mount of the price observation
year	the year of the price observation
province	the province where the price has been observed
district	the district where the price has been observed
market_price_kpr	the market price in rupies
market_price_usd	the market price in US dollars

Note: the market price

Sample Content here some sample rows from the *market_price* table

crop	decade	decade_absolute	decade_year	district	market_price_kpk	market_price_usd	month	province	year
wheat	1	72433	1	Islamabad	65	0.638885	Jan	PUNJAB	2012
rice irri-6	1	72433	1	Islamabad	125	1.228625	Jan	PUNJAB	2012
rice bas-mati	1	72433	1	Islamabad	160	1.57264	Jan	PUNJAB	2012
wheat	1	72469	1	Peshawar	3480	34.20492	Jan	KHYBER PAKHTUNKHWA	2013
wheat	1	72469	1	Bahawalpur	3175	31.207075	Jan	PUNJAB	2013
wheat	1	72469	1	Faisalabad	3330	32.73057	Jan	PUNJAB	2013
wheat	1	72469	1	Gujranwala	3340	32.82886	Jan	PUNJAB	2013
wheat	1	72469	1	Lahore	3425	33.664325	Jan	PUNJAB	2013
wheat	1	72469	1	Multan	3280	32.23912	Jan	PUNJAB	2013
wheat	1	72469	1	Islamabad	82.5	0.8108925	Jan	PUNJAB	2013

waterflow Contains waterflow data about river water inflow at rim stations

Column	Description
decade	the <i>dek</i> of the observation
decade_absolute	the <i>absolute dekad</i> of the observation
decade_year	the <i>dekad in year</i> of the observation
month	the month of the observation
river	the river
waterflow	the value
year	the year of the observation

Sample Content here some sample rows from the *waterflow* table

decade	decade_absolute	decade_year	month	river	waterflow	year
1	72469	1	Jan	Indus River at Chashma	21.22	2013
2	72470	2	Jan	Indus River at Chashma	33.75	2013
3	72471	3	Jan	Indus River at Chashma	46.28	2013
1	72472	4	Feb	Indus River at Chashma	59.89	2013
2	72473	5	Feb	Indus River at Chashma	58.81	2013
3	72474	6	Feb	Indus River at Chashma	57.72	2013
1	72475	7	Mar	Indus River at Chashma	52.54	2013
2	72476	8	Mar	Indus River at Chashma	56.63	2013
3	72477	9	Mar	Indus River at Chashma	60.72	2013
1	72478	10	Apr	Indus River at Chashma	42.02	2013

withdrawal Contains data about irrigation water supply for each district

Column	Description
decade	the <i>dekad</i> of the observation
decade_absolute	the <i>absolute dekad</i> of the observation
decade_year	the <i>dekad in year</i> of the observation
month	the month of the observation
river	the river
waterflow	the value
year	the year of the observation

Sample Content here some sample rows from the *waterflow* table

decade	decade_absolute	decade_year	month	province	dis- trict	with- drawal	year
1	72508	4	Feb	KHYBER PAKHTUNKHWA		0	2014
1	72544	4	Feb	KHYBER PAKHTUNKHWA		0	2015
1	72388	28	Oct	SINDH		1.2777	2010
1	72388	28	Oct	KHYBER PAKHTUNKHWA		0.007	2010
1	72388	28	Oct	BALUCHISTAN		0.0313	2010
2	72389	29	Oct	PUNJAB		1.6847	2010
1	72397	1	Jan	KHYBER PAKHTUNKHWA		0.0193	2011
1	72397	1	Jan	BALUCHISTAN		0.0697	2011
2	72398	2	Jan	PUNJAB		0.3113	2011

Note: the district is and province are optional. If the district is empty, the data is related to the whole province. If both province and district are empty, the data is for the whole Pakistan.

4.6 Configuration

This section shows how to modify configuration files and apply some simple changes to the GUI.

4.6.1 MapStore Configuration and Customization

This section shows how to modify the MapStore configuration and apply some simple changes to the GUI.

MapStore Configuration

MapStore Configuration File

MapStore is the web application that provides the base view.

You can find the MapStore configuration file at:

```
WEB-INF/app/static/config/mapStoreConfig.js
```

This is a *JSON* file that implement the configuration of the viewer.

When you log-in into the application as an administrator, you will be able to open the Administration Page. This is configured in another file called `managerConfig.js`:

```
WEB-INF/app/static/config/managerConfig.js
```

The objects in these two configurations are merged with the `localConfig.js` file that contain the options common to the map view and the administration page.:

```
WEB-INF/app/static/config/common/localConfig.js
```

common configuration options the first elements of the configuration file are common for every version of MapStore:

```
{
  "geoStoreBase": "/geostore/rest/",
  "proxy": "/http_proxy/proxy/?url=",
  "defaultLanguage": "en",
  "tab": true,
  "gsSources": {
    "nrl": {
      "ptype": "gxp_wmssource",
      "title": "NRL GeoServer",
      "projection": "EPSG:900913",
      "url": "/geoserver/ows",
      "layersCachedExtent": [
        5009377.085000001,
        0.0,
        10018754.169999998,
        5009377.085000001
      ],
      "layerBaseParams": {
        "format": "image/png8",
        "TILED": true
      }
    },
    "mapquest": {
      "ptype": "gxp_mapquestsource"
    },
    "osm": {
      "ptype": "gxp_osmsource"
    }
  },
}
```

```
"google": {
  "ptype": "gxp_googlesource"
},
"ol": {
  "ptype": "gxp_olsource"
}
},
"proj4jsDefs": {
  "EPSG:32642": "+proj=utm +zone=42 +ellps=WGS84 +datum=WGS84 +units=m +no_defs"
},
"map": {
  "projection": "EPSG:900913",
  "units": "m",
  "center": [
    7798771.2914706,
    3574215.5268897
  ],
  "zoom": 5,
  "maxExtent": [
    -20037508.34,
    -20037508.34,
    20037508.34,
    20037508.34
  ],
}
```

The configuration contains:

- `geoStoreBase`: the url to geostore
- `proxy`: the proxy url
- `defaultLanguage`: the default sources
- `proj4jsDefs`: the definition for EPSG:32642 projection
- `map`: base map configurations:
 - `projection`: the projection of the map
 - `units`: the unit of measure of the projection
 - `center`: the initial center of the map
 - `zoom`: the initial zoom of the map
 - `maxExtent`: the max extent of the map

Layers configuration The `layers` array contains the list of the arrays available on the map:

```
"layers": [
```

The first 4 layers are not visible and not displayed in the Layer Switcher. These hidden layers are needed to provide selection operations, and should not be removed:

```
{
  "id": "Crop_Province",
  "source": "nrl",
  "title": "nrl:province_crop",
  "name": "nrl:province_crop",
  "displayInLayerSwitcher": false,
  "visibility": false
}
```

```

},
{
  "source": "nrl",
  "title": "nrl:district_crop",
  "name": "nrl:district_crop",
  "displayInLayerSwitcher": false,
  "visibility": false
},
{
  "source": "nrl",
  "title": "nrl:province_view",
  "name": "nrl:province_view",
  "displayInLayerSwitcher": false,
  "visibility": false
},
{
  "source": "nrl",
  "title": "nrl:district_view",
  "name": "nrl:district_view",
  "displayInLayerSwitcher": false,
  "visibility": false
},
{
  "source": "nrl",
  "title": "nrl:district_select",
  "name": "nrl:district_select",
  "displayInLayerSwitcher": false,
  "visibility": false
},
{
  "source": "nrl",
  "title": "nrl:province_select",
  "name": "nrl:province_select",
  "displayInLayerSwitcher": false,
  "visibility": false
},
},
Than the visible background layers....:
{
  "source": "mapquest",
  "title": "MapQuest OpenStreetMap",
  "name": "osm",
  "group": "background",
  "visibility": true
},
{
  "source": "osm",
  "title": "Open Street Map",
  "name": "mapnik",
  "group": "background",
  "visibility": false
},
{
  "source": "bing",
  "title": "Bing Aerial",
  "name": "Aerial",
  "group": "background",
  "visibility": false
}

```

```
    },
    {
      "source": "bing",
      "title": "Bing Aerial With Labels",
      "name": "AerialWithLabels",
      "group": "background",
      "visibility": false
    },
    {
      "source": "google",
      "title": "Google Terrain",
      "name": "TERRAIN",
      "group": "background",
      "visibility": false
    },
    {
      "source": "google",
      "title": "Google Hybrid",
      "name": "HYBRID",
      "group": "background",
      "visibility": false
    },
    {
      "source": "google",
      "title": "Google Roadmap",
      "name": "ROADMAP",
      "group": "background",
      "visibility": false
    },
    {
      "source": "nrl",
      "title": "Administrative",
      "name": "nrl:g0gen_pak",
      "format": "image/jpeg",
      "group": "background",
      "visibility": true,
      "layersCachedExtent": [
        -20037508.34,
        -20037508.34,
        20037508.34,
        20037508.34
      ]
    },
    {
      "source": "ol",
      "group": "background",
      "fixed": true,
      "type": "OpenLayers.Layer",
      "visibility": false,
      "args": [
        "None",
        {
          "visibility": false
        }
      ]
    }
  ],
}
```

...and layers inside the other groups:

```

{
  "source": "nrl",
  "title": "Province Boundary",
  "name": "nrl:province_view",
  "group": "Admin",
  "visibility": true
},
{
  "source": "nrl",
  "title": "Flooded Areas 2012",
  "name": "nrl:flood_pak_2012",
  "group": "Flooding",
  "visibility": false
},
{
  "source": "nrl",
  "title": "Flooded Areas 2011",
  "name": "nrl:flood_pak_2011",
  "group": "Flooding",
  "visibility": false
},
{
  "source": "nrl",
  "title": "Flooded Areas 2010",
  "name": "nrl:flood_pak_2010",
  "group": "Flooding",
  "visibility": false
},
{
  "source": "nrl",
  "title": "Contours 1000ft",
  "name": "nrl:ETOPO2v2c_1000ft_conts_ln_pak",
  "group": "Topography",
  "visibility": false
},
{
  "source": "nrl",
  "title": "Crop Mask",
  "name": "nrl:crop_mask_pak_2012",
  "group": "Land Cover",
  "visibility": false
},
{
  "source": "nrl",
  "title": "Land cover 2010",
  "name": "nrl:LULC2010_Pak",
  "group": "Land Cover",
  "visibility": false
},
{
  "source": "nrl",
  "title": "Land cover 2000",
  "name": "nrl:LULC2000_Pak_wgs84",
  "group": "Land Cover",
  "visibility": false
},
{
  "source": "nrl",

```

```
    "title": "GlobCover 2005-06",
    "name": "nrl:GLOBC2006_v2.2",
    "group": "Land Cover",
    "visibility": false
  },
  {
    "source": "nrl",
    "title": "GlobCover 2009",
    "name": "nrl:GLOBC2009_v2.3",
    "group": "Land Cover",
    "visibility": false
  },
  {
    "source": "nrl",
    "title": "Rivers",
    "name": "nrl:rivers_pak",
    "group": "Hydrology",
    "visibility": false
  },
  {
    "source": "nrl",
    "title": "Indus River",
    "name": "nrl:indus_river_course",
    "group": "Hydrology",
    "visibility": true
  },
  {
    "source": "nrl",
    "title": "Roads",
    "name": "nrl:roads_pak",
    "group": "Transportation",
    "visibility": false
  },
  {
    "source": "nrl",
    "title": "District Boundary",
    "name": "nrl:district_view",
    "group": "Admin",
    "visibility": true
  },
  {
    "source": "nrl",
    "title": "MeteoData",
    "name": "nrl:met_stations",
    "group": "Meteo Stations",
    "visibility": false
  },
  {
    "source": "nrl",
    "title": "Populated Places",
    "buffer": "5",
    "name": "nrl:POP_settlements_pak_main",
    "group": "Admin",
    "visibility": true
  },
  {
    "source": "nrl",
    "title": "Label",
```

```

    "name": "nrl:administrative_labels",
    "group": "Admin",
    "visibility": true,
    "layersCachedExtent": [
      -20037508.34,
      -20037508.34,
      20037508.34,
      20037508.34
    ]
  }
}

```

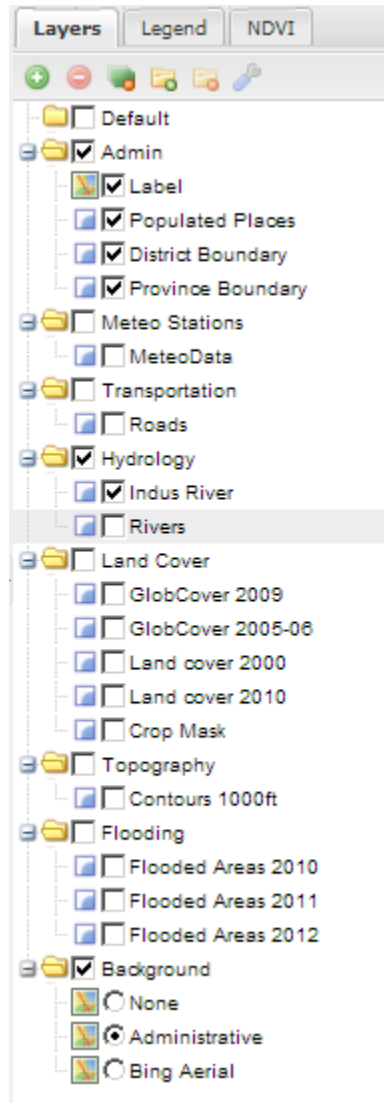


Fig. 4.34: the layers added to the map.

(close the layers array and map object):

```

]
},

```

Plugins configurations The `customTools` array contains the configuration of the additional MapStore plugins developed for this application and the customizations for some of the default ones:

```
"customTools": [
```

Portal Page The “Static Page” plugin in MapStore allow to use a page to show in the tab. This plugin is used to provide the “Portal” tab of the application.:


```
{
  "id": "gxp_portal_staticpage",
  "ptype": "gxp_staticpage",
  "url": "http://dwms.fao.org/~test/croportal/home_en.asp",
  "tabPosition": 0,
  "tabTitle": "Portal",
  "forceMultiple": true
},
```

- **id** : the identifier
- **url** : url of the page to include
- **tabPosition**: the position of the tab (0 is the first position)
- **tabTitle**: the title of the tab
- **forceMultiple**: this is a required parameter for the MapStore plugins. By default the configurations in custom-Tools are unique by type (`gxp_staticpage`) and a configuration override the existing ones. `forceMultiple` allow to add more than one plugin of this type. In this case we need also a static page for GeoNetwork.

GeoNetwork Another static page has been provided for the integrated GeoNetwork tab.

```
{
  "ptype": "gxp_staticpage",
  "url": "/geonetwork",
  "tabPosition": 10,
  "tabTitle": "Geonetwork",
  "forceMultiple": true
},
```

As you see, you have the same parameters for the same plugin type.

Zoom plugin The `gxp_zoomtoextent` plugin () is customized to zoom exactly to the extent provided in the extent array.:

```
{
  "ptype": "gxp_zoomtoextent",
  "extent": [
    6770799.251963,
    2705604.806669,
    8826743.330978,
    4442826.247111
  ],
  "actionTarget": {
    "target": "paneltbar",
    "index": 2
  }
},
```


Print SnapShot This plugin is available for chome browsers, and allow to do a snapshot of the current view of the map.:

```
{
  "ptype": "gxp_printsnapshot",
  "service": "/servicebox/",
  "customParams": {
    "outputFilename": "mapstore-print"
  },
  "actionTarget": {
    "target": "paneltbar",
    "index": 3
  },
  "disabledIn": ["internetExplorer", "Firefox"]
},
```

Print plugin The **Print plugin** is not one of the default plugins for *MapStore*, so it have to be explicitly added:

```
{
  "ptype": "gxp_print",
  "customParams": {
    "outputFilename": "mapstore-print"
  },
  "printService": "http://84.33.2.75/geoserver/pdf/",
  "legendPanelId": "legendPanel",
  "ignoreLayers": ["WFSSearch", "Marker"],
  "appendLegendOptions": true,
  "actionTarget": {
    "target": "paneltbar",
    "index": 4
  }
},
```

GeoLocation Menu A menu with GeoLocation tools (References Tool, *GeoCoder*, Address):

```
{
  "ptype": "gxp_geolocationmenu",
  "actionTarget": {
    "target": "paneltbar",
    "index": 16
  }
},
```

See also the “References Tool” section at the end of this chapter for details about how to configure it.

NDVI plugin The following plugins are the custom plugins’ configurations for the Crop Information Portal.

The NDVI plugin is a plugin that allow to add a layer on the map choosing the year, month and dekad. The layer have to accept temporal requests:

```
{
  "ptype": "gxp_ndvi",
  "dataUrl": "http://84.33.2.75/geoserver/ows",
  "layer": "ndvi:ndvi",
  "outputConfig": {
    "title": "NDVI",
    "id": "ndvi",
  }
},
```

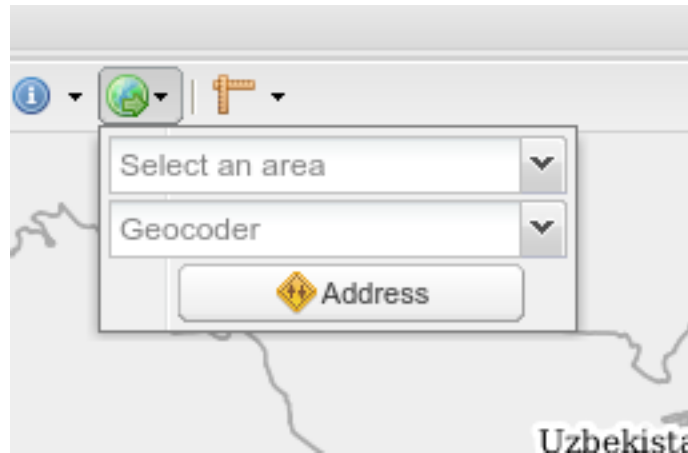


Fig. 4.35: the geolocation menu

```

        "region": "east",
        "replace": "false"
    },
    "outputTarget": "east"
},

```

Configuration Options:

- **dataURL:** the url of the WMS service
- **layer:** the layer to add

NRL plugin the **NRL plugin** is simply a container for the panels on the left of the MapStore template:

```

{
    "ptype": "gxp_nrl",
    "outputConfig": {
        "id": "nrl",
        "region": "east",
        "startTab": "nrlCropData"
    },
    "outputTarget": "west"
},

```

Crop Data plugin The **Crop Data** plugin implements the *Crop Data* tab functionalities:

```

{
    "ptype": "nrl_crop_data",
    "id": "CropData",
    "mapToolPosition": 18,
    "layerStyle": {
        "strokeColor": "red",
        "strokeWidth": 1,
        "fillOpacity": 0.2,
        "cursor": "pointer"
    },
    "dataUrl": "/geoserver/ows",
    "rangesUrl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:cropland"
}

```

```

"unitsUrl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:measure",
"highChartExportUrl": "/highcharts-export/",
"sourcesUrl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:crop",
"layers": {
  "district": "nrl:district_crop",
  "province": "nrl:province_view"
},
"outputConfig": {
  "itemId": "nrlCropData"
},
"outputTarget": "nrl",
"portalRef": "gxp_portal_staticpage",
"helpPath": "/~test/croportal/hlp_cropdata_en.asp"
},

```

Configuration Options:

- **dataUrl**: the url of the WMS service
- **layerStyle**: the style for the highlighting of the selected areas
- **rangesUrl**: the layer that expose the available crops and the yearly ranges of available data for each crop.
- **unitsUrl**: the layer that expose the available units of measure.
- **sourcesUrl**: the layer that expose the available units of sources (a list of the ones on the database)
- **highChartExportUrl**: the URL of the service that allows the chart exporting.
- **areaFilter**: a filter to apply to the layers. Is used to filter the selectable areas.
- **layers** the layers for the selection tools (see *Layers configuration*) * **district** the layer to query when the Area of interest type is *district* * **province** the layer to query when the Area of interest type is *province*
- **portalRef**: the id of the plugin to use for help
- **helpPath**: the path to the help for this tool (from the original)

Argomet plugin The **Argomet** plugin implements the *Agromet Variables* tab functionalities:

```

{
  "ptype": "nrl_agromet",
  "layerStyle": {"strokeColor": "green", "strokeWidth": 1, "fillOpacity": 0.2, "cursor": "pointer"},
  "dataUrl": "http://84.33.2.75/geoserver/ows",
  "factorsurl": "http://84.33.2.75/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:factor",
  "highChartExportUrl": "http://84.33.2.75/highcharts-export/",
  "areaFilter": "province NOT IN ('DISPUTED TERRITORY', 'DISPUTED AREA')",
  "titleText": "Agromet Variables",
  "outputConfig": {
    "id": "Agromet"
  },
  "outputTarget": "nrl"
},

```

Configuration Options:

- **dataUrl**: the url of the WMS service
- **layerStyle**: the style for the highlighting of the selected areas
- **factorsurl**: the layer that expose the available factors and the yearly ranges of available data for each factor.
- **areaFilter**: a filter to apply to the layers. Is used to filter the selectable areas.

- `highChartExportUrl`: the URL of the service that allows the chart exporting.
- `layers`: the layers for the selection tools (see [Layers configuration](#)) * `district`: the layer to query when the Area of interest type is *district* * `province`: the layer to query when the Area of interest type is *province*
- `portalRef`: the id of the plugin to use for help
- `helpPath`: the path to the help for this tool (from the original)

Fertilizers The **Fertilizers** plugin implements the *Fertilizers* tab functionalities:

```
{
  "ptype": "nrl_fertilizers",
  "layerStyle": {
    "strokeColor": "purple",
    "strokeWidth": 1,
    "fillOpacity": 0.2,
    "cursor": "pointer"
  },
  "typeNameData": "nrl:fertilizer_data",
  "dataUrl": "/geoserver/ows",
  "factorsurl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:agron",
  "metadataUrl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:fert",
  "highChartExportUrl": "/highcharts-export/",
  "titleText": "Fertilizers",
  "outputConfig": {
    "id": "Fertilizers"
  },
  "outputTarget": "nrl",
  "layers": {
    "district": "nrl:district_crop",
    "province": "nrl:province_view"
  },
  "portalRef": "gxp_portal_staticpage",
  "helpPath": "/~test/croportal/hlp_fertilizer_en.asp"
},
```

Configuration Options:

- `dataUrl`: the url of the WMS service
- `layerStyle`: the style for the highlighting of the selected areas
- `metadataUrl`: the layer that exposes the metadata for the fertilizer plugin (fertilizers and data limits).
- `areaFilter`: a filter to apply to the layers. Is used to filter the selectable areas.
- `highChartExportUrl`: the URL of the service that allows the chart exporting.
- `layers`: the layers for the selection tools (see [Layers configuration](#)) * `district`: the layer to query when the Area of interest type is *district* * `province`: the layer to query when the Area of interest type is *province*
- `portalRef`: the id of the plugin to use for help
- `helpPath`: the path to the help for this tool (from the original)

Irrigation The **Irrigation** plugin implements the *Irrigation* tab functionalities:

```
{
  "ptype": "nrl_irrigation",
  "layerStyle": {
```

```

        "strokeColor": "aqua",
        "strokeWidth": 1,
        "fillOpacity": 0.3,
        "cursor": "pointer"
    },
    "dataUrl": "/geoserver/ows",
    "factorsurl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:mea",
    "defaultUOMFlow": "1",
    "defaultUOMSupply": "1",
    "metadataFlowUrl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=n",
    "metadataSupplyUrl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=",
    "highChartExportUrl": "/highcharts-export/",
    "titleText": "Water Resources",
    "outputConfig": {
        "id": "Irrigation"
    },
    "outputTarget": "nrl",
    "layers": {
        "province": "nrl:province_view",
        "district": "nrl:district_crop"
    },
    "areaFilter": "province NOT IN ('DISPUTED TERRITORY', 'DISPUTED AREA')",
    "portalRef": "gxp_portal_staticpage",
    "helpPath": "/~test/croportal/hlp_water_en.asp"
},

```

Configuration Options:

- **dataUrl**: the url of the WMS service
- **layerStyle**: the style for the highlighting of the selected areas
- **factorsurl**: the layer that expose the units of measure for the waterflow data.
- **defaultUOMFlow**: The value of the factor of the default unit of measure for waterflow data.
- **defaultUOMSupply**: The value of the factor of the default unit of measure for water supply data.
- **metadataFlowUrl**: the layer that expose the metadata for the waterflow data (rivers and data limits).
- **metadataSupplyUrl**: the layer that expose the metadata for the water supply data (data limits).
- **areaFilter**: a filter to apply to the layers. Is used to filter the selectable areas.
- **highChartExportUrl**: the URL of the service that allows the chart exporting.
- **layers**: the layers for the selection tools (see [Layers configuration](#)) * **district**: the layer to query when the Area of interest type is *district* * **province**: the layer to query when the Area of interest type is *province*
- **portalRef**: the id of the plugin to use for help
- **helpPath**: the path to the help for this tool (from the original)

Market Prices The **Market Prices** plugin implements the *Market Prices* tab functionalities:

```

{
    "ptype": "nrl_market_prices",
    "layerStyle": {
        "strokeColor": "orange",
        "strokeWidth": 1,
        "fillOpacity": 0.3,
        "cursor": "pointer"
    }
}

```

```
},
  "currencies": [
    [
      "usd",
      "US Dollars"
    ],
    [
      "pkr",
      "PK Rupees"
    ]
  ],
  "defaultCurrency": "pkr",
  "defaultDenominator": "0.4",
  "dataUrl": "/geoserver/ows",
  "factorsurl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:meas",
  "metadataUrl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:mar",
  "highChartExportUrl": "/highcharts-export/",
  "titleText": "Market Prices",
  "outputConfig": {
    "id": "Market Prices"
  },
  "outputTarget": "nrl",
  "layers": {
    "province": "nrl:province_view",
    "district": "nrl:district_crop"
  },
  "areaFilter": "province NOT IN ('DISPUTED TERRITORY', 'DISPUTED AREA')",
  "portalRef": "gxp_portal_staticpage",
  "helpPath": "/~test/croportal/hlp_market_en.asp"
},
```

Configuration Options:

- dataUrl: the url of the WMS service
- layerStyle: the style for the highlighting of the selected areas
- currencies: The currencies available.
- defaultCurrency: default currency
- defaultDenominator: the default denominator
- factorsurl: the layer that expose the units of measure for the currency data.
- metadataUrl: the layer that expose the metadata for the market prices data (commodities and data limits).
- areaFilter: a filter to apply to the layers. Is used to filter the selectable areas.
- highChartExportUrl: the URL of the service that allows the chart exporting.
- layers: the layers for the selection tools (see [Layers configuration](#)) * district: the layer to query when the Area of interest type is *district* * province: the layer to query when the Area of interest type is *province*
- portalRef: the id of the plugin to use for help
- helpPath: the path to the help for this tool (from the original)

Crop Status The **Crop Status** plugin implements the *Agromet Variables* tab functionalities:

```
{
  "ptype": "nrl_crop_status",
  "layerStyle": {
    "strokeColor": "blue",
    "strokeWidth": 1,
    "fillOpacity": 0.2,
    "cursor": "pointer"
  },
  "factorsurl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:agron",
  "rangesurl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:cropda",
  "dataUrl": "/geoserver/ows",
  "highChartExportUrl": "/highcharts-export/",
  "outputConfig": {
    "id": "nrlCropStatus"
  },
  "outputTarget": "nrl",
  "portalRef": "gxp_portal_staticpage",
  "helpPath": "/~test/croportal/hlp_status_en.asp"
},
```

Configuration Options:

- **dataUrl**: the url of the WMS service
- **layerStyle**: the style for the highlighting of the selected areas
- **factorsurl**: the layer that expose the available factors and the yearly ranges of available data for each factor.
- **rangesurl**: the layer that expose the available crops and the yearly ranges of available data for each crop.
- **highChartExportUrl**: the URL of the service that allows the chart exporting.
- **layers** the layers for the selection tools (see [Layers configuration](#)) * **district** the layer to query when the Area of interest type is *district* * **province** the layer to query when the Area of interest type is *province*

Crop Report Plugin The **Crop Report Plugin** plugin implements the *Crop Report* tab functionalities:

```
{
  "ptype": "nrl_report_crop_data",
  "cropPluginRef": "CropData",
  "factorsurl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:agron",
  "rangesurl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:cropda",
  "unitsUrl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:measure",
  "dataUrl": "/geoserver/ows",
  "highChartExportUrl": "/highcharts-export/",
  "outputConfig": {
    "id": "nrlReportCropData"
  },
  "layers": {
    "district": "nrl:district_crop",
    "province": "nrl:province_view"
  },
  "targetLayerStyle": {
    "strokeColor": "green",
    "strokeWidth": 2,
    "fillOpacity": 0
  },
  "defaultAreaTypeMap": "district",
  "disclaimerText": "Disclaimer: Data, information and products in this report are provided \"as is\"",
  "areaFilter": "province NOT IN ('GILGIT BALTISTAN', 'AJK', 'DISPUTED TERRITORY', 'DISPUTED AREA')",
```

```
"outputTarget": "nrl",
"portalRef": "gxp_portal_staticpage",
"helpPath": "/~test/croportal/hlp_report_en.asp"
},
```

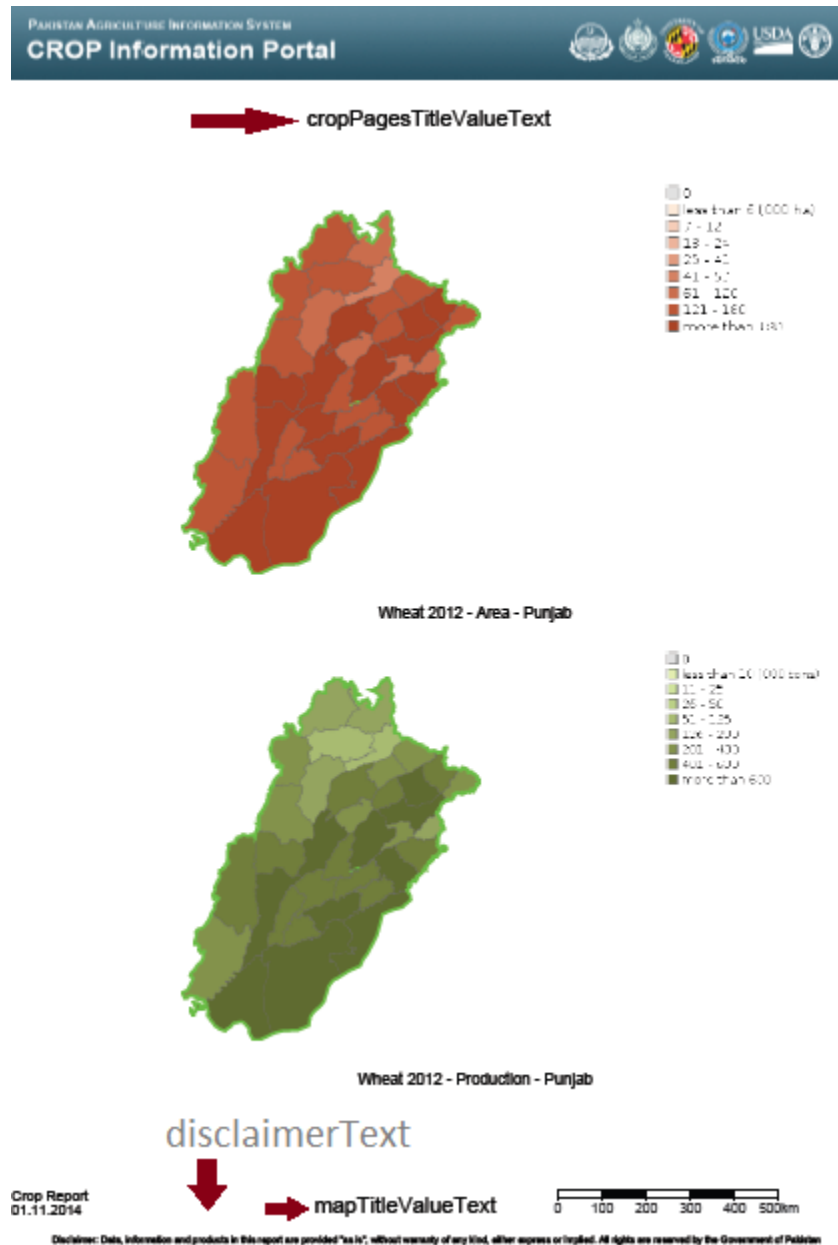


Fig. 4.36: A Page from the PDF report.

Configuration Options:

- `dataUrl`: the url of the WMS service
- `factorsurl`: the layer that expose the available factors and the yearly ranges of available data for each factor.
- `rangesUrl`: the layer that expose the available crops and the yearly ranges of available data for each crop.

- `highChartExportUrl`: the URL of the service that allows the chart exporting.
- **layers**: the layers for the selection tools (see *Layers configuration*)
 - `district`: the layer to query when the Area of interest type is *district*
 - `province`: the layer to query when the Area of interest type is *province*
- `targetLayerStyle`: the style of the highlighted regions in the generated maps inside the pdf.
- `defaultAreaTypeMap`: the default type of the map if not specified.
- `disclaimerText`: the text to add to the disclaimer of the pdf.

The **Crop Report Plugin** uses also another plugin called *gxp_printreportheper* to provide some customization options:

```
{
  "ptype": "gxp_printreportheper",
  "printService": "/geoserver/pdf/",
  "dataUrl": "/geoserver/ows",
  "defaultExtent": [
    6770799.251963,
    2705604.806669,
    8826743.330978,
    4442826.247111
  ],
  "id": "printreportheper",
  "hideAll": true,
  "mapTitleValueText": "Crop Report",
  "cropPagesTitleValueText": "Crop Maps and Charts",
  "meteorologicalPagesTitleValueText": "AgroMet Variables"
},
```

Configuration Options:

- `printService`: the url of the print service
- `dataUrl`: the url of the WFS service
- `defaultExtent`: the default extent of the maps in the generated pdf.
- `hideAll`: Don't show report window and hide layers if true.
- `mapTitleValueText` the title of the document to print.
- `cropPagesTitleValueText` the title of pages that contain crop data.
- `meteorologicalPagesTitleValueText` the title of pages that contain charts of argrometereological factors.

Login Plugin Adds the Login button to the toolbar.:

```
{
  "actions": [
    "->"
  ],
  "actionTarget": "paneltbar"
},
{
  "ptype": "gxp_geostore_login",
  "loginService": "/geostore/rest/users/user/details/",
  "enableAdminGUILogin": true,
```

```
"renderAdminToTab": true,
"autoLogin": true,
"adminGUIUrl": "/manager",
"adminGUIHome": "",
"adminLoginInvalidResponseValidator": "No AuthenticationProvider found",
"isDummy": false,
"actionTarget": "paneltbar"
},
```

Note: "actions": ["->"] is a separator added to the toolbar to align the login button on the right

Configuration Options:

- loginService: the service to call
- enableAdminGUILogin: if true, do the login on the Administration Page too.
- adminGUIUrl: URL of the admin application to link in the menu.
- adminGUIHome: path where to go after the login to the Administrator GUI.
- adminLoginInvalidResponseValidator: Error text if the service is not available.

Disclaimer The disclaimer is a custom plugin that implement the button in the bottom bar with the link to the disclaimer.:

```
{
  "ptype": "gxp_disclaimer",
  "id": "disclaimer",
  "actionTarget": "appTabs.bbar",
  "text": "Disclaimer",
  "tooltip": "Open the Disclaimer",
  "index": 26,
  "showOnStartup": false,
  "iconCls": "icon-about",
  "portalRef": "gxp_portal_staticpage",
  "disclaimerPath": "/~test/croportal/doc_cip_disclaimer_en.asp"
}
```

Configuration Options:

- text: the text of the disclaimer
- tooltip: The tooltip of the disclaimer
- iconCls: The icon of the button (css class)
- portalRef: the id of the plugin to use for help
- disclaimerPath: Error text if the service is not available.



Fig. 4.37: The disclaimer.

(then the close the customTools array and the main configuration object):

```
]
}
```

References Tool

The **References Tool** is the tool in the geolocation menu that allows to zoom to a particular region.

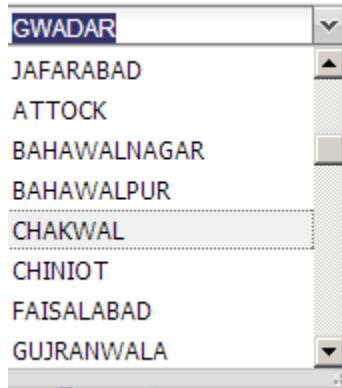


Fig. 4.38: references to areas to zoom.

The areas available for this tool are this file:

```
WEB-INF/app/static/data/georeferences.js
```

This JavaScript file instantiate the `georeferences_data` variable. The variable is an array of arrays with this form:

```
[ "AWARAN", //name of the area to display
  //bounding box in EPSG:4326
  "64.152076721191,
  25.4553985595703,
  66.267013549804,
  27.500825881958"]
```

The data provided contain a join of districts and regions:

```
var georeferences_data_districts = [ ["AWARAN" /*, ...*/ /*, ...*/];
var georeferences_data_provinces = [ [/*, ...*/ /*, ...*/];
var georeferences_data = georeferences_data_provinces.concat(georeferences_data_districts);
```

MapStore Customization

In this section we will learn how to simply customize the GUI of MapStore.

Customizing the Template

The main file that contains the application configuration is available at:

```
app/static/config/mapStoreConfig.js
```

Editing this file you can change the banner, for example, or add buttons to the bottom bar.

Change the Banner The default banner is configured in the file

```
app/static/config/common/localConfig.js
```

That is the base configuration common to all the services. You can override the header adding this to your mapStore-Config

```
[...]
"tab": true
"header": {
  "container": {
    "border": false,
    "header": false,
    "collapsible": true,
    "collapsed": false,
    "collapseMode": "mini",
    "hideCollapseTool": true,
    "split": true,
    "animCollapse": false,
    "minHeight": 90,
    "maxHeight": 90,
    "height": 109,
    "id": "header"
  },
  "html": [ "<div align=\"center\" style=\"background-image:url(theme/app/img/banner/bgimg-sindh.jpg)\">
    <img src=\"theme/app/img/banner/left-banner_en-sindh.jpg\" style=\"float:left\">
    <img src=\"theme/app/img/banner/right-banner_en-sindh.jpg\" style=\"float:right\">
  </div>\",
    <map name=\"IM_left-banner\">\",
      <area shape=\"rect\" coords=\"19,19,393,44\" href=\"http://dwms.fao.org/~test/h...
    </map>\",
    <map name=\"IM_right-banner\">\",
      <area shape=\"rect\" coords=\"330,23,382,83\" href=\"http://www.fao.org\" target=
      <area shape=\"rect\" coords=\"274,23,325,83\" href=\"http://www.usda.gov/wps/por
      <area shape=\"rect\" coords=\"205,23,268,83\" href=\"http://www.umd.edu/\" targ
      <area shape=\"rect\" coords=\"142,23,204,83\" href=\"http://www.suparco.gov.pk/
      <area shape=\"rect\" coords=\"85,23,141,83\" href=\"http://www.sindhagri.gov.pk/
    </map>\" ]
  ],
}
```

See `punjab.js` and `sindh.js` examples in the `app/static/config/` directory. You can test these configurations using the `config` parameter in the URL.

E.g. : <http://localhost:8888/MapStore/?config=sindh>

Adding Buttons and Links to the button bar `mapStoreConfig.js` file contains the `customTools` array that allows to add plugins to the application:

You can add this entry to the `json` file to add a button that opens a window:

```
{
  "ptype": "gxp_help",
  "actionTarget": "appTabs.bbar",
  "text": "Disclaimer",
  "tooltip": "About This Example",
  "title": "Crop Source Disclaimer",
  "index": 26,
  "showOnStartup": false,
  "iconCls": "icon-about",
}
```

```

"description": "<h1>SUPARCO Source</h1><br><p>The Pakistan Space and Upper Atmosphere Research Comm.
"windowOptions":{
  "constrain":true
}
}

```

An then reload the [MapStore](#) page. You can see the new button at the bottom right of the page.



Fig. 4.39: The new button.

An then reload the [MapStore](#) page.

Administration Page

The “Administration Page” interacts with various server side services to allow an administrator to manage the Crop Information Portal.

Title	Description	Status	StartDate	File	Task	Progress	
Create or update Layer	Updates or creates a layer from a given shap...	SUCCESS	2016-01-22 12:39:20.118	20160122105853---pak_NDVI...	Completed	100%	⏮ ⏭ ⏭ ⏮
CSV file ingestion	Ingest CSV files (cropdata and agromet)	SUCCESS	2016-01-18 18:58:21.002	20160118171806-ADD_cropstat...	Completed	100%	⏮ ⏭ ⏭ ⏮
NDVI file ingestion	Ingest geotiff file and add them to a NDVI mo...	SUCCESS	2016-01-18 14:30:10.336	20160118124956-ADD_mo_Dist...	Completed	100%	⏮ ⏭ ⏭ ⏮
NDVI stats generation	Generate a CSV file with a geotiff mask and a...	SUCCESS	2015-12-17 16:27:28.572	20151217144813-suparco_whe...	Completed	100%	⏮ ⏭ ⏭ ⏮
		SUCCESS	2015-12-17 16:26:47.324	20151217144732-suparco_suga...	Completed	100%	⏮ ⏭ ⏭ ⏮
		SUCCESS	2015-12-17 16:26:16.390	20151217144701-suparco_rice...	Completed	100%	⏮ ⏭ ⏭ ⏮
		SUCCESS	2015-12-17 16:25:42.886	20151217144627-suparco_cotto...	Completed	100%	⏮ ⏭ ⏭ ⏮

Fig. 4.40: The Administration Page

Just for information it interacts with:

- Geobatch (Flows)
- GeoStore (User Management, Flow archiviavtion)
- OpenSDI Manager 2 (File Manager, configuration of Crops, Agromet Variables, Units of measure)

Configuration File

When you log-in into the application as an administrator, you will be able to open the Administration Page. This is configured in a *JSON* file called `managerConfig.js`:

```
WEB-INF/app/static/config/managerConfig.js
```

The objects in theese to configurations are merged with the `localConfig.js` file that contain the options common to the map view and the administration page.:

```
WEB-INF/app/static/config/common/localConfig.js
```

common configuration options the `mapStoreConfig.js` contains some base configurations at the beginning. They are not not important for the crop information portal configuration

```
{
  "composerUrl": "",
  "socialUrl": "",
  "start": 0,
  "limit": 20,
  "msmTimeout": 30000,
  "header": {},
  "twitter": {
    "via": "geosolutions_it",
    "hashtags": ""
  },
  "mediaContent": "./externals/mapmanager/theme/media/",
  "ASSET": {
    "delete_icon": "./externals/mapmanager/theme/img/user_delete.png",
    "edit_icon": "./externals/mapmanager/theme/img/user_edit.png"
  },
  "locales": [
    [
      "en",
      "English"
    ],
    [
      "it",
      "Italiano"
    ],
    [
      "fr",
      "Français"
    ],
    [
      "de",
      "Deutsch"
    ],
    [
      "es",
      "Español"
    ]
  ],
}
```

After these base configurations there are 3 main sections:

- **tools**: the tools when a user is not logged in
- **adminTools**: the configurations of the tools when an administrator is logged in
- **loggedTools** the tools when a non administrator user is logged in

For the crop information portal only the administrator can access to the administration page, so the only important section is the **adminTools**

Workflows This is the tool that communicate with GeoBatch to ingest CSV files, shape files, NDVI images and generate statistics.

```
{
  "ptype": "mxp_geobatch_flows",
  "autoOpen": true,
  "closable": false,
  "geoBatchRestURL": "/geobatch/rest/",
  "skipFlowsNotInRunConfigs": true,
  "runConfigs": {
    [...]
  },
  "actionTarget": {
    "target": "north.tbar",
    "index": 1
  }
},
```

- **autoOpen:** open on startup
- **closable:** the tool (the tab) can not be closed
- **geoBatchRestURL:** URL of the GeoBatch ReST interface
- **skipFlowsNotInRunConfigs:** if true, the tool shows only the GeoBatch flows present in the **runConfigs** configuration
- **runConfigs:** contain the configurations of the GUI for each flow (i.e. CSV file ingestion)
- **actionTarget:** where to place the button of the tool (not important)

runConfigs contain the configurations for the GUI to run the GeoBatch Flows.

CSV file ingestion starts the ingestion of the CSV files for updates

```
"csvingestion": {
  "xtype": "geobatch_csv_ingestion_run_form",
  "baseDir": "/opt/admin_dir",
  "fileBrowserUrl": "mvc/fileManager/extJSbrowser",
  "fileRegex": "\\\\.csv$",
  "path": "/",
  "ingestionSources": [
    ["Cropdata", "Crop Data"],
    ["Cropstatus", "Crop Status"],
    ["Agromet", "Agromet Variables"],
    ["Fertilizer", "Fertilizer"],
    ["Market Prices", "Market Prices"],
    ["Water Resources", "Water Resources"]
  ],
  "propertiesLists": {
    "marketPrices": ["denominator", "exchangeRate"],
    "cropData": ["src"]
  },
  "factorUrl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:measur",
  "sourcesUrl": "/geoserver/nrl/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=nrl:crop",
  "mediaContent": "/externals/mapmanager/theme/media/"
},
```

NDVI file ingestion start the ingestion of the new NDVI image:

```
"ndviingestion": {
  "xtype": "geobatch_run_local_form",
  "baseDir": "/opt/admin_dir",
  "fileBrowserUrl": "mvc/fileManager/extJSbrowser",
```

```
"fileRegex": "\\\\.ti[f]{1,2}$",
"path": "/",
"mediaContent": "./externals/mapmanager/theme/media/"
},
```

NDVI stats generation start the generation of the CSV from the NDVI images

```
"ndvistats": {
  "xtype": "geobatch_run_form",
  "baseDir": "/opt/admin_dir",
  "fileBrowserUrl": "mvc/fileManager/extJSbrowser",
  "fileRegex": "\\\\.shp$",
  "path": "/",
  "decadConfig": {
    "dataUrl": "/geoserver/ndvi/ows",
    "layer": "ndvi:ndvi"
  },
  "mediaContent": "./externals/mapmanager/theme/media/"
},
```

Create or update Layer start the ingestion of a shape file create new vector layers or update the existing ones:

```
"createupdatelayer": {
  "xtype": "geobatch_run_local_form",
  "baseDir": "/opt/admin_dir",
  "fileBrowserUrl": "mvc/fileManager/extJSbrowser",
  "fileRegex": "\\\\.zip$",
  "path": "/",
  "mediaContent": "./externals/mapmanager/theme/media/"
}
```

User Manager This is the user manager configuration. This configuration is stanard in all the installations of Map-Store

```
{
  "ptype": "mxp_usermanager",
  "setActiveOnOutput": true,
  "addManageGroupsButton": false,
  "loginManager": "loginTool",
  "showOnStartup": true,
  "actionTarget": {
    "target": "north.tbar",
    "index": 2
  },
  "outputConfig": {
    "closable": true,
    "closeAction": "close",
    "autoWidth": true,
    "viewConfig": {
      "forceFit": true
    }
  }
},
```

File Browser This is the configuration of the File Manager tool. Also this is standard in all the installations of MapStore:


```
{
  "ptype": "mxp_filebrowser",
  "buttonText": "File Manager",
  "multitab": false,
  "actionTarget": {
    "target": "north.tbar",
    "index": 3
  },
  "mediaContent": "../externals/mapmanager/theme/media/"
},
```

Entities The entities tool for MapStore is a generic tool in MapStore that interact with a *ReST* API to perform *CRUD* operations on some entities on a server.

In the crop information portal it is used to configure crops, units of measures and agromet variables, interacting with custom services implemented server side.

The configuration is very big because it contains all the settings for the API, the forms and the tables available in the “Entities” tool.

```
{
  "ptype": "mxp_entity_manger",
  "notDuplicateOutputs": true,
  "actionTarget": {
    "target": "north.tbar",
    "index": 4
  },
  "entities": [
    {
      "api": {
        "destroy": {
          "method": "DELETE",
          "url": "mvc/cip/crops/{id}"
        },
        "dump": {
          "method": "GET",
          "url": "mvc/cip/crops/dump"
        },
        "restore": {
          "method": "POST",
          "url": "mvc/cip/crops/restore"
        }
      },
      "autoExpandColumn": "label",
      "autoload": true,
      "basePath": "mvc/cip/crops/",
      "canCreate": true,
      "canDelete": true,
      "canEdit": true,
      "columns": [
        {
          "header": "id",
          "mapping": "id",
          "name": "id"
        },
        {
          "header": "Label",
          "mapping": "label",

```

```
        "name": "label"
      },
      {
        "header": "Default Production Uom",
        "mapping": "prod_default_unit",
        "name": "prod_default_unit"
      },
      {
        "header": "Default Area Uom",
        "mapping": "area_default_unit",
        "name": "area_default_unit"
      },
      {
        "header": "Default Yield Uom",
        "mapping": "yield_default_unit",
        "name": "yield_default_unit"
      },
      {
        "falseText": "",
        "fixed": true,
        "header": "Rabi",
        "mapping": "rabi",
        "maxWidth": 100,
        "name": "rabi",
        "trueText": "Yes",
        "width": 100,
        "xtype": "booleancolumn"
      },
      {
        "falseText": "",
        "fixed": true,
        "header": "Kharif",
        "mapping": "kharif",
        "maxWidth": 100,
        "name": "kharif",
        "trueText": "Yes",
        "width": 100,
        "xtype": "booleancolumn"
      }
    ],
    "createTitle": "Create a new Crop",
    "displayField": "label",
    "editHeight": 270,
    "editTitle": "Edit Crop",
    "fields": [
      {
        "mapping": "id",
        "name": "id"
      },
      {
        "mapping": "label",
        "name": "label"
      },
      {
        "mapping": "prod_default_unit",
        "name": "prod_default_unit"
      },
      {
```

```

        "mapping": "area_default_unit",
        "name": "area_default_unit"
    },
    {
        "mapping": "yield_default_unit",
        "name": "yield_default_unit"
    },
    {
        "header": "Rabi",
        "mapping": "rabi",
        "name": "rabi",
        "type": "boolean"
    },
    {
        "header": "Kharif",
        "mapping": "kharif",
        "name": "kharif",
        "type": "boolean"
    }
],
"form": {
    "create": [
        {
            "allowBlank": false,
            "fieldLabel": "Id",
            "name": "id",
            "reandomly": false,
            "xtype": "textfield"
        },
        {
            "allowBlank": false,
            "fieldLabel": "Label",
            "name": "label",
            "xtype": "textfield"
        },
        {
            "fieldLabel": "Season",
            "items": [
                {
                    "boxLabel": "Rabi",
                    "inputValue": true,
                    "name": "rabi"
                },
                {
                    "boxLabel": "Kharif",
                    "inputValue": true,
                    "name": "kharif"
                }
            ],
            "name": "seasons",
            "xtype": "checkboxgroup"
        },
        {
            "allowBlank": false,
            "displayField": "name",
            "fieldLabel": "Production Uom",
            "forceSelected": true,
            "hiddenName": "prod_default_unit",

```

```

        "mode": "local",
        "name": "prod_default_unit",
        "store": {
            "autoLoad": true,
            "fields": ["id", "name"],
            "idProperty": "id",
            "root": "data",
            "totalProperty": "total",
            "url": "/opensdi2-manager/mvc/cip/uom/filterby?attributename=cls&valueLil",
            "xtype": "jsonstore"
        },
        "triggerAction": "all",
        "valueField": "id",
        "xtype": "combo"
    },
    {
        "allowBlank": false,
        "displayField": "name",
        "fieldLabel": "Area Uom",
        "hiddenName": "area_default_unit",
        "mode": "local",
        "name": "area_default_unit",
        "store": {
            "autoLoad": true,
            "fields": ["id", "name"],
            "idProperty": "id",
            "root": "data",
            "totalProperty": "total",
            "url": "/opensdi2-manager/mvc/cip/uom/filterby?attributename=cls&valueLil",
            "xtype": "jsonstore"
        },
        "triggerAction": "all",
        "valueField": "id",
        "xtype": "combo"
    },
    {
        "allowBlank": false,
        "displayField": "name",
        "fieldLabel": "Yield Uom",
        "hiddenName": "yield_default_unit",
        "mode": "local",
        "name": "yield_default_unit",
        "store": {
            "autoLoad": true,
            "fields": ["id", "name"],
            "idProperty": "id",
            "root": "data",
            "totalProperty": "total",
            "url": "/opensdi2-manager/mvc/cip/uom/filterby?attributename=cls&valueLil",
            "xtype": "jsonstore"
        },
        "triggerAction": "all",
        "valueField": "id",
        "xtype": "combo"
    }
],
"edit": [
    {

```

```

        "allowBlank": false,
        "fieldLabel": "Id",
        "name": "id",
        "readOnly": true,
        "xtype": "textfield"
    },
    {
        "allowBlank": false,
        "fieldLabel": "Label",
        "name": "label",
        "xtype": "textfield"
    },
    {
        "fieldLabel": "Season",
        "items": [
            {
                "boxLabel": "Rabi",
                "inputValue": true,
                "name": "rabi"
            },
            {
                "boxLabel": "Kharif",
                "inputValue": true,
                "name": "kharif"
            }
        ],
        "name": "seasons",
        "xtype": "checkboxgroup"
    },
    {
        "allowBlank": false,
        "displayField": "name",
        "fieldLabel": "Production Uom",
        "hiddenName": "prod_default_unit",
        "mode": "local",
        "name": "prod_default_unit",
        "store": {
            "autoLoad": true,
            "fields": ["id", "name"],
            "idProperty": "id",
            "root": "data",
            "totalProperty": "total",
            "url": "/opensdi2-manager/mvc/cip/uom/filterby?attributename=cls&valueLi",
            "xtype": "jsonstore"
        },
        "triggerAction": "all",
        "value": "000_tons",
        "valueField": "id",
        "xtype": "combo"
    },
    {
        "allowBlank": false,
        "displayField": "name",
        "fieldLabel": "Production Uom",
        "hiddenName": "area_default_unit",
        "mode": "local",
        "name": "area_default_unit",
        "store": {

```

```

        "autoLoad": true,
        "fields": ["id", "name"],
        "idProperty": "id",
        "root": "data",
        "totalProperty": "total",
        "url": "/opensdi2-manager/mvc/cip/uom/filterby?attributename=cls&valueLi
        "xtype": "jsonstore"
    },
    "triggerAction": "all",
    "value": "000_ha",
    "valueField": "id",
    "xtype": "combo"
},
{
    "allowBlank": false,
    "displayField": "name",
    "fieldLabel": "Production Uom",
    "hiddenName": "yield_default_unit",
    "mode": "local",
    "name": "yield_default_unit",
    "store": {
        "autoLoad": true,
        "fields": ["id", "name"],
        "idProperty": "id",
        "root": "data",
        "totalProperty": "total",
        "url": "/opensdi2-manager/mvc/cip/uom/filterby?attributename=cls&valueLi
        "xtype": "jsonstore"
    },
    "triggerAction": "all",
    "value": "kg_ha",
    "valueField": "id",
    "xtype": "combo"
}
]
},
"iconCls": "nrl_crop_ic",
"id": "Crops",
"idProperty": "id",
"name": "Crop",
"pluralName": "Crops",
"restful": true,
"root": "data"
},
{
    "api": {
        "destroy": {
            "method": "DELETE",
            "url": "mvc/cip/agromet/{factor}"
        },
        "dump": {
            "method": "GET",
            "url": "mvc/cip/agromet/dump"
        },
        "restore": {
            "method": "POST",
            "url": "mvc/cip/agromet/restore"
        }
    }
}

```

```

    },
    "autoExpandColumn": "label",
    "basePath": "mvc/cip/agromet/",
    "canCreate": true,
    "canDelete": true,
    "canEdit": true,
    "columns": [
      {
        "allowBlank": false,
        "header": "Factor",
        "mapping": "factor",
        "name": "factor"
      },
      {
        "allowBlank": false,
        "header": "Label",
        "mapping": "label",
        "name": "label"
      },
      {
        "fixed": true,
        "header": "Aggregation",
        "mapping": "aggregation",
        "maxWidth": 100,
        "name": "aggregation",
        "width": 100
      },
      {
        "fixed": true,
        "header": "Unit",
        "mapping": "unit",
        "maxWidth": 100,
        "name": "unit",
        "width": 100
      }
    ],
    "createTitle": "Create a new variable",
    "displayField": "label",
    "editHeight": 200,
    "editTitle": "Edit Factor",
    "fields": [
      {
        "mapping": "factor",
        "name": "factor"
      },
      {
        "mapping": "label",
        "name": "label"
      },
      {
        "mapping": "aggregation",
        "name": "aggregation"
      },
      {
        "mapping": "unit",
        "name": "unit"
      }
    ]
  ],

```

```
"form": {
  "create": [
    {
      "allowBlank": false,
      "fieldLabel": "Factor",
      "name": "factor",
      "xtype": "textfield"
    },
    {
      "fieldLabel": "Label",
      "name": "label",
      "xtype": "textfield"
    },
    {
      "fieldLabel": "Unit",
      "name": "unit",
      "xtype": "textfield"
    },
    {
      "fieldLabel": "Aggregation",
      "items": [
        {
          "boxLabel": "avg",
          "checked": true,
          "inputValue": "avg",
          "name": "aggregation"
        },
        {
          "boxLabel": "sum",
          "inputValue": "sum",
          "name": "aggregation"
        }
      ],
      "xtype": "radiogroup"
    }
  ],
  "edit": [
    {
      "allowBlank": false,
      "fieldLabel": "Factor",
      "name": "factor",
      "readOnly": true,
      "xtype": "textfield"
    },
    {
      "allowBlank": false,
      "fieldLabel": "Label",
      "name": "label",
      "xtype": "textfield"
    },
    {
      "fieldLabel": "Unit",
      "name": "unit",
      "xtype": "textfield"
    },
    {
      "fieldLabel": "Aggregation",
      "items": [
```



```

        {
            "boxLabel": "avg",
            "inputValue": "avg",
            "name": "aggregation"
        },
        {
            "boxLabel": "sum",
            "inputValue": "sum",
            "name": "aggregation"
        }
    ],
    "xtype": "radiogroup"
}

],
"iconCls": "nrl_factor_ic",
"id": "Agromet",
"idProperty": "factor",
"name": "Agromet",
"pluralName": "Agromet Variables",
"restful": true,
"root": "data"
},
{
    "api": {
        "destroy": {
            "method": "DELETE",
            "url": "mvc/cip/uom/{id}"
        },
        "dump": {
            "method": "GET",
            "url": "mvc/cip/uom/dump"
        },
        "restore": {
            "method": "POST",
            "url": "mvc/cip/uom/restore"
        }
    },
    "basePath": "mvc/cip/uom/",
    "canCreate": true,
    "canDelete": true,
    "canEdit": true,
    "columns": [
        {
            "fixed": true,
            "header": "ID",
            "mapping": "id",
            "maxWidth": 100,
            "name": "id",
            "width": 100
        },
        {
            "header": "Label",
            "mapping": "name",
            "name": "name"
        },
        {
            "fixed": true,

```

```
        "header": "Short Name",
        "mapping": "shortname",
        "maxWidth": 100,
        "name": "shortname",
        "width": 100
    },
    {
        "header": "Description",
        "mapping": "description",
        "name": "description"
    },
    {
        "fixed": true,
        "header": "Class",
        "mapping": "cls",
        "maxWidth": 100,
        "name": "cls",
        "width": 100
    },
    {
        "fixed": true,
        "header": "factor",
        "mapping": "coefficient",
        "maxWidth": 100,
        "name": "coefficient",
        "width": 100
    },
    {
        "fixed": true,
        "header": "filter",
        "mapping": "filter",
        "maxWidth": 100,
        "name": "filter",
        "width": 100
    }
],
"createTitle": "Create a new Unit of Measure",
"displayField": "name",
"editHeight": 310,
"editTitle": "Edit Unit of Measure",
"fields": [
    {
        "mapping": "id",
        "name": "id"
    },
    {
        "mapping": "name",
        "name": "name"
    },
    {
        "mapping": "shortname",
        "name": "shortname"
    },
    {
        "mapping": "description",
        "name": "description"
    }
]
```

```

        "mapping": "cls",
        "name": "cls"
    },
    {
        "mapping": "coefficient",
        "name": "coefficient"
    },
    {
        "mapping": "filter",
        "name": "filter"
    }
],
"form": {
    "create": [
        {
            "allowBlank": false,
            "fieldLabel": "ID",
            "mapping": "id",
            "name": "id"
        },
        {
            "allowBlank": false,
            "fieldLabel": "Label",
            "mapping": "name",
            "name": "name"
        },
        {
            "allowBlank": false,
            "fieldLabel": "Short Name",
            "mapping": "shortname",
            "name": "shortname"
        },
        {
            "fieldLabel": "description",
            "header": "Description",
            "name": "description",
            "xtype": "textarea"
        },
        {
            "allowBlank": false,
            "displayField": "label",
            "fieldLabel": "Class",
            "mode": "local",
            "hiddenName": "cls",
            "store": {
                "fields": [
                    "name",
                    "label"
                ],
                "idProperty": "name",
                "inlineData": [
                    {
                        "label": "Production",
                        "name": "production"
                    },
                    {
                        "label": "Area",
                        "name": "area"
                    }
                ]
            }
        }
    ]
}

```

```
        },
        {
            "label": "Yield",
            "name": "yield"
        },
        {
            "label": "Denominator",
            "name": "denominator"
        },
        {
            "label": "Water Supply",
            "name": "watersupply"
        },
        {
            "label": "Water Flow",
            "name": "waterflow"
        }
    ],
    "xtype": "jsonstore"
},
"triggerAction": "all",
"valueField": "name",
"xtype": "combo"
},
{
    "allowBlank": false,
    "decimalPrecision": 10,
    "fieldLabel": "factor",
    "mapping": "coefficient",
    "name": "coefficient",
    "xtype": "numberfield"
},
{
    "fieldLabel": "filter",
    "mapping": "filter",
    "name": "filter"
}
],
"edit": [
    {
        "allowBlank": false,
        "fieldLabel": "ID",
        "name": "id",
        "readOnly": true
    },
    {
        "allowBlank": false,
        "fieldLabel": "Label",
        "name": "name"
    },
    {
        "allowBlank": false,
        "fieldLabel": "Short Name",
        "mapping": "shortname",
        "name": "shortname"
    },
    {
        "fieldLabel": "description",
```

```

        "header": "Description",
        "name": "description",
        "xtype": "textarea"
    },
    {
        "allowBlank": false,
        "displayField": "label",
        "fieldLabel": "Class",
        "hiddenName": "cls",
        "mode": "local",
        "name": "cls",
        "store": {
            "fields": [
                "name",
                "label"
            ],
            "idProperty": "name",
            "inlineData": [
                {
                    "label": "Production",
                    "name": "production"
                },
                {
                    "label": "Area",
                    "name": "area"
                },
                {
                    "label": "Yield",
                    "name": "yield"
                },
                {
                    "label": "Denominator",
                    "name": "denominator"
                },
                {
                    "label": "Water Supply",
                    "name": "watersupply"
                },
                {
                    "label": "Water Flow",
                    "name": "waterflow"
                }
            ],
            "mode": "local",
            "xtype": "jsonstore"
        },
        "triggerAction": "all",
        "valueField": "name",
        "xtype": "combo"
    },
    {
        "allowBlank": false,
        "decimalPrecision": 10,
        "fieldLabel": "factor",
        "mapping": "coefficient",
        "name": "coefficient",
        "xtype": "numberfield"
    },

```

```
        {
          "fieldLabel": "filter",
          "mapping": "filter",
          "name": "filter"
        }
      ]
    },
    "iconCls": "nrl_uom_ic",
    "id": "Units",
    "idProperty": "id",
    "name": "Unit of Measure",
    "pluralName": "Units of Measure",
    "restful": true,
    "root": "data"
  }
],
},
```

4.6.2 Configuration

This section introduces the main GeoBatch concepts and shows how to modify the flow configuration files related to the CROP Information Portal.

GeoBatch Flows at a glance

GeoBatch's basic idea is to perform a *chain of actions triggered by custom defined events* called **Flow**.

The possible **event generators** include monitoring for new files added to a directory, or receiving files in the embedded FTP server. **Actions** range from geotransforming an input raster file, to creating overviews, or publishing data into a GeoServer instance.

The *Actions* and *Event Generators* that compose a *Flow* are selected and configured through a *Flow Configuration* typically implemented as an XML document.

You can find more GeoBatch information visiting the [official GeoBatch documentation](#)

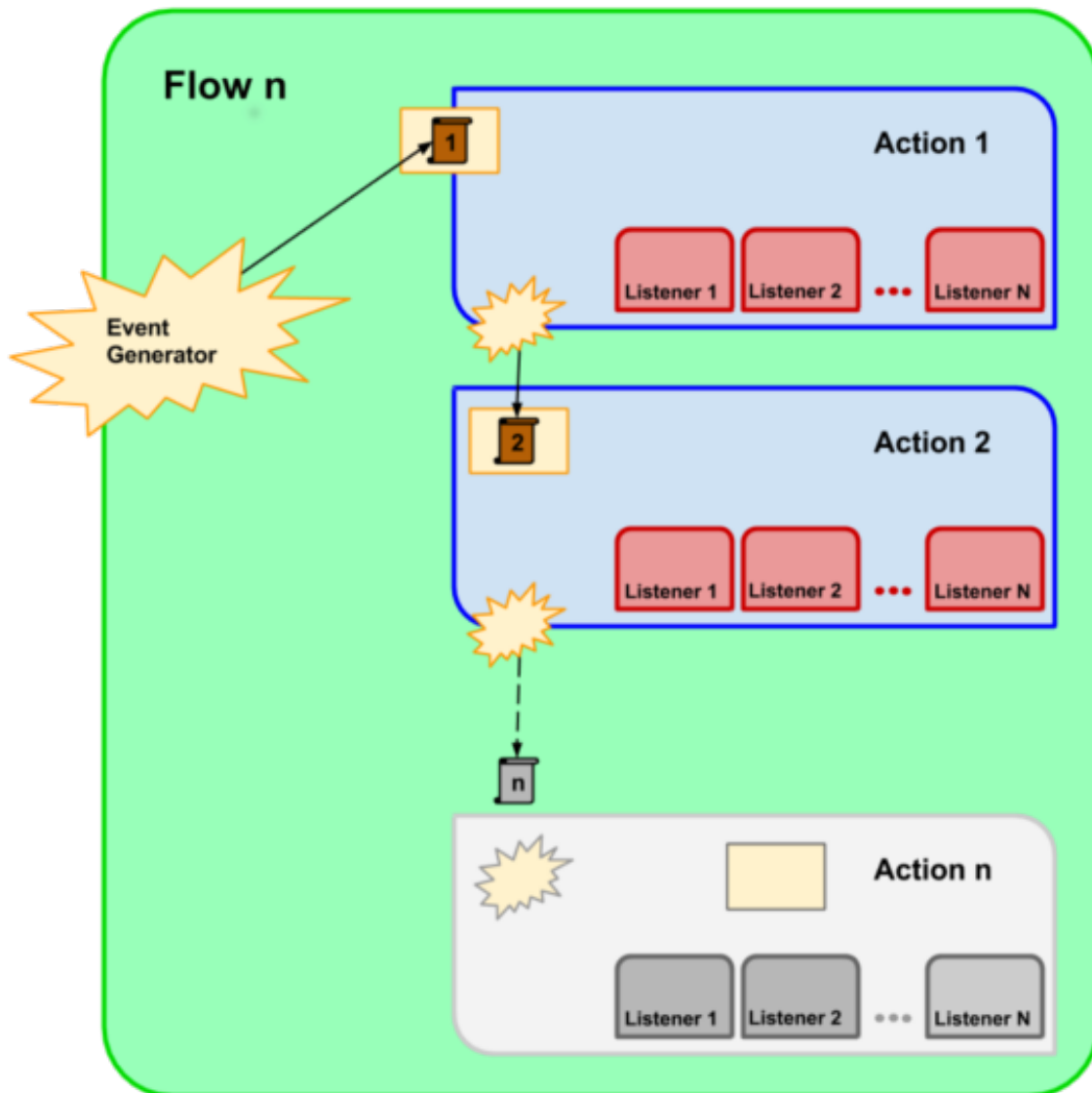
Create-Or-Update-Layer ingestion flow

```
<?xml version="1.0" encoding="UTF-8" ?>
<FlowConfiguration>

  <id>createupdatelayer</id>
  <description>Updates or creates a layer from a given shapefile</description>
  <name>Create or update Layer</name>

  <autorun>true</autorun>

  <EventGeneratorConfiguration>
    <wildCard>*.*/wildCard>
    <watchDirectory>createupdatelayer/in</watchDirectory>
    <osType>OS_UNDEFINED</osType>
    <eventType>FILE_ADDED</eventType>
    <id>Ds2dsFlow</id>
    <keepFiles>true</keepFiles>
    <serviceID>fsEventGeneratorService</serviceID>
```



```
<description>Ds2ds event generator description</description>
<name>Ds2ds</name>
</EventGeneratorConfiguration>

<EventConsumerConfiguration>
  <id>ds2ds</id>
  <description>ds2ds</description>
  <name>ds2ds</name>
  <performBackup>false</performBackup>
  <listenerId>ConsumerLogger0</listenerId>
  <listenerId>Cumulator</listenerId>

  <Ds2dsConfiguration>
    <id>Ds2dsGeneratorService</id>
    <description>Ds2ds action</description>
    <name>Ds2dsConfiguration</name>

    <listenerConfigurations/>
    <listenerId>ConsumerLogger0</listenerId>
    <listenerId>Cumulator</listenerId>

    <failIgnored>false</failIgnored>
    <purgeData>true</purgeData>
    <moveData>false</moveData>

    <outputFeature>
      <dataStore>
        <entry>
          <string>dbtype</string>
          <string>postgis</string>
        </entry>
        <entry>
          <string>host</string>
          <string>localhost</string>
        </entry>
        <entry>
          <string>port</string>
          <string>5432</string>
        </entry>
        <entry>
          <string>database</string>
          <string>NRL</string>
        </entry>
        <entry>
          <string>schema</string>
          <string>public</string>
        </entry>
        <entry>
          <string>user</string>
          <string>geoserver</string>
        </entry>
        <entry>
          <string>passwd</string>
          <string>*****</string>
        </entry>
      </dataStore>
    </outputFeature>
  </Ds2dsConfiguration>
</EventConsumerConfiguration>
```



```

    </Ds2dsConfiguration>
  </EventConsumerConfiguration>

  <ListenerConfigurations>
    <LoggingProgressListener>
      <serviceID>loggingListenerService</serviceID>
      <id>ConsumerLogger0</id>
      <loggerName>it.geosolutions.ConsLogger</loggerName>
      <appendToListenerForwarder>true</appendToListenerForwarder>
    </LoggingProgressListener>

    <CumulatingProgressListener>
      <serviceID>cumulatingListenerService</serviceID>
      <id>Cumulator</id>
      <appendToListenerForwarder>true</appendToListenerForwarder>
    </CumulatingProgressListener>

    <StatusProgressListener>
      <serviceID>statusListenerService</serviceID>
      <id>Status</id>
      <appendToListenerForwarder>true</appendToListenerForwarder>
    </StatusProgressListener>
  </ListenerConfigurations>
</FlowConfiguration>

```

NDVI stats flow

The **NDVI stats flow** is responsible for compute statistics over a raster using a vector layer to determine a Region Of Interest in which the computation will be performed.

The output format of the statistics is CSV.

The Statistics are internally computed using the ZonalStats operators of the [JaiTools project](#) project.

The flow is composed by a single Action called **NDVIStatsAction** that perform all the operations needed.

The flow periodically check for an XML file inside a FileSystem location: for each XML file found a new execution of the flow will be thrown.

In the following paragraphs will be shown the format of the XML input file and the most important configuration settings.

The input XML file format

In order to run the flow, an user must place inside the watch directory an XML file. See the following file as example:

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<statsBean>
  <classifier>DISTRICT</classifier>
  <forestMask>DISABLED</forestMask>
  <ndviFileName>dv_19980401_19980410.tif</ndviFileName>
</statsBean>

```

- **classifier** The Vector data used as classifier layer. Accepted values are:
 - **PROVINCE** Aggregate results by provinces

- **DISTRICT** Aggregate results by districts
- **CUSTOM** Use a custom classifier specifying its absolute path. The path must be write adding one more field inside the statsBean tag called **classifierFullPath**
- **forestMask** The mask used as Region Of Interest to compute the statistics. Accepted Values are:
 - **STANDARD** Use the mask specyified in the flow configuration (see later in the *Action Configuration* paragraph)
 - **CUSTOM** Similar to the value CUSTOM used in classifier. The tag to add is called **forestMaskFullPath**. Place there the absolute path of the shapefile.
 - **DISABLED** Don't use any mask
- **ndviFileName** the file name of the NDVI mosaic granule to use to compute the NDVI statistics

Generate the input XML using the Crop Portal administration webapp

The Crop Information Portal has its own Administration interface that allow the user to run the flow without dealing with XML files and FileSystem directly usage.

Refer to the Administration Page documentation for more details.

Through that web interface is it possible to run the flow directly from the browser. The Crop Portal administration application will create the xml and will place it in the flow watch dir.

EventGenerator Configuration

```
<wildCard>*.xml</wildCard>
<watchDirectory>ndvistats/in</watchDirectory>
```

- **wildCard** field is possible to specify which file name/extension looking for. In this case the first (and only) action is developed to accept only XML files so the flow will be launched only if an XML file will be placed in the whatch directory.
- **watchDirectory** is the directory where GB watches for new input files. It is relative to **Geobatch Config Dir**. See the official GB documentation for more info about the Geobatch Config Directory

```
<interval>0 0/15 4-6 * * ?</interval>
<interval>* * * * * ?</interval>
<interval>SEC MIN HR DOM MON DOW</interval>
```

- **interval** in this field is possible to specify the GeoBatch frequency to checks for new files in the flow's watch directory.

The format to use is the *Quartz syntax* <<http://quartz-scheduler.org/documentation/quartz-2.1.x/tutorials/crontrigger>> that basically is an evolution of the Linux Crontab syntax.

Action Configuration

```
<defaultMaskUrl>file:/opt/gs_data_dir/data/spatial/CROPMASKS/crop_mask_pak_2012.shp</defaultMaskUrl>
```

- **defaultMaskUrl** is the default mask used if no other mask are specified.

```
<dbType>postgis</dbType>
<dbHost>localhost</dbHost>
<dbPort>5432</dbPort>
<dbSchema>public</dbSchema>
<dbName>NRL</dbName>
<dbUser>geoserver</dbUser>
<dbPasswd>*****</dbPasswd>
```

- The Postgres connection parameters.

```
<tiffDirectory>/opt/mosaics/ndvi</tiffDirectory>
```

- **tiffDirectory** The directory where the NDVI rasters are stored (That is the Mosaic Directory used by geoserver)

```
<outputDirectory>/opt/admin_dir/</outputDirectory>
```

- **outputDirectory** The directory where the output csv will be stored.

The whole Flow Configuration

```
<?xml version="1.0" encoding="UTF-8"?>
<FlowConfiguration>

  <id>ndvistats</id>
  <name>NDVI stats generation</name>
  <description>Generate a CSV file with a geotiff mask and a zone filter</description>

  <corePoolSize>2</corePoolSize>
  <maximumPoolSize>2</maximumPoolSize>
  <keepAliveTime>1500</keepAliveTime>
  <workQueueSize>100</workQueueSize>

  <autorun>true</autorun>

  <EventGeneratorConfiguration>
    <id>ndvistats_event_gen</id>
    <serviceID>fsEventGeneratorService</serviceID>
    <wildCard>*.xml</wildCard>
    <watchDirectory>ndvistats/in</watchDirectory>
    <osType>OS_UNDEFINED</osType>
    <eventType>FILE_ADDED</eventType>
    <interval>* * * * * ?</interval>
  </EventGeneratorConfiguration>

  <EventConsumerConfiguration>
    <id>ndvistats_consumer</id>

    <listenerId>Logger0</listenerId>

    <listenerId>Cumulator</listenerId>
    <performBackup>false</performBackup>
    <preserveInput>true</preserveInput>

    <NDVIStatsConfiguration>
      <listenerId>Logger</listenerId>
      <listenerId>Cumulator</listenerId>
      <listenerId>Status</listenerId>
```

```

    <id>NDVIIngestConfiguration</id>
    <name>NDVI CSV stats preparation</name>
    <description>Prepare time interval in TIF filenames </description>
    <defaultMaskUrl>file:/opt/gs_data_dir/data/spatial/CROPMASKS/crop_mask_pak_2012.shp</defaultMaskUrl>
    <dbType>postgis</dbType>
    <dbHost>localhost</dbHost>
    <dbPort>5432</dbPort>
    <dbSchema>public</dbSchema>
    <dbName>NRL</dbName>
    <dbUser>geoserver</dbUser>
    <dbPasswd>*****</dbPasswd>
    <tiffDirectory>/opt/mosaics/ndvi</tiffDirectory>
    <outputDirectory>/opt/admin_dir/</outputDirectory>
    <csvSeparator>,</csvSeparator>
  </NDVIStatsConfiguration>

</EventConsumerConfiguration>

<ListenerConfigurations>
  <LoggingProgressListener>
    <serviceID>loggingListenerService</serviceID>
    <id>Logger0</id>
    <loggerName>it.geosolutions.geobatch</loggerName>
  </LoggingProgressListener>
  <LoggingProgressListener>
    <serviceID>loggingListenerService</serviceID>
    <id>Logger</id>
    <loggerName>it.geosolutions.geobatch</loggerName>
  </LoggingProgressListener>
  <CumulatingProgressListener>
    <serviceID>cumulatingListenerService</serviceID>
    <appendToListenerForwarder>true</appendToListenerForwarder>
    <id>Cumulator</id>
  </CumulatingProgressListener>
  <StatusProgressListener>
    <serviceID>statusListenerService</serviceID>
    <id>Status</id>
  </StatusProgressListener>
</ListenerConfigurations>
</FlowConfiguration>

```

NDVI ingestion flow

```

<?xml version="1.0" encoding="UTF-8"?>
<FlowConfiguration>

  <id>ndviingestion</id>
  <name>NDVI file ingestion</name>
  <description>Ingest geotiff file and add them to a NDVI mosaic</description>

  <corePoolSize>2</corePoolSize>
  <maximumPoolSize>2</maximumPoolSize>
  <keepAliveTime>1500</keepAliveTime>
  <workQueueSize>100</workQueueSize>

  <autorun>true</autorun>

```

```

<EventGeneratorConfiguration>
  <id>ndvi_event_gen</id>
  <serviceID>fsEventGeneratorService</serviceID>
  <wildCard>*.tif</wildCard>
  <watchDirectory>ndviingestion/in</watchDirectory>
  <osType>OS_UNDEFINED</osType>
  <eventType>FILE_ADDED</eventType>
  <interval>0 0/15 4-6 * * ?</interval>
</EventGeneratorConfiguration>

<EventConsumerConfiguration>
  <id>ndvi_consumer</id>

  <listenerId>LoggingListener</listenerId>
  <listenerId>CumulatingListener</listenerId>

  <performBackup>false</performBackup>
  <preserveInput>true</preserveInput>

  <NDVIIngestConfiguration>
    <id>NDVIIngestConfiguration</id>
    <name>NDVI file preparation</name>
    <description>Prepare time interval in TIF filenames </description>

    <destinationDir>/opt/mosaics/ndvi</destinationDir>
  </NDVIIngestConfiguration>

  <ImageMosaicActionConfiguration>

    <id>ImageMosaicService</id>

    <name>NDVI image mosaic</name>
    <description>Add geotiff to the NDVI mosaic</description>

    <listenerId>LoggingListener</listenerId>
    <listenerId>CumulatingListener</listenerId>
    <listenerId>StatusListener</listenerId>

    <crs>EPSG:4326</crs>
    <envelope/>

    <dataTransferMethod>EXTERNAL</dataTransferMethod>

    <geoserverUID>admin</geoserverUID>
    <geoserverPWD>*****</geoserverPWD>
    <geoserverURL>http://localhost/geoserver</geoserverURL>

    <defaultNamespace>ndvi</defaultNamespace>
    <defaultStyle>raster</defaultStyle>

    <wmsPath>/</wmsPath>
    <backgroundValue>NaN</backgroundValue>
    <outputTransparentColor/>
    <inputTransparentColor/>
    <allowMultithreading>true</allowMultithreading>
    <useJaiImageRead>false</useJaiImageRead>

```

```

<tileSizeH>256</tileSizeH>
<tileSizeW>256</tileSizeW>
<NativeMinBoundingBoxX>-30</NativeMinBoundingBoxX>
<NativeMinBoundingBoxY>25</NativeMinBoundingBoxY>
<NativeMaxBoundingBoxX>45</NativeMaxBoundingBoxX>
<NativeMaxBoundingBoxY>70</NativeMaxBoundingBoxY>

<latLonMinBoundingBoxX>-30</latLonMinBoundingBoxX>
<latLonMinBoundingBoxY>25</latLonMinBoundingBoxY>
<latLonMaxBoundingBoxX>45</latLonMaxBoundingBoxX>
<latLonMaxBoundingBoxY>70</latLonMaxBoundingBoxY>

<!--NONE, REPROJECT_TO_DECLARED, FORCE_DECLARED-->
<projectionPolicy>NONE</projectionPolicy>

<!-- ref dir is GB_CONF_DIR/FLOWID/ACTIONID -->
<datastorePropertiesPath>../datastore.properties</datastorePropertiesPath>

<!-- METADATA -->
<!-- file name is file_20130201_20130210.tif -->
<DomainAttribute>
  <dimensionName>time</dimensionName>
  <attribName>time</attribName>
  <regEx><![CDATA[(\?<=dv_) [0-9]{8}(\?=_.\*)]]></regEx>
  <endRangeAttribName>endtime</endRangeAttribName>
  <endRangeRegEx><![CDATA[(\?<=dv_[0-9]{8})_ [0-9]{8}(\?=_.\*)]]></endRangeRegEx>
</DomainAttribute>

</ImageMosaicActionConfiguration>

</EventConsumerConfiguration>

<ListenerConfigurations>
  <LoggingProgressListener>
    <serviceID>loggingListenerService</serviceID>
    <id>LoggingListener</id>
    <loggerName>it.geosolutions.geobatch</loggerName>
  </LoggingProgressListener>
  <CumulatingProgressListener>
    <serviceID>cumulatingListenerService</serviceID>
    <id>CumulatingListener</id>
  </CumulatingProgressListener>
  <StatusProgressListener>
    <serviceID>statusListenerService</serviceID>
    <id>StatusListener</id>
  </StatusProgressListener>
</ListenerConfigurations>
</FlowConfiguration>

```

The action automatically do a *retiling* and generate the overviews to increase performances for web usage. Processing the file name, the action publish the tif as a granule in geoserver for the right time period (expressed in the file name). The granule will be available using the WMS `time` parameter. for instance if we have `dv98041_pak` (1st dekad of January 1998) the image will be available passing from Jan 1st 1998 to Jan 9th 1998 (whole day).

File format

The name of the Geotiff file must have this format: `dvYYMMDD_*.tif` with:

- **YY:** Year of the image: two last digits of the year. *e.g. 1998 becomes 98; 2000 becomes 00.*
- **MM:** Month of the image: between 01 (January) and 12 (December)
- **D:** 1,2 or 3. This represents the dekad.

The image to ingest must have these characteristics:

- **DataType:** Byte, 1 band
- **Bounding Box** 59.7723211 23.0669642 80.4330354 37.6919642
- **Size:** 2314x1638 px

here a sample gdalinfo output of a compatible file:

```
Driver: GTiff/GeoTIFF
Files: dv98041_pak.tif
Size is 2314, 1638
Coordinate System is:
GEOGCS["WGS 84",
    DATUM["WGS_1984",
        SPHEROID["WGS 84",6378137,298.257223563,
            AUTHORITY["EPSG","7030"]],
        AUTHORITY["EPSG","6326"]],
    PRIMEM["Greenwich",0],
    UNIT["degree",0.0174532925199433],
    AUTHORITY["EPSG","4326"]]
Origin = (59.772321143118091,37.691964174177784)
Pixel Size = (0.008928571417941,-0.008928571417941)
Metadata:
  AREA_OR_POINT=Area
  TIFFTAG_RESOLUTIONUNIT=1 (unitless)
  TIFFTAG_SOFTWARE=IMAGINE TIFF Support
Copyright 1991 - 1999 by ERDAS, Inc. All Rights Reserved
@(#) $RCSfile: etif.c $ $Revision: 1.10.1.9.1.9.2.11 $ $Date: 2004/09/15 18:42:01
EDT $
  TIFFTAG_XRESOLUTION=1
  TIFFTAG_YRESOLUTION=1
Image Structure Metadata:
  INTERLEAVE=BAND
Corner Coordinates:
Upper Left  ( 59.7723211,  37.6919642) ( 59d46'20.36"E, 37d41'31.07"N)
Lower Left  ( 59.7723211,  23.0669642) ( 59d46'20.36"E, 23d 4' 1.07"N)
Upper Right ( 80.4330354,  37.6919642) ( 80d25'58.93"E, 37d41'31.07"N)
Lower Right ( 80.4330354,  23.0669642) ( 80d25'58.93"E, 23d 4' 1.07"N)
Center      ( 70.1026783,  30.3794642) ( 70d 6' 9.64"E, 30d22'46.07"N)
Band 1 Block=64x64 Type=Byte, ColorInterp=Gray
```

Delete NDVI To delete an NDVI granule follow this instructions: Using rest interface a user should use curl to get the id of the granule he want to remove.

```
curl `GET /geoserver/rest/workspaces/ndvi/coveragestores/ndvi/coverages/ndvi/index/granules.json`

Response:
{
  "type": "FeatureCollection",
  "bbox": [
    59.7633927459,
    23.0580358005,
```

```
      80.4419634523,
      37.6919644531
    ],
    "crs":{
      "type":"name",
      "properties":{
        "name":"EPSG:4326"
      }
    },
    "features":[
      {
        "type":"Feature",
        "geometry":{
          "type":"Polygon",
          "coordinates":[
            [
              [
                59.7723,
                23.067
              ],
              [
                59.7723,
                37.692
              ],
              [
                80.433,
                37.692
              ],
              [
                80.433,
                23.067
              ],
              [
                59.7723,
                23.067
              ]
            ]
          ]
        },
        "properties":{
          "location":"dv_20130101_20130110.tif",
          "time":"2013-01-01T00:00:00.000+0000",
          "endtime":"2013-01-10T00:00:00.000+0000"
        },
        "id":"ndvi.1"
      },
      [...]
    ]
  }
}
```

You can find the granule you want to delete checking the date *properties.time* that is the start time of the granule. For instance, if you want the 2016, Jan, 1st 10-day period, you have to look for the *2016-01-01T00:00:00.000+0000*

Then he can use the same interface to remove the granule by id : .. sourcecode:: bash

```
curl DELETE/geoserver/rest/workspaces/ndvi/coveragestores/ndvi/coverages/ndvi/index/granules/<granuleId>.json
```


E.g. : *DELETE/geoserver/rest/workspaces/ndvi/coveragestores/ndvi/coverages/ndvi/index/granules/ndvi.1.json*

CSV ingestion flow

```
<?xml version="1.0" encoding="UTF-8"?>
<FlowConfiguration>

  <id>csvingestion</id>
  <description>Ingest CSV files (cropdata and agromet)</description>
  <name>CSV file ingestion</name>

  <autorun>true</autorun>

<EventGeneratorConfiguration>
  <serviceID>fsEventGeneratorService</serviceID>
  <wildCard>*. *</wildCard>
  <watchDirectory>csvingestion/in</watchDirectory>
  <osType>OS_UNDEFINED</osType>
  <eventType>FILE_ADDED</eventType>
</EventGeneratorConfiguration>

  <EventConsumerConfiguration>
    <CSVIngestConfiguration>
      <id>csv</id>
      <name>CSVIngestion</name>
      <description>Ingest cropdata, cropdescriptor, agromet</description>
      <csvSeparator>,</csvSeparator>
      <emptyFieldsAsZero>>false</emptyFieldsAsZero>
      <rowByRow>>false</rowByRow>
      <!-- Listeners -->
      <listenerID>ConsumerLogger0</listenerID>
      <listenerID>Cumulator</listenerID>
      <listenerID>Status</listenerID>
    </CSVIngestConfiguration>
  </EventConsumerConfiguration>

  <ListenerConfigurations>
    <LoggingProgressListener>
      <serviceID>loggingListenerService</serviceID>
      <id>ConsumerLogger0</id>
      <loggerName>it.geosolutions.ConsLogger</loggerName>
      <appendToListenerForwarder>true</appendToListenerForwarder>
    </LoggingProgressListener>
    <CumulatingProgressListener>
      <serviceID>cumulatingListenerService</serviceID>
      <id>Cumulator</id>
      <appendToListenerForwarder>true</appendToListenerForwarder>
    </CumulatingProgressListener>
    <StatusProgressListener>
      <serviceID>statusListenerService</serviceID>
      <id>Status</id>
      <appendToListenerForwarder>true</appendToListenerForwarder>
    </StatusProgressListener>
  </ListenerConfigurations>

</FlowConfiguration>
```

Maintenance

In this section we're going to learn how to compile, install and update the platform.

All code you need it's on the [github repository](#). Please follow the instructions in the link before continue with this training.

Also, you need a version of this products:

- GeoStore
- Http-proxy
- GeoServer
- GeoNetwork

and more infrastructure applications:

- Apache Tomcat 6.0.37
- Apache Httpd (Web server)
- PostgreSQL - PostGIS

On the actual deployment all this stuff is already installed and configured, but if you want to install the application in a new server, you must install it and configure it.

5.1 Compile

In this section we're going to learn how to compile different parts of the application. The source code is available on Github repository:

- <https://github.com/geosolutions-it/crop-information-portal>

Please, follow the instructions in the link before continue.

The code of the project is divided in three different applications:

- OpenSDI-Manager2
- GeoBatch
- MapStore

OpenSDI-Manager2 and GeoBatch costumizations use Maven to compile it and MapStore uses Ant.

5.1.1 OpenSDI-Manager2

To compile the OpenSDI-Manager application:

```
crop-information-portal/OpenSDI-Manager/src$ mvn clean install -Dmaven.test.skip
...
[INFO] Reactor Summary:
[INFO]
[INFO] OpenSDI-Manager Portal ..... SUCCESS [0.489s]
[INFO] OpenSDI Manager - Test core ..... SUCCESS [2.008s]
[INFO] OpenSDI Manager - core ..... SUCCESS [3.558s]
[INFO] OpenSDI Manager - Admin Webapp ..... SUCCESS [2.577s]
[INFO] OpenSDI Manager - Log operation ..... SUCCESS [0.041s]
[INFO] OpenSDI Manager - Log operation - core ..... SUCCESS [0.676s]
[INFO] OpenSDI Manager - Log operation - web extension ... SUCCESS [4.435s]
[INFO] SHP file browser - OpenSDI-Manager extension ..... SUCCESS [4.362s]
[INFO] OpenSDI Manager - Crop Information Portal ..... SUCCESS [0.082s]
[INFO] OpenSDI Manager - Crop Information Portal - core .. SUCCESS [1.093s]
[INFO] OpenSDI Manager - Crop Information Portal - MVC extension SUCCESS [1.053s]
[INFO] OpenSDI Manager - Crop Information Portal - webapp extension SUCCESS [4.636s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 35.109s
[INFO] Finished at: Fri Jan 10 16:20:45 CET 2014
[INFO] Final Memory: 52M/147M
crop-information-portal/OpenSDI-Manager/src$
```

Now you have a `opensdi2-manager` application to be deployed on `crop-information-portal/OpenSDI-Manager/src/web/target/opensdi2-manager.war`.

You can deploy it on your web container and change the configuration as you need.

Please, read the Configuration of the Administration Interface training

5.1.2 GeoBatch

To compile the GeoBatch application:

```
crop-information-portal/OpenSDI-Manager/geobatch$ mvn clean install -Dmaven.test.skip
...
[INFO] Reactor Summary:
[INFO]
[INFO] NRL GeoBatch Root ..... SUCCESS [7.362s]
[INFO] GeoBatch action: CSV ingest ..... SUCCESS [12.779s]
[INFO] GeoBatch action: NDVI preparation ..... SUCCESS [17.881s]
[INFO] GeoBatch action: NDVI stats generation ..... SUCCESS [7.814s]
[INFO] NRL GeoBatch webapp ..... SUCCESS [21.242s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1:08.150s
[INFO] Finished at: Wed Jan 15 17:58:24 CET 2014
[INFO] Final Memory: 32M/97M
[INFO] -----
crop-information-portal/OpenSDI-Manager/geobatch$
```

Now you have a GeoBatch application to be deployed on *crop-information-portal/OpenSDI-Manager/geobatch/webapp/target/geobatch.war*.

You can deploy it on your web container and change the configuration as you need.
Please, read the GeoBatch configuration training

5.1.3 Mapstore

To compile the MapStore application:

1. **Init:** Download dependencies and build ringo jars:

```
crop-information-portal/mapstore$ ant init
Buildfile: crop-information-portal/mapstore/build.xml

init:
    [echo] building ringo jars

init:

compile:
    [mkdir] Created dir: crop-information-portal/mapstore/externals/ringojs/build/classes
    [javac] Compiling 52 source files to crop-information-portal/mapstore/externals/ringojs/build/classes

jar:
    [jar] Building jar: crop-information-portal/mapstore/externals/ringojs/lib/ringo.jar
    [jar] Building jar: crop-information-portal/mapstore/externals/ringojs/run.jar

BUILD SUCCESSFUL
Total time: 2 seconds
crop-information-portal/mapstore$
```

2. **Package:** Compile an package it into a war:

```
crop-information-portal/mapstore$ ant war
...
war:
    [war] Building war: crop-information-portal/mapstore/mapcomposer/build/mapcomposer.war
    [war] Building war: crop-information-portal/mapstore/mapmanager/build/mapstore.war

BUILD SUCCESSFUL
Total time: 24 seconds
crop-information-portal/mapstore$
```

Now you have a MapStore application to be deployed on *crop-information-portal/mapstore/mapcomposer/build/mapcomposer.war*.

You can deploy it on your web container and change the configuration as you need.
Please, read the MapStore Configuration training

5.2 Install

In this section we're going to learn how to install the platform in a new server.
The platform for this training doc is a CentOS 6.

5.2.1 Dependencies

You need to install this components on your server:

- JDK 6
- JAI Image-IO
- GDAL
- Apache Tomcat 6.0.37 (four instances)
- Apache Httpd (Web server)
- PostgreSQL - PostGIS

We assume you know how to install this components.

Services table

When you will finished you must have something like that:

Application	Local path	Port	Access by AJP proxy
Tomcat GeoServer	/opt/tomcat_geoserver	gs_port	/geoserver
Tomcat GeoBatch	/opt/tomcat_geobatch	gb_port	/geobatch
Tomcat GeoStore	/opt/tomcat_geostore	gst_port	/geostore
Tomcat GUI	/opt/tomcat_gui	gui_port	/MapStore, /opensdi2-manager, /geonetwork
PostgreSQL/PostGIS	/usr/local/postgres	pg_port	NO

So, we explain a little bit how to install those components.

JDK 6

1. Download jdk-6u45-linux-x64-rpm.bin from the Oracle site.
2. Install:

```
#INSTALL JDK6
sh jdk-6u45-linux-x64-rpm.bin

mkdir jvm
mv *.rpm jvm/
cd jvm/
mkdir java
mv jvm dk-6u45-linux-x64-rpm.bin java
```

JAI - Image IO

1. Download JAI:

```
#download installer native JAI
wget http://download.java.net/media/jai/builds/release/1_1_3/jai-1_1_3-lib-linux-amd64-jdk.bin
```

2. Copy to Java dir:

```
#copy to java dir
cp jai-1_1_3-lib-linux-amd64-jdk.bin /usr/java/jdk1.6.0_45/
cd /usr/java/jdk1.6.0_45/
```

3. Install:

```
#install native jai
sh jai-1_1_3-lib-linux-amd64-jdk.bin
# ***** accept license, type yes *****
```

4. Clean the installation:

```
#remove binary
rm jai-1_1_3-lib-linux-amd64-jdk.bin
```

5. Download JAI Image-IO:

```
#download
wget http://download.java.net/media/jai-imageio/builds/release/1.1/jai_imageio-1_1-lib-linux-amd64-jdk.bin
#needed for compatibility versions
export _POSIX2_VERSION=199209
```

6. Copy to Java dir:

```
#copy
cp jai_imageio-1_1-lib-linux-amd64-jdk.bin /usr/java/jdk1.6.0_45/
cd /usr/java/jdk1.6.0_45/
```

7. Install:

```
#install
sh jai_imageio-1_1-lib-linux-amd64-jdk.bin
# ***** accept license, type yes *****
```

8. Clean the installation:

```
#remove binary
rm jai_imageio-1_1-lib-linux-amd64-jdk.bin
```

GDAL

1. Download repository:

```
curl -O http://mirror.i3d.net/pub/fedora-epel/6/x86_64/epel-release-6-8.noarch.rpm
```

2. Install repository:

```
rpm -ivh epel-release-6-8.noarch.rpm
```

3. Install GDAL:

```
yum install gdal
```

PostgreSQL/PostGIS

To install and configure the database you need to access to the server and follow the next steps:

1. Download and install PostgreSQL repository:

```
wget http://yum.pgrpms.org/9.1/redhat/rhel-5-x86_64/pgdg-centos91-9.1-4.noarch.rpm
yum install pgdg-centos91-9.1-4.noarch.rpm
```

2. Install PostgreSQL:

```
# POSTGRES
yum install postgresql91-contrib postgresql91-server postgresql91-contrib proj geos
```

2. Install PostGIS:

```
#POSTGIS
yum install postgis2_91 postgis2_91-devel
```

3. Install extra packages:

```
#other useful packages
yum install proj-devel geos-devel libxml2-devel libpgxx-devel proj-epsg
```

4. Initialize database:

```
#initialize database
service postgresql-9.1 initdb
```

5. Local access: Edit the access configuration to allow user login from localhost:

```
vim /var/lib/pgsql/9.1/data/pg_hba.conf
# "local" is for Unix domain socket connections only
local    all             all                                     trust
# IPv4 local connections:
host     all             all             127.0.0.1/32             md5
# IPv6 local connections:
host     all             all             ::1/128                 md5
```

6. Start database:

```
#start database
service postgresql-9.1 start
```

7. Auto run and user:

```
# Auto run on Startup
chkconfig postgresql-9.1 on
chkconfig --add postgresql-9.1
#alter postgres password
su postgres
psql
ALTER USER postgres WITH ENCRYPTED PASSWORD 'postgres';
\q
exit
```

Dump restore

You need to create two different databases:

- NRL
- geostore

The default system configuration use this roles (user@password: schema syntax) to connect:

- geoserver@lgeosolutions2:NRL
- geostore@geostore:geostore

You can restore the default databases and roles state with the dump:

```
psql -f crop.dump -U postgres
```

The dump file is attached on the training docs on ROOT/training/data/maintenance.

Apache Tomcat

1. Prepare folders:

```
mkdir /opt/
mkdir tomcat
cd tomcat
```

2. Download and copy in opt:

```
#download and copy in opt
wget ftp://mirror.nohup.it/apache/tomcat/tomcat-6/v6.0.37/bin/apache-tomcat-6.0.37.tar.gz
tar xzvf apache-tomcat-6.0.37.tar.gz -C /opt/

#create symbolic link
ln -s /opt/apache-tomcat-6.0.37/ /opt/tomcat
ls /opt
# should be: (contain apache tomcat and tomcat)
# apache-tomcat-6.0.37  sun  tomcat
```

3. Create Catalina base:

```
#create catalina base
#mkdir -p /var/lib/tomcat/base/{bin,conf,logs,temp,webapps,work}
cp /opt/tomcat/conf/* /var/lib/tomcat/base/conf/
```

4. Customize different instances: You need to create different instances for tomcat as you can see on [Services table](#).

5. User tomcat:

```
#create tomcat user
adduser -m -s /bin/bash tomcat
# don't change password for now
# passwd tomcat
```

AJP configuration

You must prepare various tomcat instances configuring different ports with the [AJP Connector](#) on each server.xml.

For example, you need to change the server shutdown port:

```
<Server port="8008" shutdown="SHUTDOWN">
```

and this connector:

```
<Connector port="8012" protocol="AJP/1.3" redirectPort="8446" />
```

for each instance.

Apache Httpd

1. Install:

```
yum install httpd
```

2. Basic configuration:

```
chkconfig --level 2345 httpd on
# enable incoming connection
#enable incoming tcp connection on port 80
```

4. Open port 80:

```
iptables -I INPUT -p tcp --dport 80 -j ACCEPT
service iptables save
```

5. Create configurations to the tomcat instances as you can see on [Services table](#).

Mod proxy AJP configuration

You must configure the access for each service present on the [Services table](#) with the Apache Module `mod_proxy_ajp`.

For example, you need to add this configuration for the MapStore instance:

```
ProxyPass /MapStore ajp://localhost:gst_port/MapStore
ProxyPassReverse /MapStore ajp://localhost:gst_port/MapStore
```

Services

All tomcat instances should be installed as default services on the system. The configuration files are available on `ROOT/training/data/maintenance/config.tar.gz`:

```
## APACHE WEB SERVER

cp httpd/conf.d/* /etc/httpd/conf.d/

#CONFIGURE MAPSTORE
configure urls

##TOMCAT STARTUP SCRIPTS
cp config/init.d/* /etc/init.d/

# start services on startup
chkconfig geoserver on
chkconfig --add geoserver

chkconfig geobatch on
chkconfig --add geobatch

chkconfig geostore on
chkconfig --add geostore
```

```
chkconfig gui on
chkconfig --add gui
```

5.2.2 Platform install

You need to compile the platform. Then you will have:

- OpenSDI-Manager2 war (opensdi2-manager.war)
- GeoBatch war (geobatch.war)
- MapStore war (mapcomposer.war)

Also, you need a version of this products:

- GeoStore
- Http-proxy
- GeoServer
- GeoNetwork

GeoStore

1. Download code from the [GeoStore Github repository](#)
2. Compile and prepare the war for a PostgreSQL deployment:

```
geostore/src$ mvn clean install -Dovrdir=postgres -Ppostgres
```

4. Copy `geostore/src/server/app/target/geostore.war` to `/opt/tomcat_geostore/webapps` folder.

Http-proxy

1. Download code from the [Http proxy Github repository](#)
2. [Configure it](#)
3. [Build http proxy](#):

```
http-proxy/src$ mvn clean install -Dmaven.test.skip
```

4. Copy `http-proxy/http-proxy.war` to `/opt/tomcat_geostore/webapps` folder

GeoServer

1. Download the latest war from [GeoServer downloads page](#)
2. Copy to `/opt/tomcat_geoserver/webapps` folder

OpenSDI-Manager

1. Compile OpenSDI Manager
2. Copy war to `/opt/tomcat_gui/webapps` folder
3. Configure it. Follow the instructions on OpenSDI Manager configuration page.

GeoBatch

1. Compile GeoBatch
2. Copy war to `/opt/tomcat_geobatch/webapps` folder
3. Configure it. Follow the instructions on GeoBatch configuration page.

Mapstore

1. Compile MapStore
2. Copy war to `/opt/tomcat_gui/webapps` folder
3. Configure it. Follow the instructions on MapStore configuration page.

5.3 Update

In this section we're going to learn how to update a production instance. Your start point must be a system with a valid deployment (see install section).

5.3.1 Applications

Note: We recommend to backup each war of each deployment before update it. If you have any problem with the update process, you can restore it and see the error log with the system up.

You can update this applications from the code:

- OpenSDI-Manager
- GeoBatch
- MapStore

OpenSDI-Manager

1. Compile OpenSDI Manager
2. Backup your previous configuration present on `/opt/tomcat_gui/webapps/admin/WEB-INF/classes/opensdi-c`
3. Stop gui seervice (sometimes it isn't necceary):

```
$ service gui stop
```

4. Remove old war and content:

```
$ rm -r /opt/tomcat_gui/webapps/opensdi2-manager**
```

5. Copy war to /opt/tomcat_gui/webapps folder

6. Start system:

```
$ service gui start
```

7. Configure it. Follow the instructions on OpenSDI Manager configuration page. You can restore your older configuration, but be carefull with new options present on the new deployed war comparing the actual /opt/tomcat_gui/webapps/admin/WEB-INF/classes/opensdi-config-ovr.properties with the backed one.

8. Restart again to reload the configuration:

```
$ service gui restart
```

GeoBatch

1. Compile GeoBatch

2. Stop geobatch service:

```
$ service geobatch stop
```

3. Remove old war and content:

```
$ rm -r /opt/tomcat_gui/webapps/geobatch**
```

5. Copy war to /opt/tomcat_geobatch/webapps folder

6. Start system:

```
$ service geobatch start
```

7. Configure it if you need. Follow the instructions on GeoBatch configuration page (older configuration must be used, you only need to configure it if there are changes on your flow configurations)

MapStore

1. Compile MapStore

2. Backup your previous configuration present on /opt/tomcat_gui/webapps/MapStore/WEB-INF/app/static/con

3. Stop gui service (sometimes it isn't necceary):

```
$ service gui stop
```

4. Remove old war and content:

```
$ rm -r /opt/tomcat_gui/webapps/MapStore**
```

5. Copy war to /opt/tomcat_gui/webapps folder

6. Start system:

```
$ service gui start
```

7. Configure it. Follow the instructions on MapStore configuration page. You can restore your old configuration, but you must be careful if there are new options on the new `/opt/tomcat_gui/webapps/MapStore/WEB-INF/app/static/config/mapStoreConfig.js` file and merge it.

5.3.2 Database

If you're going to execute an update, it's recommended to backup the database status before the update. Most updates will be attached as a SQL script.

NRL

1. Backup (as postgres):

```
postgres@server$ pg_dump NRL > path_to_backup.dmp
```

2. Execute the update:

```
pgsql NRL < update.sql
```

Geostore

It isn't common, but:

1. Backup (as postgres):

```
postgres@server$ pg_dump geostore > path_to_backup.dmp
```

2. Execute the update:

```
pgsql geostore < update.sql
```

5.3.3 Custom packs

Note: Be careful with this method, if your update has new configuration parameters, you must merge it. See:

- [OpenSDI-Manager](#) step 7.
 - [MapStore](#) step 7.
-

If you're updating one server with a customized pack, you need to restore your customization after the update. The custom packs are available on the [Github repository](#).

For example, to apply the `pakistan` custom pack:

1. Upload the [Github custom pack content](#) to the server.
2. Open the custom pack:

```
cd custom_pack/pakistan/
```

3. Copy content:

```
cp -R opt/* /opt/
```

4. Change owner to tomcat:

```
chown tomcat:tomcat /opt -R
```

Glossary

dek, dekad The **dekad** is the base time unit of most of the data present on the Crop Information Portal. It is a 10 day period used to approximate the year to a 360 period, with each month is 30 days (3 dekad) long. For example we refer to the 3rd dekad of January we mean the period between Jan 21st and Jan 31st.

dekad in year The *dekad* from the start of the year (from 1 to 36) . i.e. Feb 2nd dekad => 4th dekad in year
Formula: $dek_in_year = MOD(absolute_dek - 1, 36) + 1$

absolute dekad the absolute value of the *dekad* from the year 0. Formula: $absolute_dekad = year * 36 + dekad_in_year$.

GeoCoder, Geocoding Geocoding (sometimes called forward geocoding) uses a description of a location, most typically a postal address or place name, to find geographic coordinates. The GeoCoder of the application use Google and Nominatim services to find the address.

json, JSON JSON (JavaScript Object Notation) is an open standard format that uses human-readable text to transmit data objects consisting of attribute–value pairs. It is the most common data format used for asynchronous browser/server communication (AJAJ), largely replacing XML which is used by AJAX.

CRUD CRUD (Create Read Update Delete) are the four basic functions of a persist storage.

ReST, REST Representational State Transfer (ReST) is the software architectural style of the World Wide Web. (*ref* https://en.wikipedia.org/wiki/Representational_state_transfer)

shape file, shapefile The The shapefile format is a popular geospatial vector data format for geographic information system (GIS) software. It is developed and regulated by Esri as a (mostly) open specification for data interoperability among Esri and other GIS software products. The shapefile format can spatially describe vector features like:

- points
- lines
- polygons

C

- cippak.admin, 47
- cippak.admin.arch, 74
- cippak.admin.arch.model, 78
- cippak.admin.conf, 84
- cippak.admin.conf.geobatch, 122
- cippak.admin.conf.geobatch.createupdatelayer, 122
- cippak.admin.conf.mapstore, 84
- cippak.admin.conf.mapstore.configuration, 85
- cippak.admin.conf.mapstore.csvingestion, 133
- cippak.admin.conf.mapstore.customization, 103
- cippak.admin.conf.mapstore.general_note, 122
- cippak.admin.conf.mapstore.manager, 105
- cippak.admin.conf.mapstore.ndviingestion, 128
- cippak.admin.conf.mapstore.ndvistats, 125
- cippak.admin.crop_agro, 63
- cippak.admin.crop_agro.complete, 71
- cippak.admin.intro, 51
- cippak.admin.use_man, 56
- cippak.admin.workflows, 55
- cippak.glossary, 147
- cippak.maintenance, 133
- cippak.maintenance.compile, 135
- cippak.maintenance.install, 137
- cippak.maintenance.update, 144
- cippak.overview, 3
- cippak.system_config, 1
- cippak.using, 5
- cippak.using.agromet_tool, 34
- cippak.using.crop_data_tool, 15
- cippak.using.crop_status_tool, 33
- cippak.using.gen_tools, 9
- cippak.using.layer_switcher, 10
- cippak.using.legend_panel, 13
- cippak.using.ndvi_tool, 13
- cippak.using.reporting_tool, 38

A

absolute dekad, [149](#)

C

cippak.admin (module), [47](#)
cippak.admin.arch (module), [74](#)
cippak.admin.arch.model (module), [78](#)
cippak.admin.conf (module), [84](#)
cippak.admin.conf.geobatch (module), [122](#)
cippak.admin.conf.geobatch.createupdatelayer (module),
[122](#)
cippak.admin.conf.mapstore (module), [84](#)
cippak.admin.conf.mapstore.configuration (module), [85](#)
cippak.admin.conf.mapstore.csvingestion (module), [133](#)
cippak.admin.conf.mapstore.customization (module),
[103](#)
cippak.admin.conf.mapstore.general_note (module), [122](#)
cippak.admin.conf.mapstore.manager (module), [105](#)
cippak.admin.conf.mapstore.ndviingestion (module), [128](#)
cippak.admin.conf.mapstore.ndvstats (module), [125](#)
cippak.admin.crop_agro (module), [63](#)
cippak.admin.crop_agro.complete (module), [71](#)
cippak.admin.intro (module), [51](#)
cippak.admin.use_man (module), [56](#)
cippak.admin.workflows (module), [55](#)
cippak.glossary (module), [147](#)
cippak.maintenance (module), [133](#)
cippak.maintenance.compile (module), [135](#)
cippak.maintenance.install (module), [137](#)
cippak.maintenance.update (module), [144](#)
cippak.overview (module), [3](#)
cippak.system_config (module), [1](#)
cippak.using (module), [5](#)
cippak.using.agromet_tool (module), [34](#)
cippak.using.crop_data_tool (module), [15](#)
cippak.using.crop_status_tool (module), [33](#)
cippak.using.gen_tools (module), [9](#)
cippak.using.layer_switcher (module), [10](#)
cippak.using.legend_panel (module), [13](#)
cippak.using.ndvi_tool (module), [13](#)

cippak.using.reporting_tool (module), [38](#)
CRUD, [149](#)

D

dek, [149](#)
dekad, [149](#)
dekad in year, [149](#)

G

GeoCoder, [149](#)
Geocoding, [149](#)

J

JSON, [149](#)
json, [149](#)

R

REST, [149](#)
ReST, [149](#)

S

shape file, [149](#)
shapefile, [149](#)