D5.5

Version:	1.1
Date:	2009-12-22
Dissemination status:	PU
Document reference:	D5.5

Initial 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 Initial Prototype of Compliance Governance Dashboards

Revision 1.1

Due date of deliverable: 2009-12-31

Actual submission date: 2009-12-22

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

Project funded by the European Commission within the Seventh Framework Programme		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

History chart

Issue	Date	Changed page(s)	Cause of change	Implemented by
0.1	2009-11-16	All sections	New document	UNITN
0.2	2009-11-19	All sections	Complete the First Draft	UNITN
0.3	2009-11-27	All sections	Updates according to our internal review	UNITN + USTUTT + TUV
0.4	2009-12-07	All sections	Updates according to THALES and TARC-PL reviews	UNITN + THALES + TARC-PL
0.5	2009-12-10	All section	Updates according to UCBL review	UNITN + UCBL
1.0	2009-12-12	All section	Updates according to UCBL review	UNITN
1.1	2009-12-22		Approve & Release	TUV

Authorisation

No.	Action	Company/Name	Date
1	Prepared	UNITN	2009-12-12
2	Approved	TUV	2009-12-22
3	Released	TUV	2009-12-22

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.

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

List of figures

List of tables

Abstract

Assessing whether a company's business practices 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 for the information 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.

This deliverable shows how to address these issues, which concepts and models underlie the problem, and, eventually, how IT can effectively support compliance analysis in Service-Oriented Architectures (SOAs). To achieve this, we first position the dashboards in the COMPAS runtime architecture. After that, we describe the WatchMe reference scenario taken from one of our project case studies and used along our examples of this deliverable. Then, we describe CGDs, their main concepts based on COMPAS conceptual model, and the navigation design used to do drill down/up. Thus, we present how to use CGDs on practices and how they should be implemented. Finally, we draw conclusions and point out the future work.

It is important to highlight that the content of this deliverable makes part of the paper named *On the Design of Compliance Governance Dashboards for Effective Compliance and Audit Management* accepted at the NFPSLAM-SOC'09 [SRC+09]. In addition, the reader should keep in mind that the content of this deliverable is a preliminary version of the CGDs in the COMPAS context. The final prototype will be finished at the end of the project, more precisely on month 35 [DoW].

1. Introduction

Compliance is a term generally used to refer to the conformance to a set of laws, regulations, contracts, 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: It is *important* because there is increasing regulatory pressure on companies to meet a variety of policies and laws (e.g., Basel II, MiFID, SOX). This increase has been to a large extent fuelled by high-profile bankruptcy cases (Parmalat, Enron, WorldCom, the recent crisis) or safety mishaps (the April 2009 earthquake in Italy has already led to stricter rules and certification procedures for buildings and construction companies). Failing to meet these regulations means safety risks, hefty penalties, loss of reputation, or even bankruptcy [Tre08].

Managing and auditing/certifying compliance is a very *expensive* endeavor. A report by AMR Research [HHG+08] estimated that companies were supposed to spend US$32B only on governance, compliance, and risk in 2008 and more than US$33B in 2009. Audits are themselves expensive and invasive activities, costly not only in terms of auditors' salaries, but also in terms of internal costs for preparing for and assisting the audit – not to mention the cost of non-compliance in terms of penalties and reputation.

Finally, the problem is *complex* because each corporation has to face a large set of compliance requirements in the various business segments, from how internal IT is managed to how personnel is trained, how product safety is ensured, or how (and how promptly) information is communicated to shareholders. Furthermore, rules are sometimes vague and informally specified. As a result, compliance governance requires understanding/interpreting requirements and implementing and managing a large number of control actions on a variety of procedures across the business units of a company. Each compliance source may require its own control mechanism and its own set of indicators to assess the compliance status of the procedure [BEF+07]. Today, compliance is to a large extent managed by the various business units in rather ad-hoc ways (each unit, line of business, or even each business process has its own methodology, policy, controls, and technology for managing compliance) [SRA+06]. As a result, today it is very hard for any CFO or CIO to answer questions such as: *Which rules does my company have to comply with? Which processes should obey which rules? Which processes are following compliance sources? Where do violations occur? Which processes do we have under control?* [CB06]. Even more, it is hard to do so from a perspective that not only satisfies the company but also the company's *auditors,* which is crucial as the auditors are the ones that certify compliance.

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.

The dashboard is targeted at several classes of users: chief officers of a company, line of business 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. They 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 root of the problem.

Technically, building a dashboard that shows a bunch of indicators and that allows drill-downs is easy. Indeed, the main challenges in this case are *conceptual* more than technological [All06]. These challenges, which also correspond to the main contributions of this deliverable, are:

1. Provide a *conceptual model for compliance dashboards* that covers a broad class of compliance issues. Identify the key abstractions and their relationships. Otherwise the dashboard loses its value of single entry point for compliance assessment.
2. Combine the above *broadness with simplicity and effectiveness*. The challenge here is to derive a model that, despite being broad, remains simple and useful/usable. If the abstractions are not carefully crafted and kept to a minimum, the dashboard will be too complex and remain unused. Models that are too generic are often too complex to use. As we have experienced, this problem may seem easy but is instead rather complex, up to the point that discussions on the conceptual model in the projects took well over a year. There is no clarity in this area, and this is demonstrated by the fact that while everybody talks about compliance, there are no generic but simple compliance models readily available.
3. Define, besides the conceptual abstractions, a *user interaction and navigation model* that captures the way the different kinds of users need to interact with the dashboard, to minimize the time to accesses spent in getting the information users need and to make sure that key problems do not remain unnoticed.
4. Derive a model that is in line with the *criteria and approach that auditors have* to verify compliance. In this deliverable, this last contribution is achieved “by design”, in that the model is derived also via a joint effort of two of the major auditing companies and reflects the desired method of understanding of and navigation among the various compliance concerns.

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 their interaction with other architecture components.

In such an architecture (Figure 1) we can see that all events from different fonts (i.e., Business process engine, Runtime compliance monitoring, 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 structured according to a data model can be used as input for the Analysis and Business Intelligence Components. This lasts are in charge of computing Key Compliance Indicators (KCIs) and providing root-cause analysis of violations (more details about both are documented in [D5.4]).

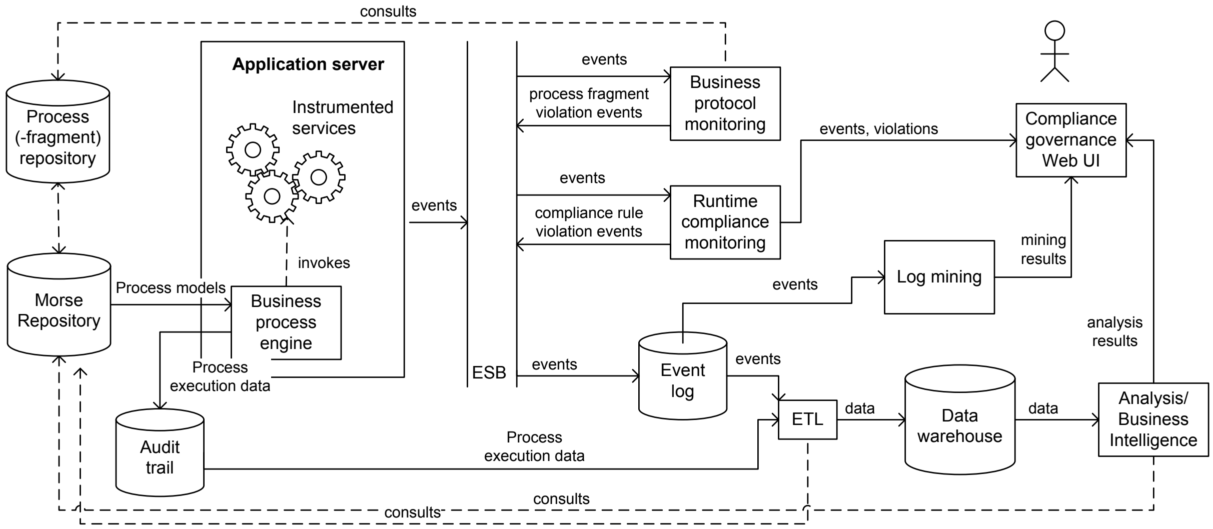


Figure 1 Compliance governance runtime architecture, an extension of [D5.3]

Given that, the CGDs make part of the Compliance governance Web User Interface (UI) and are one of the presentation components used to do offline monitoring, showing KCIs and details about violations. Both, KCIs and low level violations take into account different summarization levels and analysis perspectives. 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. More details about such compliance visualisation are provided in this deliverable. The runtime monitoring as a results from the Runtime rule evaluator and Business protocol monitoring should be respectively provided by UBCL and Telcordia in the [D5.4].

1.2. Document overview

This deliverable starts explaining the complexities and challenges faced in order to provide CGDs, as well as, the role of dashboards in COMPAS. To better contextualize and explain our solution, we first present a reference scenario at the beginning of the document, named WatchMe, in which we based the content of the CGDs presented here. Then, the remaining sections are as follows: Section 2 describes and specifies the CGDs, together with a conceptual model for governance dashboards and a navigation design model. Section 3 shows the CGDs in practice and the details regarding their design and implementation. We conclude in Section 4 by drawing the conclusions and anticipating on future work.

1.3. Reference Scenario

The WatchMe scenario focuses on the advanced telecom services offered by Mobile Virtual Network Operators (MVNO) [D5.3]. In that greenfield scenario, one of the challenges is how to monitor the compliance status of an organization that deals with different licenses adopted by distinct audio and video providers, as well as, a diversity of offered plans to consumers (i.e., Pay-per-view, Time-based). Typically, such licenses and plans are stated by means of contracts/agreements that follow one or more sources of compliance regarding copyright laws (e.g., USA Copyright Law - www.copyright.gov, European Copyright Law - www.eucopyright.com, Entertainment Law, USA Digital Millennium Copyright Act - www.copyright.gov/legislation/dmca.pdf, and USA Privacy and Publicity Rights - memory.loc.gov/ammem/copothr.html).

Compliance source is a document (e.g., contracts, agreements) that is the origin of compliance requirements [D7.1] that must be followed in order to keep the company complaint. 0 contains the list of compliance requirements adopted in the WatchMe scenario, as well as the descriptions settled to each of them. More specifically, requirements concern Internal policy, Licensing and Quality of Service (QoS). As stated at the conceptual model of compliance, initially proposed in [D5.3], compliance requirements are high level interpretations of compliance sources. At run-time, such requirements are defined as compliance rules that are monitored in order to govern the compliance performance of the organization. In this deliverable we do not discuss the implementation of these concepts, our dashboards implementation is independent and mainly focus on the presentation components and the set of data that is going to be provided by the CGDs. Details about how data are stored, how to detect violations and compute indicators are respectively documented in [D5.3][D5.4].

	Compliance Requirements	**Description of Compliance Requirements**	**Control**
Internal policy	Protection of the WatchMe service	The usage of WatchMe service is only allowed for registered users.	A user has to identify himself when interacting with the WatchMe service.
Licensing	Pay-per-view plan	When the WatchMe company subscribes for the Pay-per-view plan it acquires a *limited* number of streams based on *the amount paid* to the media supplier.	When WatchMe company subscribes for the Pay-per-view plan it has to pay **29.90 euro first** and **then receive 300** streams from the media supplier.
	Time-based plan	When the WatchMe company subscribes for the Time-based plan it acquires *any* number of times *any* possible streams in a certain period, based on *the amount paid* to the media supplier.	When WatchMe company subscribes for the time-based plan it has to pay **89.90 euro first** and then receive *an unlimited* number of times *any* available stream from the media supplier **in a 30 days** period starting from the contract start date.
	Composition permission	**Only pre-defined combinations** of video and audio providers are allowed due to the licenses specified by the video provider.	**VideoTube** can only have audios streams from **AudioTube** or **QuickAudio**. **QuickVideo** can only have audio streams from **QuickAudio**.
QoS	Delivery Rate	The WatchMe service must deliver in a fixed period of time the specified number of URLs for downloading a stream.	The **WatchMe** service must deliver a valid URL at least in **90% of requests per customer subscription**.
	Availability	The WatchMe service must be available as specified to the customer in the	The **WatchMe** service must be available **99% of the time per customer subscription.**

		contractual agreement of the subscription.	
	Response time	The response time for getting a URL of the requested media is as specified to the customer in the contractual agreement of the subscription.	The **WatchMe** service provides a URL of the requested media within **45** seconds to the customer.

Table 1 Compliance requirements of the WatchMe scenario

1.4. Definitions and glossary

The most important terminology concerning the COMPAS project is listed on the public COMPAS Web-Site [D7.1] available at http://www.compas-ict.eu, section Terminology. This helps to make the overall COMPAS approach more comprehensive for the general public.

1.5. Abbreviations and acronyms

ARF	Attribute-Relation File
CFO	Chief Financial Officer
CGD	Compliance Governance Dashboard
CIO	Chief Information Officer
DB	Database
DW	Data Warehouse
ESB	Enterprise Service Bus
HTTP	HyperText Transfer Protocol
IT	Information Technology
JSON	JavaScript Object Notation
KCI	Key Compliance Indicator
KPI	Key Performance Indicator
MiFID	Markets in Financial Instruments Directive
MVNO	Mobile Virtual Network Operators
OLAP	On-line Analytical Processing
PwC	PricewaterhouseCoopers
QoS	Quality of Service
SOA	Service-Oriented Architecture
SOX	Sarbanes-Oxley Act
XML	Extensible Markup Language

WebML	Web Modeling Language
Weka	Waikato Environment for Knowledge Analysis

2. Compliance Governance Dashboards description

To aid the internal evaluation and to help a company pass external audits, a concise and intuitive visualization of its compliance state is paramount. To report on compliance, we advocate the use of a web-based *Compliance Governance dashboards* (CGDs), whose good design is not trivial [Few06][Pap08]. It is important to understand: (i) what typical information auditors expect to find; (ii) how large amounts of data can be visualized in an effective manner, and how data can be meaningfully grouped and summarized; and (iii) how to structure the available information into multiple pages, that is, how to interactively and intuitively guide the user through the wealth of information. Each page of the dashboard should be concise and intuitive, yet complete and expressive. It is important that users are immediately able to identify the key information in a page, but also providing facilities to drill-down into details.

Designing CGDs requires mastering some new concepts in addition to those presented in the COMPAS compliance conceptual model [D7.1]. Then, the new concepts must be equipped with a well-thought navigation structure to effectively convey the necessary information.

2.1. A Conceptual Model for Compliance Governance Dashboards

In Figure 2 we extend the conceptual model [D7.1] to capture the necessary constructs for the development of CGDs (bold lines and labels represent new entities and their respective interrelations). The extensions aim at (i) providing different *analysis perspectives* (in terms of time, user roles, and organizational structures), (ii) *summarizing* data at different levels of abstraction, and (iii) *enabling* drill-down/roll-up features (from aggregated data to detailed data, and vice versa).

The *Dashboard view* entity represents individual views over the compliance status of the company. A view is characterized by the user role that accesses it, e.g., IT specialists, compliance experts, managers, or similar. Each of these roles has different needs and rights. For instance, managers are more interested in aggregated values, risk levels, and long time horizons (to take business decisions); IT personnel are rather interested in instance-level data and short time spans (to fix violations), they also seek to associate with each compliance violation, the regulation in question, the origin and nature of the violation, and the compliance risk level. A view is further characterized by the *time interval* considered for the visualization of data (e.g., day, week, month, or year), also providing for the historical analysis (e.g., last year) and supporting different reporting purposes (operative, tactical, strategic). Finally, a view might be restricted to only some of the *business units* in the company, based on the role of the user. Business units can be composed of other business units, forming a hierarchical organizational structure. In summary, views support different summarization levels of the overall available data, ranging over multiple granularity levels.

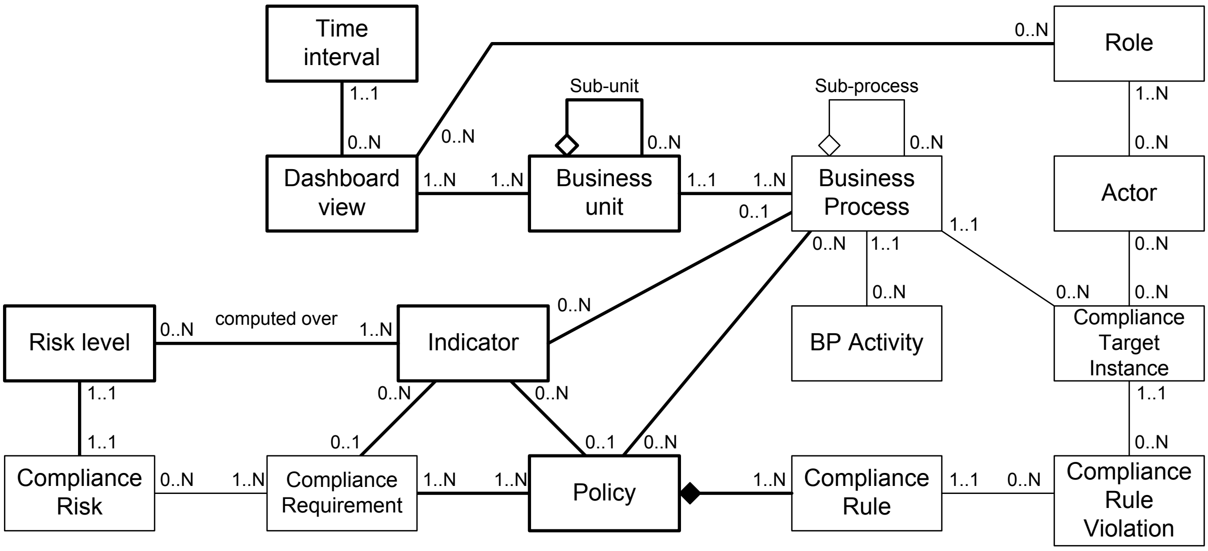


Figure 2 Conceptual model for CGDs (dashboard-specific constructs are highlighted in bold)

Effective summarization of data is one of the most challenging aspects in the design of CGDs dashboards, commonly instrumented by indicators [Lam07]. Mainly, due to the difficulties of defining, quantifying, and managing operational risk; of adding tangible value to the business; and of interpretation, internalization and representation of compliance requirements in a way that can be used to manage compliance at runtime using IT support and taking into account complex events and real time constraints.

In that context, an *indicator* is a quantitative summarization of a particular aspect of interest in the business, i.e., a metric of how well an objective is being reached. Typically, KPIs (key performance indicators), are used to summarize the level at which business objectives are reached. In our context, we speak about KCIs, referring to the achievement of the stated compliance objectives (e.g., the number of unauthorized accesses to our payroll data). More details about KCIs are available in [D5.4].

The described model extension aims at relating general compliance concepts, initially defined in COMPAS conceptual model [D7.1], with concepts that are specific to the design of dashboards. The model is general and extensible, so as to allow for the necessary flexibility to accommodate multiple concrete compliance scenarios.

2.2. Navigation Design for Compliance Governance Dashboards

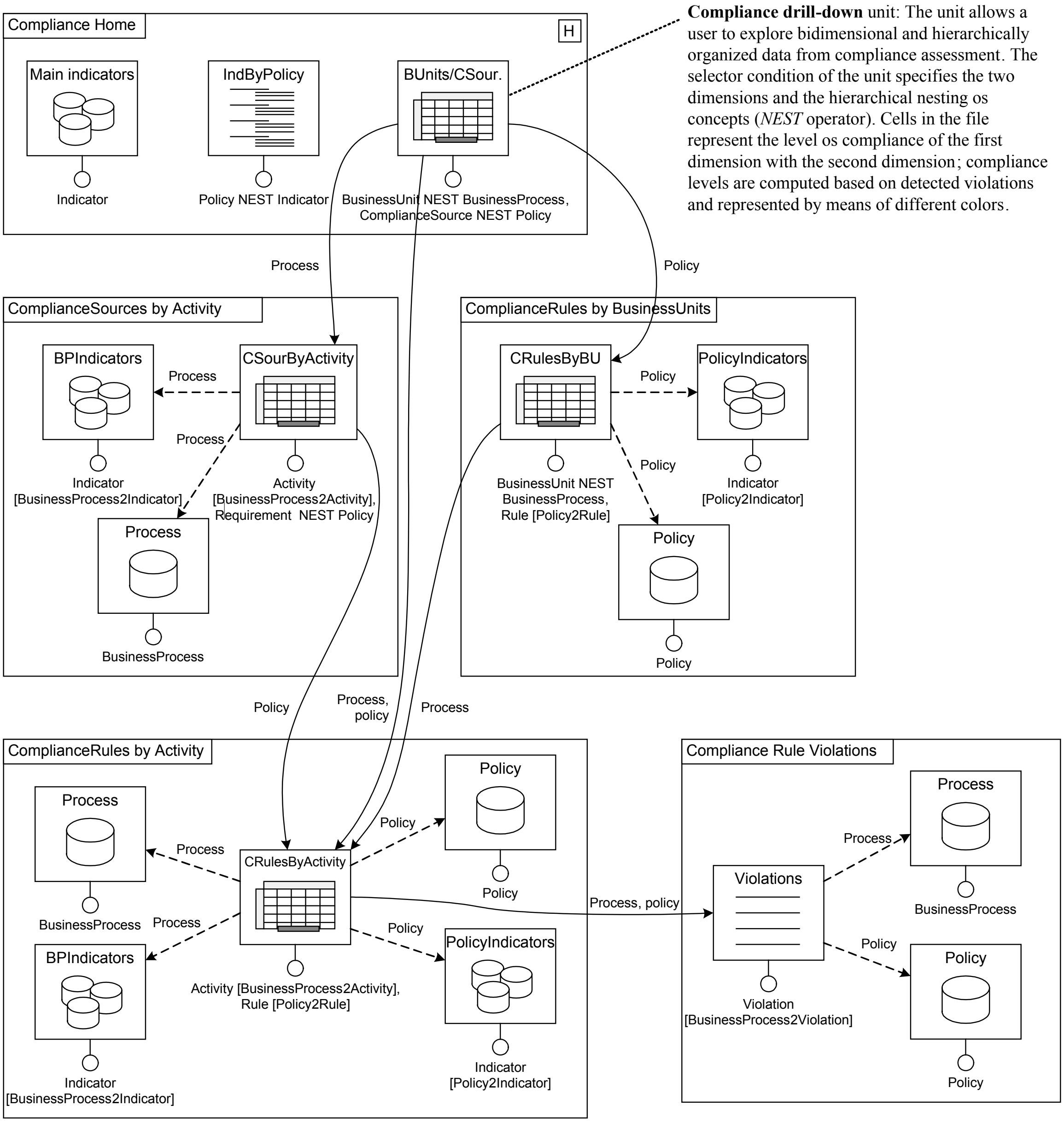
After discussing the *static* aspects of the design of CGDs, we now focus on the *dynamic* aspect, i.e., on how to structure the interaction of users with the dashboard, and on how users can explore the data underlying the dashboard application. Specifically, on top of the conceptual model for CGDs, we now describe how complex data can be organized into hypertext pages and which navigation paths are important.

For this purpose, we adopt the Web Modeling Language (WebML) [CFB+02], a conceptual modeling notation and methodology for the development of data-intensive web applications. We use the language for the purpose of illustration only (we show a simplified, not executable WebML schema) and intuitively introduce all the necessary constructs along with the description of the actual CGDs navigation structure.

The WebML hypertext schema (Figure 3) describes the organization of our ideal web CGDs. It consists of five *pages* (the boxes with the name labels in the upper left corner), Compliance Home being the home page (note the H label). Each page contains a number of *content units*, which represent the publication of contents from the data schema in Figure 2 (the *selector* condition below the units indicates the source data entity). Usually, there are many *hyperlinks* (the arrows) in a hypertext schema, representing the possible navigations a user might perform, but, for simplicity, we limit our explanation to only those links that represent the main navigation flow. Links carry *parameters*, which represent the selection performed by the user when activating a link (e.g., the selection of a process from a list). For the purpose of reporting on compliance, we define a new content unit (not part of the WebML), the *compliance drill-down* unit, which allows us to comfortably show compliance data in a table-like structure (see the legend in Figure 3 and the examples in Figure 4).

Let's examine the CGD's structure (Figure 3): The home page of the CGD provides insight into the compliance state of the company at a glance. It shows the set of most important indicators (Main indicators *multidata* unit) and a set of indicators grouped by their policy (IndByPolicy *hierarchical index* unit). Then, we show the (BUnits/C.Sour.) unit that allows the user to drill-down from business units to processes and from compliance sources to policies. A click on one of: i) the processes leads the user to the ComplianceSources by Activity page; ii) compliance source leads her to the ComplianceRules by BusinessUnits page; and iii) the cell of the table leads her to the ComplianceRules by Activity page. After the selection of a process, in the ComplianceSources by Activity page the user can inspect the compliance state of each activity of the selected process with the given compliance sources and policies (CSourByActivity), a set of related indicators (BPIndicators unit; the unit consumes the Process parameter), and the details of the selected process (Process *data* unit). Similar details are shown for policies in the ComplianceRules by BusinessUnits page, which allows the user to inspect the satisfaction of individual compliance rules at business unit or process level (ComplianceRulesByBU). A further selection in the compliance drill-down units in these last two pages or the selection of a cell in the BUnits/C.Sour unit in the home page leads the user to the ComplianceRules by Activity page, which provides the user with the lowest level of aggregated information. It visualizes the satisfaction of the compliance rules of the chosen policy by the individual activities of the chosen process (ComplianceRulesByActivity), along with the details of the chosen policy and process and their respective indicators. A further selection in this page leads the user to the Compliance Rule Violations page, which shows the details of the violations related to the chosen process/policy combination at an instance level in the Compliance Rule Violations *index* unit.

The navigation structure in Figure 3 shows one of the possible views over the data in Figure 2, e.g., the one of the internal compliance expert. Other views can easily be added. Each page provides a different level of summarization (overview, process-specific, policy-specific, process- and policy-specific, violation instances), guiding the user from high-level information to low-level details. The time interval to be considered for the visualization can be chosen in each of the pages.



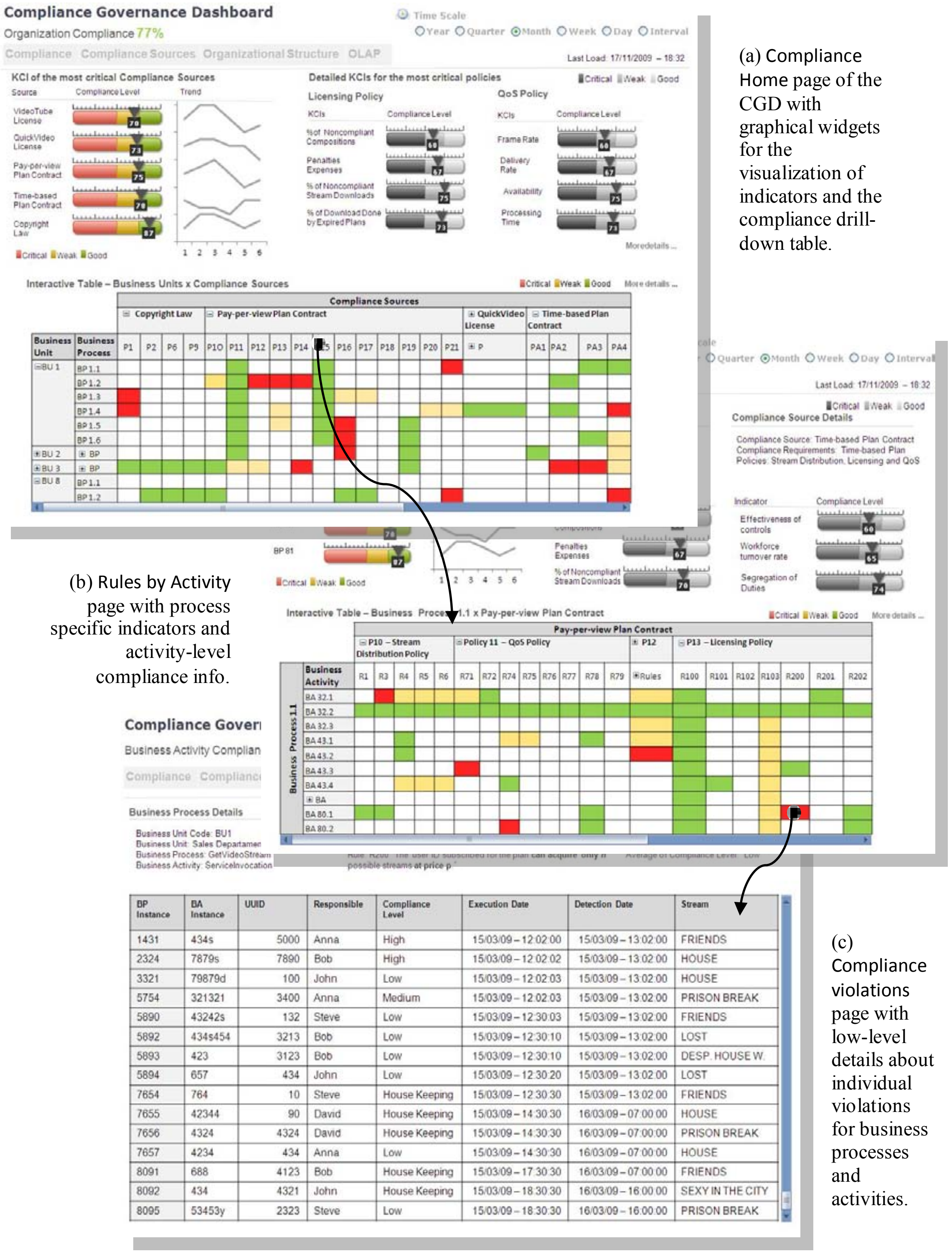
Compliance drill-down unit: The unit allows a user to explore bidimensional and hierarchically organized data from compliance assessment. The selector condition of the unit specifies the two dimensions and the hierarchical nesting os concepts (*NEST* operator). Cells in the file represent the level os compliance of the first dimension with the second dimension; compliance levels are computed based on detected violations and represented by means of different colors.

Figure 3 WebML hypertext schema structuring the navigation of CGDs concepts and data

3. Compliance Governance Dashboards in Practice

To provide the look-and-feel idea we have implemented, in Figure 4 we illustrate screenshots from our prototype CGD. The screenshots show views that clarify and consistently present our ideal CGD. Figure 4(a) shows the Compliance Home page (Figure 3), Figure 4(b) the Compliance Rules by Activity page, and Figure 4(c) the Compliance Rules Violations page.

Compliance Home concentrates on the most important information at a glance, condensed into just one page (see Figure 4). It represents the highest granularity of information. The five colored indicators (top left) are the most relevant, showing the most critical non compliant sources. The gray indicators (right) report on the compliance with the two main policies. In the bottom, there is the interactive compliance drill-down table


Compliance Governance Dashboard
Organization Compliance 77%
Time Scale
○ Year ○ Quarter ● Month ○ Week ○ Day ○ Interval
Compliance Compliance Sources Organizational Structure OLAP
Last Load: 17/11/2009 – 18:32
KCI of the most critical Compliance Sources
Source Compliance Level Trend
VideoTube License
QuickVideo License
Pay-per-view Plan Contract
Time-based Plan Contract
Copyright Law
Critical Weak Good
Detailed KCIs for the most critical policies
Licensing Policy
QoS Policy
% of Noncompliant Compositions
Penalties Expenses
% of Noncompliant Stream Downloads
% of Download Done by Expired Plans
Frame Rate
Delivery Rate
Availability
Processing Time
More details ...
Interactive Table – Business Units x Compliance Sources
Compliance Sources
Copyright Law
Pay-per-view Plan Contract
QuickVideo License
Time-based Plan Contract
Business Unit
Business Process
BU 1
BU 2
BU 3
BU 8
Compliance Source Details
Compliance Source: Time-based Plan Contract
Compliance Requirements: Time-based Plan
Policies: Stream Distribution, Licensing and QoS
Indicator
Effectiveness of controls
Workforce turnover rate
Segregation of Duties
Interactive Table – Business Process 1.1 x Pay-per-view Plan Contract
Pay-per-view Plan Contract
P10 – Stream Distribution Policy
Policy 11 – QoS Policy
P12
P13 – Licensing Policy
Business Activity
Business Process 1.1
Compliance Governance
Business Activity Compliance
Business Process Details
Rule: R200 "The user ID subscribed for the plan can acquire only n possible streams at price p"
Average of Compliance Level: Low

Business Unit Code: BU1
Business Unit: Sales Departamen
Business Process: GetVideoStream
Business Activity: ServiceInvocation

BP Instance	BA Instance	UUID	Responsible	Compliance Level	Execution Date	Detection Date	Stream
1431	434s	5000	Anna	High	15/03/09 – 12:02:00	15/03/09 – 13:02:00	FRIENDS
2324	7879s	7890	Bob	High	15/03/09 – 12:02:02	15/03/09 – 13:02:00	HOUSE
3321	79879d	100	John	Low	15/03/09 – 12:02:03	15/03/09 – 13:02:00	HOUSE
5754	321321	3400	Anna	Medium	15/03/09 – 12:02:03	15/03/09 – 13:02:00	PRISON BREAK
5890	43242s	132	Steve	Low	15/03/09 – 12:30:03	15/03/09 – 13:02:00	FRIENDS
5892	434s454	3213	Bob	Low	15/03/09 – 12:30:10	15/03/09 – 13:02:00	LOST
5893	423	3123	Bob	Low	15/03/09 – 12:30:10	15/03/09 – 13:02:00	DESP. HOUSE W.
5894	657	434	John	Low	15/03/09 – 12:30:20	15/03/09 – 13:02:00	LOST
7654	764	10	Steve	House Keeping	15/03/09 – 12:30:30	15/03/09 – 13:02:00	FRIENDS
7655	42344	90	David	House Keeping	15/03/09 – 14:30:30	16/03/09 – 07:00:00	HOUSE
7656	4324	4324	David	House Keeping	15/03/09 – 14:30:30	16/03/09 – 07:00:00	PRISON BREAK
7657	4234	434	Anna	Low	15/03/09 – 14:30:30	16/03/09 – 07:00:00	HOUSE
8091	688	4123	Bob	House Keeping	15/03/09 – 17:30:30	16/03/09 – 07:00:00	FRIENDS
8092	434	4321	John	House Keeping	15/03/09 – 18:30:30	16/03/09 – 16:00:00	SEXY IN THE CITY
8095	53453y	2323	Steve	Low	15/03/09 – 18:30:30	16/03/09 – 16:00:00	PRISON BREAK

(a) Compliance Home page of the CGD with graphical widgets for the visualization of indicators and the compliance drill-down table.

(b) Rules by Activity page with process specific indicators and activity-level compliance info.

(c) Compliance violations page with low-level details about individual violations for business processes and activities.

Figure 4 Example CRD screenshots of our prototype implementation.

containing the compliance performance of business units and processes (rows) in relation to compliance sources and policies (columns). The user can easily reach lower levels of granularity by drilling down on the table or navigating to pages. For instance, the ComplianceRules by Activity page condenses lower level information concerning a combination of Business Process 1.1 and the Pay-per-view Plan Contract. The colors of the cells represent the compliance performance of each combination. For instance, the Business activity BA 80.1 presents a critical situation regarding Rule 200 of P13 – Licensing Policy (red cell) and weak performance regarding Rule 103 (yellow cell).

A drill-down on the red cell, for instance, leads us to the Compliance Rule Violations page, which provides the lowest level of abstraction in form of a table of concrete, registered violations of the selected rule. The page illustrates the main information that must be reported to assist internal and external auditors. The data in the particular page reports all violations of one activity in Business Process 1.1 of Business Unit 1, detected considering Rule 200 of Pay-per-view Plan Contract - P13 – Licensing Policy. Each row of the table represents a distinct violation and the columns contain the typical information required by auditors, e.g., responsible of activity, date and time, compliance level, stream name, cause of violation, and UUID (Universally Unique Identifier), which uniquely identify a model or a model element (see [HZD09]).

In each page of the CGD, the amount and the position of the graphical widgets for 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].

3.1. Design and implementation

The CGD is set on the top of a data DW (optimized for reporting purposes) that implements the conceptual model described in Figure 2. It is however important to recognize that this does not affect the logic behind the designed navigation structure (Figure 3), 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.

In order to implement the CGDs we designed an architecture composed of three distinct layers, as depicted by Figure 5, in which each layer has very specific roles and functionalities. In a bottom-up approach, the first layer is named *DB server* and has as main and unique component the DW. Then, the second layer, called *Web server*, is responsible for the communication and exchange of data between the other two layers, as well as the creation of dedicated files (i.e., ARF, XML) to be used as input in the layer above. Finally, there is the Client browser composed of the web pages where the CGDs and additional visualization components (Interactive table, Pivot table, and Decision trees) become available to the end users. The further items describe the design and implementation details of the Client browser and Web server layers together with their main components. The prototype of CGDs is available at http://compas.disi.unitn.it:8080/CGDs/main.jsp and details about the DB Server can be found in [D5.3].

Client browser

- *Dashboard widgets*: are the components used to graphically visualize the KCIs, as illustrated in the top side of Figures 4 (a) and (b). They are implemented using Java code and Fusion Charts libraries [Fus09]. The latter requires files in a pre-defined XML format containing the data to be shown in the widgets. Such file is generated by Widget data loader in the above layer and received by the client by means of HTTP protocol.
- *Interactive table*: is used to provide a drill-down approach of the compliance status of the different business execution levels, according to most important entities of the conceptual model for CGDs (see Section 2.2). Very likely this table will be implemented using Java and its content will be obtained using the *Interactive table*

loader. This later invokes the BD query connector (Web server layer) and receives a JSON message [JAS09].

- *Pivot table*: is an OLAP feature of the data warehousing environment that allows users to build queries just dragging and dropping the dimensions and facts of DW data model. As the interactive table, this table either provides a drill-down approach but showing the data stored in the DW, instead of computed indicators. Initially, our idea is to implement such component using Java and a dedicated library to create pivot tables named JIDE Pivot Gride [Jid09]. In addition to a *Data loader* component that invokes the DB query connector and receives JSON messages. However, by the time of implementation such specifications could be changed.
- *Decision tree*: is the most common visualization component to show the results obtained using classification methods. Very probably, such trees will be generated using Weka [Wek09], at the Web server layer, and passed by JSON to a java code in order to be published in the Client browser.

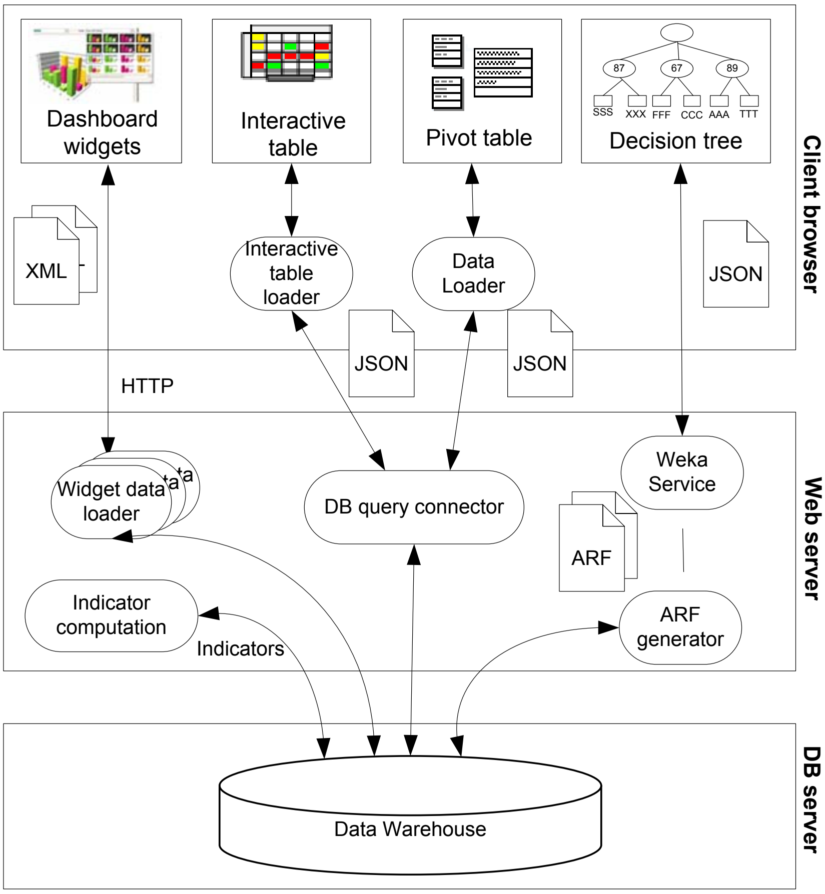


Figure 5 Functional architecture for reporting compliance governance.

Web server

- *Indicator computation*: calculates the KCIs and stores the final results in the DW.

- *Widget data loader*: according to the invocation emitted by the Client browser, the loader implemented in Java retrieves the DW and generates the one or more XML files with the requested data.
- *Weka service and ARF generator*: Weka service is responsible for executing the classification algorithm (C4.5) and generating decision trees. To generate trees, Weka first invokes *ARF generator*, in order to receive the input data in the ARF format required by the algorithm. Such generator retrieves the DW and captures the requested data and structures them according to the required format. Then, Weka is executed and its results are passed with JSON to the Decision tree component in the client layer.
- *DB query connector*: this Java component receives invocations that have as parameters the data to be extracted from the DW. Hence, it connects the database and sends JSON messages with the requested content.

Since this deliverable only describes the preliminary version of the CGDs prototype, more specifications and definition of dashboard architecture and implementation will be published in the final version of the Dashboards prototype by month 35 as stated in the [DoW].

4. Conclusion and Future work

In this deliverable we have discussed a relevant aspect in modern business software systems, i.e., compliance governance. Increasingly, both 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. Our contribution is a conceptualization of the issues involved in the design of compliance governance dashboards in service- and process-centric systems, the definition of a navigation structure that naturally supports drill-down and roll-up features at adequate levels of detail and complexity, and a set of concrete examples hat demonstrate the concepts at work. Our aim was to devise a solution with in mind the real needs of auditors (internal and external ones) and – more importantly – with the help of people who are indeed involved every day in the auditing of companies.

As a continuation of this work, we are planning to implement the components present here using the data from the WatchMe scenario. That way, we expect to assess the acceptance of the proposed CGD by auditors in their everyday work, and understand which support for actions for mitigating compliance problems or violations directly through the dashboard is desirable.

5. Reference documents

5.1. Internal documents

[D5.3] "Final goal-oriented data model", ver. 1.0 of 2009-07-31.

[D5.4] "Reasoning mechanisms to support the identification and the analysis of problems associated with user requests", ver. 1.0 of 2009-12-15.

[D7.1] "Public Web-Site", http://www.compas-ict.eu

[DoW] "Description of Work", ver. 15 of 2007-09-25.

5.2. External documents

[All06] E. Allman. Complying with Compliance. ACM Queue, 4(7), pp. 18-21, September, 2006.

[BEF+07] R. Bellamy, T. Erickson, B. Fuller, W. Kellogg, R. Rosenbaum, J. Thomas, T. Vetting Wolf. Seeing is believing: Designing visualizations for managing risk and compliance. IBM Systems Journal, 46(2), pp. 205-218, 2007.

[CB06] J. Cannon, M. Byers. Compliance deconstructed. ACM Queue, 4(7), pp. 30-37, September, 2006.

[CFB+02] S. Ceri, P. Fraternali, A. Bongio, M. Brambilla, S. Comai, M. Matera. Designing Data-Intensive Web Applications. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.

[Few06] S. Few. Information Dashboard Design: The Effective Visual Communication of Data. O'Reilly Media, Inc., p. 223, 2006.

[Fus09] Fusion Solutions, 2009. http://www.fusioncharts.com/

[HHG+08] J. Hagerty, J. Hackbush, D. Gaughan, S. Jacobson. The Governance, Risk Management, and Compliance Spending Report, 2008-2009: Inside the $32B GRC Market. AMR Research, 2008.

[HZD09] T. Holmes, U. Zdun, S. Dustdar. MORSE: A Model-Aware Service Environment 4th IEEE Asia-Pacific Services Computing Conference, *to appear*, IEEE Computer Society Press, IEEE Conference Proceeding, 2009.

[JAS] JSON (JavaScript Object Notation), 2009. http://www.json.org/.

[JID09] JIDE Pivot Grid, 2009. http://www.jidesoft.com/products/pivot.htm.

[Lam07] J. Lam. Operational Risk Management – Beyond Compliance to Value Creation. White Paper, Open Pages, 2007.

[Pap08] M. Papazoglou. Compliance Requirements for Business-process-driven SOAs. E-Government Ict Professionalism and Competences Service Science, July, Volume 280/2008. pp. 183-194, 2008.

[SRC+09] P. Silveira, C. Rodríguez, F. Casati, F. Daniel, V. D'Andrea, C. Worledge, Z. Taheri. On the Design of Compliance Governance Dashboards for Effective Compliance and Audit Management. Proceedings of NFPSLAM-SOC'09, November 2009, Springer, Stockholm, Sweden.

[SRA+06] E. Sloane, E. Rosow, J. Adam, D. Shine. JEDI - An Executive Dashboard and Decision Support System for Lean Global Military Medical Resource and Logistics Management. Proceedings of the EMBS Annual International Conference, pp. 5440-5443, 2006.

[Tre08] H. Trent. Products for Managing Governance, Risk, and Compliance: Market Fluff or Relevant Stuff? In-Depth Research Report, Burton Group, 2008.

[Wek09] Weka 3: Data Mining Software in Java, 2009. http://www.cs.waikato.ac.nz/ml/weka/.