Final Prototype of Compliance Governance Dashboards

Project acronym: COMPAS

Project name: Compliance-driven Models, Languages, and Architectures for Services

Call and Contract: FP7-ICT-2007-1

Grant agreement no.: 215175

Project Duration: 01.02.2008 – 28.02.2011 (36 months)

Co-ordinator: TUV Technische Universitaet Wien (AT)

Partners: CWI Stichting Centrum voor Wiskunde en Informatica (NL)

UCBL Université Claude Bernard Lyon 1 (FR)

USTUTT Universitaet Stuttgart (DE)

TILBURG UNIVERSITY Stichting Katholieke Universiteit Brabant (NL)

UNITN Universita degli Studi di Trento (IT)

TARC-PL Telcordia Poland (PL)

THALES Thales Services SAS (FR)

PWC Pricewaterhousecoopers Accountants N.V. (NL)

This project is supported by funding from the Information Society Technologies Programme under the 7th Research Framework Programme of the European Union.
Project no. 215175

COMPAS

Compliance-driven Models, Languages, and Architectures for Services

Specific Targeted Research Project
Information Society Technologies

Start date of project: 2008-02-01  Duration: 36 months

D5.5 Final Prototype of Compliance Governance Dashboards

Revision 2.0

Due date of deliverable: 2010-12-31
Actual submission date: 2010-12-29

Organisation name of lead partner for this deliverable:
UNITN – University of Trento, Italy

Contributing partner(s):
PWC – Price waterhouse coopers Accountants N.V., Netherlands
TARC-PL – Telcordia Poland
THALES – Thales Services SAS
TUV – Technische Universitaet Wien
UCBL – Université Claude Bernard Lyon 1

<table>
<thead>
<tr>
<th>Dissemination Level</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PU</td>
<td>Public</td>
</tr>
<tr>
<td>PP</td>
<td>Restricted to other programme participants (including the Commission Services)</td>
</tr>
<tr>
<td>RE</td>
<td>Restricted to a group specified by the consortium (including the Commission Services)</td>
</tr>
<tr>
<td>CO</td>
<td>Confidential, only for members of the consortium (including the Commission Services)</td>
</tr>
</tbody>
</table>
History chart

<table>
<thead>
<tr>
<th>Issue</th>
<th>Date</th>
<th>Changed page(s)</th>
<th>Cause of change</th>
<th>Implemented by</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>2009-08-10</td>
<td>All sections</td>
<td>New document</td>
<td>UNITN</td>
</tr>
<tr>
<td>0.2</td>
<td>2009-08-10</td>
<td>All sections</td>
<td>Updates</td>
<td>UNITN</td>
</tr>
<tr>
<td>0.3</td>
<td>2010-11-08</td>
<td>All sections</td>
<td>Create a concise version of the document</td>
<td>UNITN</td>
</tr>
<tr>
<td>0.4</td>
<td>2010-12-15</td>
<td>All sections</td>
<td>Reduce the size of the document and applying comments collected during our internal review</td>
<td>UNITN</td>
</tr>
<tr>
<td>0.5</td>
<td>2010-12-20</td>
<td>Specific sections</td>
<td>Updated the text according to the Thales’ review</td>
<td>UNITN</td>
</tr>
<tr>
<td>0.6</td>
<td>2010-12-22</td>
<td>Section 5</td>
<td>Update to include a conclusion</td>
<td>UNITN</td>
</tr>
<tr>
<td>1.0</td>
<td>2010-12-24</td>
<td></td>
<td>Approval</td>
<td>TUV</td>
</tr>
</tbody>
</table>

Authorisation

<table>
<thead>
<tr>
<th>No.</th>
<th>Action</th>
<th>Company/Name</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Prepared</td>
<td>UNITN</td>
<td>2010-12-24</td>
</tr>
<tr>
<td>2</td>
<td>Approved</td>
<td>TUV</td>
<td>2010-12-24</td>
</tr>
<tr>
<td>3</td>
<td>Released</td>
<td>TUV</td>
<td>2010-12-29</td>
</tr>
</tbody>
</table>

Disclaimer: The information in this document is subject to change without notice. Company or product names mentioned in this document may be trademarks or registered trademarks of their respective companies.

All rights reserved.

The document is proprietary of the COMPAS consortium members. No copying or distributing, in any form or by any means, is allowed without the prior written agreement of the owner of the property rights.

This document reflects only the authors’ view. The European Community is not liable for any use that may be made of the information contained herein.
Contents

1. Introduction ............................................................................................................................ 6
  1.1. Purpose and scope ........................................................................................................... 6
  1.2. Document overview ........................................................................................................ 7
  1.3. Abbreviations and acronyms ........................................................................................... 7
2. User Guide for the CGDs and the Root Cause Analysis Prototypes ...................................... 8
  2.1. Compliance Governance Dashboards ............................................................................. 9
  2.2. Root Cause Analysis ..................................................................................................... 13
3. Implementation Details ........................................................................................................ 15
4. Source Code and Additional information ............................................................................. 17
5. Conclusion ............................................................................................................................ 18
6. Reference documents ........................................................................................................... 18
  6.1. Internal documents ........................................................................................................ 18
  6.2. External documents ....................................................................................................... 18

List of figures

Figure 1 Compliance governance runtime architecture, an extension of [D5.3] ................... 7
Figure 2 Compliance Monitoring Tools Web Portal of the COMPAS Project ...................... 8
Figure 3 Compliance Home page of the CGD with graphical widgets for the visualization of indicators and the compliance drill-down table ................................................................. 9
Figure 4 Compliance Sources by Activity page with activity specific indicators and activity-level compliance info ............................................................................................................... 10
Figure 5 Compliance Policy by Activity page with activity specific indicators and activity-level compliance info ................................................................................................................. 11
Figure 6 Compliance Requirement by Activity page with activity specific indicators and activity-level compliance info ................................................................................................... 12
Figure 7 Low-level events (violations) with details associated to the Composition Permission Compliance Requirement ...................................................................................................................... 12
Figure 8 Feature Selection Page of the Decision Tree Solution for Root Cause Analysis .. 14
Figure 9 Decision tree computed over instances of the WatchMe business process .......... 14
Figure 10 Web application involving the implementation of the CGD and the Root Cause Analysis 15
Abstract

Assessing whether a company’s business processes conform to laws and regulations and follow standards and best practices, i.e., compliance governance, is a complex and costly task. Few software tools aiding compliance governance exist; however, they typically do not really address the needs of who is actually in charge of assessing and controlling compliance, that is, compliance experts and auditors.

In order to support compliance governance we advocate the use of Compliance Governance Dashboards (CGDs), whose design and implementation is however challenging for at least three reasons: (i) it is fundamental to identify the right level of abstraction of compliance performance to be shown; (ii) it is not trivial to visualize different analysis perspectives; and (iii) it is difficult to manage the large amount of involved concepts, instruments, and data.

All these issues are properly identified and documented in On the Design of Compliance Governance Dashboards for Effective Compliance and Audit Management accepted at the NFPSLAM-SOC'09 [SRC+09] and Analyzing Compliance of Service-Based Business Processes for Root-Cause Analysis and Prediction accepted at the 1st Workshop on Engineering SOA and the Web (ESW'10) in conjunction with ICWE'10 [RSD+10], as well as in the book chapter called Aiding Compliance Governance in Service-Based Business Processes published in the Handbook of Research on Non-Functional Properties for Service-Oriented Systems: Future Directions [SRA+10]. Together these references show which concepts and models underlie the problem, and, eventually, how IT can effectively support compliance monitoring and analysis in Service-Oriented Architectures (SOAs).

In this deliverable we simply present a user guide for CGDs and Root Cause Analysis prototypes, as well as their implementation details CGDs and how to install them.
1. Introduction

*Compliance* is a term generally used to refer to the conformance to a set of laws, regulations, internal policies, SLA, standards, or best practices (compliance sources according to COMPAS conceptual model [D7.1]). *Compliance governance* refers to the set of procedures, methodologies, and technologies put in place by a corporation to carry out, monitor, and manage compliance.

Compliance governance is an important, expensive, and complex problem to deal with as described in [SRC+09]. To address these and similar compliance problems, COMPAS proposes a conceptual model for compliance [D7.1] and for CGDs, along with a dashboard architecture and a prototype implementation. The aim of CGDs is to report on compliance, to create an awareness of possible problems or violations, and to facilitate the identification of root causes for non-compliant situations. In addition, COMPAS also proposes a root cause analysis solution, based on decision trees, in order to discover the causes of the violations and predict their future occurrences, when possible.

The dashboard addresses several classes of profiles: chief officers of a company, Business Line managers, internal auditors, and external auditors (certification agencies). These two latter typically focus on a fairly narrow set of processes and examine historical data to verify non-compliant situations and how they have been dealt with. Via the dashboard, they also have *access to key compliance indicators (KCIs)* [D5.4] defined for each process. Managers (especially high-level ones) are interested in a much broader set of compliance regulations and at *quasi-real* time compliance information that allows them to detect problems as they happen and identify the causes, so that they can correct them before they become (significant) violations. Compliance stakeholders have access and navigate through the entire set of compliance sources, business processes, and business units and also observe the overall compliance status (through aggregate KCIs). In addition, once problems are identified (unsatisfactory values for indicators) they drill-down to the low-level events (violation), as well as use the root cause analysis solution to indentify the problem or predict future non-compliance behaviors.

In that context, this deliverable describes how to use CGDs and Root Cause Analysis prototypes, as well as their implementation details and complementary information to download and install them.

1.1. Purpose and scope

To better understand the purpose and scope of this deliverable let us consider Figure 1, which shows a high-level view of the COMPAS compliance governance runtime architecture (part of the overall architecture described in [DA.1]) and helps us to point out the role of CGDs and root cause analysis and their interactions with other architecture components.

In such an architecture (Figure 1) we can see that all events from different fonts (i.e., Business process engine, Complex Event Processing, and Business protocol monitoring) are published in the ESB (Enterprise Service Bus), to afterwards be stored in the Event log. After that, an extraction, transformation and load (ETL) routine is executed to capture those raw data events and to store them into the Data warehouse (DW) [D5.3]. That done, the events conformed to a data model can be used as input for the Analysis/Business Intelligence component that shows their final results in the Compliance governance Web UI. Such component is in charge of computing KCIs, showing CGDs, and providing root cause analysis of violations using decision tree algorithms.
Given that, the CGDs and the Root Cause Analysis make part of the Compliance governance Web User Interface (UI). They are the presentation components used to do offline and on-line runtime compliance monitoring, showing violations, as well as KCIs. Both, KCIs and low-level violations take into account different summarization levels (granularity of business executions, i.e., process and activities) and analysis perspectives (i.e., business process or compliance). This allows different user profiles (internal and external auditors, compliance experts, CIO) to monitor the compliance status of an organization, starting from indicators (high-level) until violation instances. Yet, the users also can build decision trees in order to discover the causes of the non-compliant behaviors. More details about such compliance visualisation are provided in this deliverable (Section 2). The Complex event processing and Business protocol monitoring, which result in runtime monitoring, were respectively provided by UBCL and Telcordia in the [D5.4]. The Log mining component was developed by UCBL and its results are documented in the [D5.6].

1.2. Document overview

This deliverable starts explaining briefly the complexities and challenges faced in order to provide CGDs, as well as, the role of dashboards in COMPAS. Then, the remaining sections are as follows: Section 2 describes how to use the CGDs and the Root Cause Analysis prototypes. Section 3 describes their implementation details. Section 4 introduces how to obtain and install such prototypes.

1.3. Abbreviations and acronyms

<table>
<thead>
<tr>
<th>CGD</th>
<th>Compliance Governance Dashboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIO</td>
<td>Chief Information Officer</td>
</tr>
<tr>
<td>DoW</td>
<td>Description of Work</td>
</tr>
<tr>
<td>DW</td>
<td>Data Warehouse</td>
</tr>
<tr>
<td>IT</td>
<td>Information Technology</td>
</tr>
<tr>
<td>KCI</td>
<td>Key Compliance Indicator</td>
</tr>
</tbody>
</table>
2. User Guide for the CGDs and the Root Cause Analysis Prototypes

In this section we present and explain how to use the COMPAS’ CGDs and the Root Cause Analysis prototypes. Both respectively are provided and accessed by the Telecom CGD, Banking CGD and Root Cause Analysis tabs in the COMPAS Compliance Monitoring Tools web portal (Figure 2) and available at http://compas.disi.unitn.it:8080/CGDs/main.jsp. This same portal also supports the Compliance Request Language Tools (CRLT) for design-time compliance monitoring and the Complex Event Processing (CEP) engine for runtime compliance monitoring. The CRLT does not make part of this deliverable, it was developed by the Tilburg University and is documented in D2.6. Regarding the CEP, it shows near to real time compliance violations to allow on-line runtime monitoring of business process and activities. Such visualization tool is an additional development effort, since it was not initially specified in the DoW, and its design and implementation descriptions are introduced D1.4 and D6.2.
2.1. Compliance Governance Dashboards

To provide the look-and-feel idea of what we have been implemented, the sequence of figures presented below illustrates screenshots from our CGD prototype (Figures 3 to 7). Such sequence is disposed according to the navigation model proposed in [SRC+09], starting from the highest-level information (Compliance Home page) until the lowest-level (Compliance Rules Violations page). It is important to highlight that here we just show screenshots from the WatchMe scenario (Telecom CGD) in order to avoid replications since the same structure presented here is applied to the THALES scenario (Banking CGD). To visualize the CGD of the THALES scenario please visit the COMPAS Monitoring web portal.

The CGD web pages are composed of three colored indicators (top left), in which the highest level of compliance performance status of each perspective is highlighted. Near to the colored indicators are located the trend line charts, where are shown the compliance performance during the last period (e.g., the last five months). Hence, for example, the project manager can see if his past actions impact beneficially or harmfully the general compliance performance (e.g., the enrolment of new employees decreased the performance in the first month, than after the compliance introductory courses such problem was solved). The gray indicators (top right) also report on the compliance, but taking into account the two main policies present in the WatchMe scenario (i.e., Licensing and Qos policies). It is worth to mention that all the indicators are organized in a descended compliance performance order, thus the worst and most risky performances are always located at the top. At the bottom of the page, there is the interactive compliance drill-down table containing the compliance performance according two main perspectives: compliance (columns) and business (rows). The content of such table, as well as the indicators, change every time the users navigate on the different perspectives as will be presented along this section. The Time Scale (top left) also updates the content of the CGDs, using it the users can navigate thought the time searching for past performances, or grouping the information according to their needs (e.g., monthly results, year results). The examples listed here consider the complete data set corresponding to the 2010th year.

![Compliance Governance Dashboard](image.png)

**Figure 3**  Compliance Home page of the CGD with graphical widgets for the visualization of indicators and the compliance drill-down table

In each page of the CGD, the amount and the position of the graphical widgets for the indicators, the tables, the summaries, and so on are carefully chosen in accordance with our
short-term memory and the convention of most western languages that are read from left to right and from top to bottom [Few06].

Now that we are aware of the main components of the CGDs, let’s start with the navigation details. In order to navigate through the difference analysis perspectives and granularity levels the users may use the interactive table simply clicking on its cells. In that way, it is possible to do drill-down (clicking the + symbol in the table) or roll-up (clicking the – symbol). When the lowest level of perspectives is reached (i.e., compliance requirement X business process activity) the click in the cells conduces to the violation web page that is further presented. The colours of the table cells follow the same taxonomy adopted to build the indicators. Red and yellow cells always lead to violations (low-level events listed in the violations page), and green and white cells just show details about the compliance requirement and business activity. It happens, because green and white cells do not have violations associated to them. The white cell means that none compliance requirement is associated to the process or activity. It is worth to notice that the compliance perspective follows the compliance conceptual model of the project, in which a compliance source is composed by requirements that could be groups in one or more compliance policies. The sequence of pages presented in the next paragraphs is a navigation example and is consistent with the WebML model introduced in [SRC+09].

The main and initial page of the CGD is the Compliance Home page, which concentrates on the most important information at a glance, condensed into just one page (see Figure 3). It represents the highest granularity of information, in which the users can see the compliance of the sources and business processes combinations. For example, the WatchMe processes are 76% compliant regarding the VideoSport License, which is considered a weak performance and is highlighted in red. Hence, in order to see more details about this non-compliant behavior the users can navigate to a page with lower level information, using the table (more specifically clicking in the WatchMe Business Process cell), and see which activities of the-WathMe process have the worst performance.

![Figure 4 Compliance Sources by Activity page with activity specific indicators and activity-level compliance info](image)

The action described above lead the users to the ComplianceSouce by Activity page (Figure 4). Such page condenses lower level information concerning a combination of all the activities of
the WatchMe Business Process (Business Process Activities in the third column) and the all the Compliance Sources. At this level the colour indicators show the compliance status of the activities with the worst compliance performance level. The gray indicators continue showing the compliance status of the Licensing and QoS policies, but taking into account the activity level to compute them. In the example illustrated here we cannot see this difference of values because the WatchMe scenario just has one process, for that reason the values are still the same in both levels. At that level we can see that Receive stream from audio and video provider activity is the only activity evaluate by the three compliance sources SportVideoSport License, FootbalGames License, and Consumer Contract. Hence, in order to investigate more details about the weak performance of this activity, users can click in the cells containing compliance sources. As a result they will see the Compliance Policy by Activity page (Figure 5), where the compliance status of the activities are grouped by policies. In the WatchMe scenario we just deal with two policies QoS and Licensing, however the table was developed to support bigger numbers. It is important to mention that the compliance policy view is also reached from the Compliance Home page (Figure 3), just clicking the compliance perspective.

![Compliance Governance Dashboard](image)

**Figure 5** Compliance Policy by Activity page with activity specific indicators and activity-level compliance info

By clicking in the policy cells (Figure 5) the users will see the compliance requirements that are grouped in each policy at the Compliance Source by Activity page (Figure 6). In the WatchMe scenario the licensing policies contain just one requirement named Composition permission. On the other hand, the QoS policy is composed of the Availability, Response Time and Delivery rate requirements. Due to size limitations we cannot show the complete view of the table here, for more details you can visit the COMPAS web portal. When the table shows compliance requirements as columns and activities as rows, it reached its lowest level of information. From here users can select a combination of a requirement and activity, intersection cell of a compliance requirement and a business process activity, in order to explore all the violations associated to it. For example, clicking the in 23% value, which means the intersection between Composition Permission and Receive stream video and audio the users can discover all the low-level events (violations) associated to this weak performance.
Figure 6  Compliance Requirement by Activity page with activity specific indicators and activity-level compliance info

Figure 7 presents the all the event violations associated to the compliance requirement and business activity previously selected in the interactive table. The top of the page contains details about the business process (i.e., unit, name, UUID, activity, and activity UUID of the process), the compliance requirement (i.e., source, name, policy, and description of the compliance requirement) and the execution (i.e., total of events, total of violations and average of the compliance level). UUID is a Universally Unique Identifier, which uniquely identifies a model or a model element (see [HZD09]).

At the bottom, the page lists the most relevant content of the events. Each row of the table represents a distinct violation and the columns contain the typical information required by auditors, e.g., process instance, activity instance, key of the event, type of the event, source of the event, and timestamp. When available additional information is also presented (e.g., video and audio providers, stream name and language of the stream downloaded in the WatchMe scenario). In order to visualize such information the users need to click the More... link at the
end of the rows. In our example, the table shows all the violations associated to the Composition Permission and Receive stream from video and audio providers activity. Such requirement states: “Football Games can only have audios streams from AudioSport or SportingAudio. VideoSport can only have audio streams from AudioSport”. The violation in the first line occurred due to the non-compliant integration of the FiFa WorldCup Final 1992 video stream, provided by the VideoSport provider, with an audio stream distributed by the FootballGames Audio provider. This same behavior was detected in 24% of the cases, 26 events, from the 34 executed in the total.

2.2. Root Cause Analysis

The root cause analysis in the COMPAS project is implemented as decision trees. In this subsection we show how to build such a tree and how it correlates business and compliance data of executions of the WatchMe business process. Hence, based on that it will be possible to understand how business parameters are associated with the fulfillment (or not) of the company's compliance requirements.

As in any decision tree, internal nodes represent the attributes on which split points are defined to draw the various branches from the root node to the leaf nodes of the tree. On the other hand, leaf nodes represent the class attribute and it tells us to which class the object in study should be classified. In our case, the objects of study are tuples that contain data corresponding to instantiations of business process executions.

To build the tree users need to select some parameters, as depicted in Figure 8. First of all, it is required to choose the compliance target to be analyzed (e.g., a business process or an activity). Then, users need to define a data time interval in which we want to analyze the data. After that, the business data used to build the tree need to the selected, which will constitute the internal nodes of the tree. The predefined attributes listed by the tool are the parameters (columns) retrieved from the F_BPInstance table stored in the DW (Figure 5 in [SRA+10]). In our example, business data are represented by the following attributes:

- **Denial of Service**: counts the number of requests that were not fulfilled because of performance issues or any other factor that avoids the fulfillment of clients' requests.
- **Service_Type**: indicates the type of service that is being requested.
- **Throughput**: is the average number of requests that were fulfilled with success according to the establish compliance requirement.

Finally, the user needs to select a class attribute, which we expect to be one of the compliance data (note, however, that our tool does not restrict the user to select only compliance data as class attributes). In our example, the compliance data are represented by the following attributes:

- **Licensing**: corresponds to policies related to license issue. Whenever an instance of the process violates one of the terms of the license, this policy is violated and thus Compliance(Licensing) = False (otherwise, Compliance(Licensing)=True).
- **Internal_Policy**: this refers to a set of policies that needs to be checked and, when violated, affects negatively the revenue of the company. In our example, part of the internal policy consists in checking "access by unauthorized users". Whenever an unauthorized user accesses the service, the evaluation is True, otherwise, it is evaluated as False.
By selecting the parameters as illustrated in Figure 8, we obtain the decision tree depicted in Figure 9. In this tree, the internal nodes are composed by the attributes Denial_of_Service, Internal_Policy, Service_Type, and Throughput. Licensing, instead, is the class attribute for the decision tree. This is, we would like to understand how the former attributes are related to or might affect Licensing.

*Figure 8  Feature Selection Page of the Decision Tree Solution for Root Cause Analysis*

From the resulting decision tree (Figure 9), we can see that we have three types of services, namely, audio streaming, packet data and video streaming all defined by an IT expert. For packet data, we can see that Licensing policies were always (or "most of the time") fulfilled,
thus, regardless of the rest of the attributes, Packet data is licensing-compliant for the set of process instances in study. Audio streaming, however, is somehow related to Denial_of_Service and the estimated threshold for being compliant or not is 75. Finally, video streaming licensing-compliance is related to the Internal_Policy attribute, and we can see that when this attribute is False ("Access by unauthorized users") we are licensing-compliant, otherwise, we are not (and this makes sense as, typically, a user cannot access the service if he/she does not have a valid license).

Finally, it is worth to mention that the tree we presented in this section has been built on synthetic data, which we have generated manually for the purpose of presenting the approach in this deliverable with an example that can be easily understood. However, the results that we have shown here are real, in the sense that they have been computed on our synthetic data using the running version of our prototype and data warehouse that are already deployed in COMPAS’s test bed.

3. Implementation Details

The CGD is set on the top of a data DW (optimized for reporting purposes) that implements the conceptual model described in D5.3. It is however important to recognize that this does not affect the logic behind the designed navigation structure [SRC+09], which represents a best practice for the rendering of compliance information to auditors, according to the experience by the industrial partners involved in the project.

The front-end of the CGD is an interactive web application implemented according to Figure 10 and composed of JSP pages. Dashboard.jsp is responsible for loading the dashboard widgets and the interactive table as illustrated in the previous section in Figure 3 to 6. Each
time this page is loaded, i.e., at the first access or when a new time scale is selected, it invokes the `widgets.java` class, which runs a set of pre-defined, parameterized SQL queries over the DW, filled with variable input parameters (e.g., time scale and business level) representing user selections. Listing 1, for example, shows a typical query used to retrieve the necessary data to render indicators in their colorful or gray UI widgets. These graphical representations of indicators are implemented using Google Charts (http://code.google.com/apis/chart/), which require a predefined invocation URL as listed in Listing 2.

```sql
SELECT Req.Source_Name AS Name,
      ROUND(100*(Count(*)-Sum(Proc_Inst.CausedViolation))/Count(*)) AS Value
FROM F_BPInstance Proc_Inst JOIN F_ComplianceEvaluation Evaluation
     JOIN D_ComplianceRequirement Req
     JOIN D_Calendar_Date dc
WHERE D_Calendar_Date.System_Date >= ?('startDate', 'DD/MM/YYYY') AND
     D_Calendar_Date.System_Date < DATE_ADD(?,'INTERVAL 1 DAY')
GROUP BY Req.Source_Name ORDER BY Value ASC;
```

Listing 1  Percentage of non-compliance for a given sources in a pre-defined time interval

![Google Chart invocation URL](http://chart.apis.google.com/chart?cht=lc&chs=140x30&chbh=0,0,0&chd=t:100|0,100,100,100&chco=00000000&chxt=x&chxs=0,000000,0,0,t&chxtc=0,0&chf=c,lg,0,e6e6e6,0,a4a4a4,0.80,6e6e6e,0.95,000000,1&chs=0,9,0,t,000000|1,,9,0,0,t,000000&chxp=0,0,80,95&chm=V,FFFFFF,1,0.8,2|V,FFFFFF,1,0.95,2|V,000000,1,0.73,7,1>

Listing 2  Line of code to create and invoke a Google chart

When loading the dashboard into the client browser, also the content of the interactive table is updated according to the same parameters and the functions contained in `table.java`. In addition, the rows of the interactive table are also refreshed when a new business or compliance perspective is selected, e.g., via a mouse click. This table is fed with the DW data based on two SQL queries, one that brings all the data associated with business process and the other with business process activities. In both cases, the results of the query are locally stored on the client side as hash table objects, which allow the CGDs to dynamically show or hide their content in an HTML table according to the selected business and compliance perspectives.

The `Violation.jsp` page shows the lowest level information (events) and can be accessed through the interactive table, more specifically via a mouse click on one of the coloured cells of the interactive table rendering the intersection of a business process activity and a compliance requirement. White cells correspond to activity-requirements combinations that do not have compliance requirements to be checked, thus no events associated to them are stored in the DW. The business process activity and the compliance requirement of the clicked cell are passed as parameters to the classes of the `violation.java` class that connects to the DW and retrieves the respective events. Similar to the previous approach, the `Violation.jsp` file also adopts an SQL query as depicted in Listing 3.
SELECT
    Sum(IsViolation) AS Violations,
    Count(*) AS Total,
    ROUND(100*(Count(*)-Sum(IsViolation))/Count(*)) AS KCI
FROM
    F_BPInstance Proc_Inst JOIN F_ComplianceEvaluation Evaluation
    ON Proc_Inst.BPInstance_Key = Evaluation.BPExecutionKey
    JOIN D_ComplianceRequirement Req
    ON Req.ComplianceRequirement_Key = Evaluation.ComplianceRequirement_KeY
    JOIN D_Calendar_Date
    ON Evaluation.Calendar_Date_Key = D_Calendar_Date.Calendar_Date_Key
WHERE Evaluation.Service_Key is null AND
    Evaluation.BP_Key = ? AND
    Evaluation.BPActivity_Key = ? AND
    Evaluation.ComplianceRequirement_Key = ? AND
    System_Date >= ? AND
    System_Date < DATE_ADD(?, INTERVAL 1 DAY);

Listing 3 SQL query used to retrieve all the low-level events associated to a specific compliance requirement and business process activity

The decision tree page guides the user in the process of decision tree mining for root cause analysis and prediction. In this process, the page helps users in choosing the parameters of interest for building the decision tree. More precisely, it helps users in selecting the attributes for the internal nodes, the class attributes to which tuples will be classified, the time range used for filtering process instances, among other parameters of the algorithm used for the mining task. Once these parameters have been selected, DecTree.jsp processes the instances from the process instance table (stored in the DW) in order to prepare the training tuples that will be used to feed the mining algorithm. The concrete algorithm used for building the decision tree is J48 and we rely on Weka [HFH+09] as the supporting library for the whole mining process as it is a well-known and mature data mining library. Note, however, that we can choose any other decision tree mining algorithm among the ones available in Weka. We choose J48, which is an open source implementation of the C4.5 algorithm, due to its popularity and the fact that this is a well-understood algorithm for building decision trees. In addition, it is equipped with features such as continuous number categorization and missing value handling, which are key features when doing classification on data coming from realistic settings.

All pages of the CGD were developed using Eclipse Java EE IDE for Web Developers, and the DW runs on Oracle 11g (first version) and MySQL 5.1 (second version). An example of the CGD web portal as used in the Compas project is available at http://compas.disi.unitn.it:8080/CGDs/main.jsp selecting the CGD tab. From the same start page it is also possible to access the decision tree component by selecting the Root Cause Analysis tab.

4. Source Code and Additional information

The CGDs code is released as an open source project under the GNU Affero General Public License (AGPLv3) 2. In order to actually reach the open source community, we made the application available at the well-known open source community, Source Forge [SF10]. More specifically, the source is available at https://cgd.svn.sourceforge.net/svnroot/cgd/ using the
following credentials login: guest and password: compas. The code is organized in two separate folders:
- **CGDs** folder contains the whole dashboards code
- **DB** stores the data warehouse data model schema that is used as input for the dashboards.

In each folder there are text files containing additional details in order to install and configure both prototypes.

The CGDs prototype is deployed at http://compas.disi.unitn.it:8080/CGDs/main.jsp together with other compliance governance monitoring tools (e.g., Complex Event Processing, CTRL and Root Cause Analysis). At the moment the dashboards provide detailed compliance information about the two case study scenarios explored by the COMPAS project, named Loan Approval (Banking CGD) and WatchMe (Telecom CGD).

Videos about the CGDs and the data warehousing environment, as well as additional documentation, are available at [http://compas.disi.unitn.it/CGD/home.html](http://compas.disi.unitn.it/CGD/home.html).

5. Conclusion

In this deliverable we have presented the CGDs and the Root Cause Analysis prototypes, both relevant for to support compliance governance. Increasingly, industry and academia are investing money and efforts into the development of compliance governance solutions. Yet, we believe compliance governance dashboards in particular, probably the most effective means for visualizing and reporting on compliance, have mostly been neglected so far. It is important to implement sophisticated solutions to check compliance, but it is at least as important (if not even more) to effectively convey the results of the compliance checks to a variety of different actors, ranging from IT specialists to senior managers.

6. Reference documents

6.1. Internal documents

[D5.4] “Reasoning mechanisms to support the identification and the analysis of problems associated with user requests”, ver. 1.0 of 2009-12-15.
[D5.6] “Compliance mining tool”, ver. 1.0 of XXX.

6.2. External documents


[SF10] Source Forge. Available at http://sourceforge.net/