Integration with eazyBI

eazyBI is a powerful add-on for Jira Server and Jira Cloud, providing easy-to-use drag-and-drop creation of custom reports, charts, and dashboard gadgets. It allows you to import your data from Jira Software, Jira Service Desk, Tempo Timesheets, Xray and other popular add-ons. It also enables you to visualize and analyze all your data using many chart types.

The integration with Xray is available in eazyBI's since version 4.2.

Learn more
For more information, please take a look at eazyBI's documentation, which includes some ready-to-use examples.

- How to use
  - Importing Xray's Data
- Examples
  - Overall Test Run results
  - Top Defects report
  - Test Burn-Down / Test evolution chart
  - Test Plan Board
    - Test Runs and defects per folder
    - Test Run Statistics
  - Time Tracking
    - Tests estimated time comparison with elapsed time
    - Compare estimated Test time vs Test Run elapsed time, between Test Executions
  - Requirement coverage
- Report Dashboards
- Learn more

How to use

To take advantage of eazyBI features, you don’t need to do anything in Xray. The integration is done at eazyBI’s side by processing Xray’s internal data; you just need to use an updated version of eazyBI (i.e., 4.2.0 or later).

Importing Xray’s Data

eazyBI will automatically detect if you have Xray installed and give you an additional option in eazyBI’s Jira Import settings. When you enable it, a few dozen new Xray dimensions and measures will be imported.
You’re also able to import custom fields, such as Test Environments. First, you have to configure the custom field mappings as mentioned in eazyBI’s documentation. In order to obtain the custom fields IDs, go to Jira’s administration > Issues > Custom fields.

Examples

Some of these reports may be available when you install eazyBI.

For the specifics of each report, please see eazyBI’s documentation.

Overall Test Run results

<table>
<thead>
<tr>
<th>Test Plan</th>
<th>PASS</th>
<th>TODO</th>
<th>EXECUTING</th>
<th>FAIL</th>
<th>ABORTED</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xray Test Run count</td>
<td>12</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>3</td>
<td>23</td>
</tr>
<tr>
<td>Xray Test Run count %</td>
<td>52%</td>
<td>4%</td>
<td>9%</td>
<td>22%</td>
<td>13%</td>
<td>100%</td>
</tr>
</tbody>
</table>

Xray Test Run count: 23

Top Defects report
Test Burn-Down / Test evolution chart

Test Plan Board

Test Runs and defects per folder

**Aim:** analyse the amount of runs and defects for the Tests contained within each Board's folder.
eazyBI report definition

```json
{
  "cube_name": "Issues",
  "cube_reports": [ {
    "name": "Test Plan Board: Runs and Defects",
    "folder_name": "For Sérgio",
    "result_view": "table",
    "definition": { "columns": { "dimensions": [{ "name": "Measures", "selected_set": ["[Measures].[Xray Tests Run count]", "[Measures].[Distinct reported defect count]"], "members": [] } ], "rows": { "dimensions": [{ "name": "Xray Test Plan", "selected_set": ["[Xray Test Plan].Plan\ calor Members"] }, { "name": "Orphans", "parent_key": "CALC-1021", "expanded": true, "drilled_into": true, "drill_level": ["[Xray Test Plan].Test Plan Board\+]" }, { "name": "CALC-1033", "drillable": true, "key": "CALC-1033", "full_name": "[Xray Test Plan].Test Plan Board\+]" } ] } }, { "columns": { "dimensions": [{ "name": "Measures", "selected_set": ["[Measures].[Xray Tests Run count]", "[Measures].[Distinct reported defect count]"], "members": [] } ], "rows": { "dimensions": [{ "name": "Xray Test Plan", "selected_set": ["[Xray Test Plan].Plan\ calor Members"] }, { "name": "Orphans", "parent_key": "CALC-1021", "expanded": true, "drilled_into": true, "drill_level": ["[Xray Test Plan].Test Plan Board\+]" }, { "name": "CALC-1033", "drillable": true, "key": "CALC-1033", "full_name": "[Xray Test Plan].Test Plan Board\+]" } ] } }, { "columns": { "dimensions": [{ "name": "Measures", "selected_set": ["[Measures].[Xray Tests Run count]", "[Measures].[Distinct reported defect count]"], "members": [] } ], "rows": { "dimensions": [{ "name": "Xray Test Plan", "selected_set": ["[Xray Test Plan].Plan\ calor Members"] }, { "name": "Orphans", "parent_key": "CALC-1021", "expanded": true, "drilled_into": true, "drill_level": ["[Xray Test Plan].Test Plan Board\+]" }, { "name": "CALC-1033", "drillable": true, "key": "CALC-1033", "full_name": "[Xray Test Plan].Test Plan Board\+]" } ] } } ] }
```
Test Run Statistics

Aim: analyse the amount of runs per each Test Run status, per each Board's folder.
Time Tracking

**Tests estimated time comparison with elapsed time**

**Aim:** compare the sum of estimated time on Test issues vs the actual elapsed time of related Test Runs, with the drill-down possibility to Test level.

This report gives the ability to analyse the Tests of a Test Plan, for example, and related runs. It assumes the duration estimate is implemented by defining the Original Estimated Time on Test issues.
eazyBI report definition

```json
{
  "cube_name": "Issues",
  "cube_reports": [
    {
      "name": "Test original estimated time vs Test Runs elapsed time",
      "result_view": "table",
      "definition": {
        "columns": {
          "dimensions": [
            {"name": "Measures", "selected_set": "[Measures].[Xray Test Original estimated hours]", "[Measures].[Xray Test Runs]", "[Measures].[Total of Test Original estimate based on test run count]"},
            "result_view": "table",
            "definition": {
              "columns": {
                "dimensions": [{"name": "Xray Test", "selected_set": "[Xray Test].[All Xray Tests]", "members": [{"name": "All Xray Tests", "full_name": "[Xray Test].[All Xray Tests]"}, {"drillable": true, "type": "all", "expanded": true, "drilled_into": false}, {"depth": 0, "name": "Xray Test", "full_name": "[Xray Test]"}, {"drillable": true, "key": "CALC", "expanded": true, "drilled_into": false}]
              },
              "rows": {
                "dimensions": [{"name": "Xray Test Plan", "selected_set": "[Xray Test Plan].[All Xray Test Plans]"}, {"name": "Xray Test Run Assignee", "selected_set": "[Xray Test Run Assignee].[All Users]"}, {"name": "Xray Test Executed by", "selected_set": "[Xray Test Executed by].[All Users]"}],
                "bookmarked_members": [],
                "current_page_members": 
              }
            },
            "filter_by": {
              "conditions": [{"expression": "[Measures].[Xray Test Original estimated hours]", "operator": ">", "value": "0"}]
            },
            "nonempty_crossjoin": true,
            "pages": {
              "dimensions": [{"name": "Xray Test Plan", "selected_set": "[Xray Test Plan].[All Xray Test Plans]"}, {"name": "Xray Test Run Assignee", "selected_set": "[Xray Test Run Assignee].[All Users]"}, {"name": "Xray Test Executed by", "selected_set": "[Xray Test Executed by].[All Users]"}],
              "bookmarked_members": [],
              "current_page_members": 
            }
          },
          "options": {
            "nonempty": true,
            "view": {"current": "table", "maximized": false, "table": {"show_key": "[Xray Test], [Xray Test Execution]"}}
        }
      }
    }
  ],
  "calculated_members": {
    "name": "Xray average Test Run duration", "format_string": "MinutesFormatter",
    "formula": "{[Measures].[Xray Test Run duration] / 60", "annotations": {"group": "Xray", "drill_through_dimension_levels": "[Xray Test].[Test], [Xray Test Execution].[Execution]"},
    "definition": "Xray Test Original estimated hours", "dimensions": "[Measures].formula:" Sum(n)
  -- set of test which are related report context and has at least one execution
  Filter(n Descendants([Xray Test].CurrentHierarchyMember, [Xray Test].[Test] & [Measures].[Xray Tests with executions] > 0 n ),
      n -- for each test get Original estimate
  DefaultContext(n [Issue].[Issue].GetMemberByKey([Xray Test].CurrentHierarchyMember.Key),
    [Measures].[Original estimated hours] n )
  n --transform hours to minutes for formatting
  + 60, "format_string": "MinutesFormatter"),
    "name": "Xray Total Test run duration", "dimension": "Measures", "formula": "Sum(n -- set of test which are related to report context and has at least one execution
  Filter(n Descendants([Xray Test].CurrentHierarchyMember, [Xray Test].[Test] & [Xray Tests with executions] > 0 n ),
      n -- for each test calculate total execution time
  [Measures].[Xray average Test Run duration] * [Measures].[Xray Test Runs] n )", "format_string": "MinutesFormatter"),
    "name": "Total of Test Original estimate based on test run count", "dimension": "Measures", "formula": "[Measures].[Xray Test Original estimated hours]* [Measures].[Xray Test Runs] n", "format_string": "MinutesFormatter"
  }
}

Compare estimated Test time vs Test Run elapsed time, between Test Executions

Aim: compare the estimated time on Test issues vs the actual elapsed time of the corresponding Test Run, on each Test Execution.

This report gives the ability to analyse the Tests and the related Test Executions of a Test Plan, in order to check how the elapsed time of Test Runs evolve from Test Execution to Test. It assumes the duration estimate is implemented by defining the Original Estimated Time on Test issues.
eazyBI report definition

```json
{
  "cube_name": "Issues",
  "cube_reports": [ {
    "name": "Compare estimated Test time vs Test Run elapsed time, between Test Executions",
    "result_view": "table",
    "definition": {
      "columns": {
        "dimensions": [{
          "name": "Xray Test Execution",
          "selected_set": ["Xray Test Execution", [All Xray Test Executions]]
        }],
        "members": [ {
          "depth": 0,
          "name": "All Xray Test Executions",
          "full_name": "[Xray Test Execution].[All Xray Test Executions]",
          "drillable": true,
          "type": "all",
          "expanded": true,
          "drilled_into": false,
          "depth": 1,
          "name": "Calculator",
          "full_name": "[Xray Test Execution].[Calculator]",
          "drillable": true,
          "key": "CALC",
          "expanded": true,
          "drilled_into": false,
          "removed": true,
          "parent_full_name": "[Xray Test Execution].[All Xray Test Executions]"
        }],
        "bookmarked_members": []
      },
      "measures": {
        "selected_set": ["Xray Test Original estimated hours", "Xray average Test Run duration", "Xray Total Test run duration"],
        "members": []
      }
    }
  }
],
  "options": {
    "nonempty": true,
    "view": {
      "current": "table",
      "maximized": false,
      "table": {
        "show_key": ["Xray Test"]
      }
    }
  },
  "calculated_members": {
    "dimension": "Measures",
    "name": "Xray average Test Run duration",
    "format_string": "MinutesFormatter",
    "formula": " Measure.([Xray Test Run duration] / 60) "
  },
  "drill_through_dimension_levels": ["Xray Test", "Xray Test Execution", "Execution"],
  "name": "Xray Test Original estimated hours",
  "dimensions": ["Measures", "formula": "Sum(\n    -- set of test which are related to report context and has at least one execution\n    Filter(\n      Descendants([Xray Test].CurrentHierarchyMember,[Xray Test].[Test]),\n      [Measures].[Xray Tests with executions] > 0\n    )\n    DefaultContext((\n      [Issue].[Issue].GetmemberByKey([Xray Test].CurrentHierarchyMember.Key),\n      [Measures].[Original estimated hours]\n    )) \n    -- transform hours to minutes for formatting\n    * 60,\n    "format_string": "MinutesFormatter"),\n    "name": "Xray Total Test run duration",
    "dimension": "Measures",
    "formula": "Sum(\n      -- set of test which are related to report context and has at least one execution\n      Filter(\n        Descendants([Xray Test].CurrentHierarchyMember,[Xray Test].[Test]),\n        [Measures].[Xray Tests with executions] > 0\n      )\n      DefaultContext((\n        [Issue].[Issue].GetmemberByKey([Xray Test].CurrentHierarchyMember.Key),\n        [Measures].[Xray Tests with executions] > 0\n      ))\n      -- for each test calculate total execution time\n      [Measures].[Xray average Test Run duration] * [Measures].[Xray Test Runs]\n    )"\n  }
}
```

Requirement coverage
Aim: provide a quick overview of coverage per requirement: show requirements, their workflow status, amount of Tests covering each requirement, amount of Runs created and amount of defects created and their status. More info [here](#).

eazyBI report definition

```json
{
  "cube_name": "Issues",
  "cube_reports": [{
    "name": "Xray Requirement coverage",
    "result_view": "table",
    "definition": {
      "columns": {
        "dimensions": [
          "Measures.Xray Requirement status",
          "Measures.Xray Tests created",
          "Measures.Xray Tests with executions",
          "Measures.Xray Test Runs",
          "Measures.Xray Test Run Summary",
          "Measures.Xray Tests defect count",
          "Measures.Xray Defect list"
        ],
        "members": []
      },
      "rows": {
        "dimensions": [
          "Xray Requirement",
          "Measures.Xray Requirement",
          "Measures.Xray Requirement status",
          "Measures.Xray Tests created",
          "Measures.Xray Tests with executions",
          "Measures.Xray Test Runs",
          "Measures.Xray Test Run Summary",
          "Measures.Xray Tests defect count",
          "Measures.Xray Defect list"
        ],
        "members": []
      }
    },
    "calculated_members": [{
      "dimension": "Measures",
      "name": "Xray Requirement status",
      "formula": 
      "[Status].[Status].getMemberNameByKey([Xray Requirement].CurrentHierarchyMember.get('Status ID'))",
      "annotations": {
        "group": "Xray",
        "drill_through_dimension_levels": "[Xray Test].[Test].[Xray Test Execution].[Execution]",
        "dimension": "Measures.Xray Test Run summary",
        "format_string": "",
        "formula": "Sum(\n          Descendants([Xray Test Execution Status].CurrentMember, [Xray Tests with executions] > 0)\n          .\n          [Measures.Xray Tests Run count]"
      }
    }
  }
},
```
```json
```
Report Dashboards

eazyBI supports the creation of dashboards that you can use to combine several Xray-related reports in one display.

You can also use eazyBI’s reports and dashboards in Jira dashboards by using proper gadgets.

Learn more

For specific questions concerning eazyBI, please use eazyBI support channel.

Please refer to the following resources for more information:

- eazyBI’s specific documentation for Xray
- eazyBI's blog post detailing integration with Xray, including usage examples
- eazyBI’s Xray Test burnup for Test Plan example
- Example: Requirement Coverage
- Xray's blog post