PROJECT FINAL REPORT

Front Page

Project Information				
Grant Agreement Number	619706			
Project Acronym	ASAP			
Project Title	ASAP: An Adaptable Scalable Analytics Platform			
Funding Scheme	STREP			
Period covered	From	2014-03-01	to	2017-02-28

Contact Information	
Project coordinator name	Polyvios Pratikakis
Project coordinator organization	FORTH: Foundation for Research and Technology – Hellas
Tel	+30 281 039 1949
Fax	+30 281 039 1601
E-mail	polyvios@ics.forth.gr
Project website address	http://www.asap-fp7.eu/

Document Revision History			
Version	Date	Author	Comments
0.1	21 Jul 2017	P. Pratikakis	Initial Version
1.0	05 Aug 2017	P. Pratikakis	Final Version

Contents

1 Final Publishable Summary Report

1.1 Executive Summary

ASAP (March 2014–February 2017) is a FP7 project tasked by the European Commission to develop a dynamic, open-source execution framework for scalable data analytics. The underlying idea behind ASAP is that in the complex workflows of analytics applications no single execution model is suitable for all types of tasks, and no single data model (or store) is suitable for all types of data. Complex analytical tasks in multi-engine environments therefore require integrated profiling, modeling, planning and scheduling functions.

Project ASAP built a platform that facilitates managing such complex analytics application workflows by providing a common way to construct, manage, maintain, and execute workflows. The ASAP project reached its goal to support the building and execution of complex analytics workflows at all stages of their lifetime, by achieving its four main goals:

- *Facilitate the development of analytics operators:* ASAP developed a generic task-parallel programming model, in conjunction with two runtime systems for distributed or parallel execution in the cloud. The runtimes include state-of-the-art features such as irregular general-purpose computations, resource elasticity and synchronization, data-transfer, locality and scheduling abstraction, the ability to handle large sets of irregularly distributed data, and fault-tolerance.

- *Continuous modeling of analytics workflows:* ASAP developed a modeling framework that constantly evaluates the cost, quality and performance of available computational resources in order to decide on the most advantageous store, indexing and execution pattern.

- *Online monitoring and adaptation of analytics workflows:* ASAP developed adaptation methodology to enable analytics experts to amend submitted workflows by changing or modifying a workflow while it is being processed. Users can change the parameters of operators already comprised in the workflow, or the structure of the workflow by removing or adding operators.

- *Visualization and understanding of analytics results:* ASAP developed a visual analytics dashboard to show query results and metadata in an intuitive manner, with special focus on the interactive exploration of datasets, dynamic temporal controls, on-the-fly query refinement mechanisms, and the geospatial projection of structured and unstructured data.

1.2 Project Context and Objectives

The field of data analytics includes techniques, algorithms and tools used to inspect collections of data to extract patterns, generalizations and other useful information, in order to extract value from data. As data becomes Big Data —i.e., large volumes of variable kinds of data that increase at high velocity and have imperfect veracity— the value extracted from data increases, but so does the complexity of an analytics application. Big data analytics is very important in risk assessment, pharmaceutics, fraud detection, epidemiology, business process effectiveness, market analysis, anti-terrorism, etc. More importantly, large-scale analytical data processing has become a necessity in the majority of industries. Enabling engineers, analytics experts and scientists alike to tap the potential of vast amounts of business-critical data has grown increasingly important. Such data analysis demands a high degree of parallelism, in both storage and computation. Business data centers host vast amounts of data, stored over large numbers of nodes with multiple storage devices, and process them using thousands or millions of cores.

To be useful and effective, analytics applications must produce near-real-time (or interactive) response times to the engineers' queries, a task that directly conflicts with the size of the data, their diversity of structure and representation, their location, and the ad-hoc nature of analytical queries. These issues have given rise to many diverse programming models, execution engines and data stores to assist with large-scale data management. Such systems target specific kinds of data or computations, where they greatly outperform general-purpose solutions like traditional relational databases. For instance, key-value stores drop the relational model in favor of much greater scalability and parallelism, reaching high IOPS (Input/Output Operations per Second) performance at a fraction of the cost of a relational system. For instance, the most widespread programming model for scaling applications to big data is Map-Reduce, which is very effective with computations expressed as data-parallel "mapper" tasks and "reducer" tasks that merge the results. Many operations, however, act on irregular data that may be heterogeneous, structured or unstructured, streaming or stored in various formats. Complex analytics applications may also perform computations with dynamic dependencies. For example, a graph computation may be data dependent, not knowing the next data to be processed before visiting the previous node. Similarly, online games or analytics of social graphs perform graph traversals or fixpoint computations that are difficult and inefficient in Map-Reduce, as each phase must store the intermediate graph state and redistribute it across machines before the next phase starts. To alleviate these difficulties, specialized models and systems have been proposed, such as Pregel, Hama, Dremel, and Powerdrill, implementing different computing models. While all these systems have had great success, they still showcase their advantages on a limited subset of applications and types of data: For instance, graph-processing engines limit the amount of freedom in the computation at each node (or part of a graph) and fail to fully exploit possible parallelism.

In practice, modern analytics applications try to take advantage of the strong points of many of these tools and combine these diverse programming and storage models into *workflows*. Analytics workflows often form static or dynamic data flows where each part of the workflow may use a different storage system, programming model, or runtime system, that best fit its specific computation. As applications increase in complexity, so does the design of workflows, because it is not always obvious what the best implementation is for a given computation. A storage format that may fit random data may not work well with sorted data; a runtime system that may work well with graph algorithms may not work well for machine learning; or the optimal combination of tools may even depend dynamically on the size of the problem.

Overall, the design, development, deployment, and execution of complex analytics workflows can become quite difficult:

- First, dynamic data processing is often very heterogeneous in terms of complexity and time consumption. In graph-structured data, for instance, the amount of stored data and required computation may differ from node to node. Thus, computing graph-queries in lock-step (i.e., by repeating whole-graph processing steps as in Pregel) loses parallelism, as all computations of a phase depend on the computations of the previous phase. Moreover, this requires the storing of all intermediate results which may not all be necessary for the next phase. For example, a graph query may need to repeat reductions over the whole graph until the longest path or cycle converges.

- Second, modern data centers often include many heterogeneous storage formats, each used in multiple contexts and by different applications. For instance, a data center running multiple applications may include data stores ranging from traditional row-stores and modern column-stores, unstructured data in raw files and semi-structured data in XML, RDF or similar formats stored either in files, adapted relational stores or specialized stores (e.g., RDF stores). Depending on the data format, the computation of a query may differ in complexity and performance. As it is not always optimal to convert data formats and storage, for legacy (existing working applications) or performance optimization (data format optimized for application queries) reasons, data analytics framework that scales the data center needs to support and adapt to multiple data storage formats.

- Lastly, data centers often host multiple applications. For example, an application may involve the processing of a data stream, by fast querying and updating data in one or more formats and data stores, while other applications perform long-running queries with multiple phases over the same data. Taking this discussion one step further, it is evident that the ad-hoc manner of data analytics, together with the sheer size and complexity of data, call for increased human participation: Scientists and engineers often “experiment” by posing long-running queries on huge datasets,

trying to identify trends and fuse data into new "signals". As most of these operations are particularly I/O- and time-consuming, an early evaluation and re-calibration of the submission parameters would greatly assist the process.

The ASAP project addresses these issues, delivering a *fully automated and highly customizable system* for the easy development and execution of arbitrary data analytics workflows on large heterogeneous data stores. The **ASAP** (**A**daptive, highly **S**calable **A**nalytics **P**latform) project provides a complete software stack that efficiently executes complex analytics workflows over large, heterogeneous, irregular or unstructured data. To achieve that, the consortium

- developed a programming model for writing analytics queries at a high level of abstraction;
- developed tools for easily combining such queries into complex workflows;
- integrated a set of new and existing execution engines that execute all parts of a workflow over large data sets that may span various sources and stores, and have irregular dependencies;
- developed an adaptive scheduler that adaptively models the workflow computations and optimally schedules their execution;
- developed tools for monitoring the execution and allowing the adaptation of running workflows by the analytics expert;
- developed tools for visualizing and understanding the results and integrated them into the workflow, to facilitate and maximize the extraction of value.

The outcome of the project is a *modular, open-source system* that offers a unified way for the rapid development, efficient execution, online monitoring and adaptation of complex analytics workflows with arbitrary dependencies over heterogeneous, irregular or unstructured data. The platform developed is functional and immediately applicable; to demonstrate that, the consortium applied its analytics platform to two real-world analytics applications used by the consortium industrial partners; one in the area of business analytics on telecommunication data, and one in the area of web analytics.

1.3 Scientific and Technological Results

The ASAP project built a unified, open-source execution framework for scalable data analytics. The main idea behind ASAP is that (i) no single execution model is suitable for all types of tasks; (ii) no single indexing and data-store is suitable for all types of data; and (iii) an adaptive system that has correctly modeled analytics tasks, costs and is able to monitor its behavior during tasks is a more general, efficient way of tackling this problem.

Complex analytics applications combine multiple operators that process data in various formats, possibly from multiple data stores and data sources. The ASAP platform can be used to support the collaboration between:

- *Operator Developers*: Expert programmers that design and implement analytics operators.
- *Workflow Designers*: Domain experts that combine operators and data sources to construct complex workflows.
- *Users*: Non-expert workflow users, such as marketing experts trying to discover trends, media coverage, or customer data.

ASAP connects these roles by providing a common platform for the management of analytics workflows at all these levels of abstraction. In terms of platform development, ASAP was implemented by the project partners by means of a portfolio of interconnected components:

1. A *Workflow Management Tool* (WMT) that allows the analytics user to see high-level descriptions of operators and data stores, design abstract workflows, and optimize them.
2. A new *Operator Definition Language* that help developers express irregular operators, not supported by previously existing analytics engines.
3. An *Intelligent Resource Scheduling* (IReS) platform that profiles operators, learns their cost, selects optimal plans for workflow materialization, and executes them.
4. The *Spark-Nesting* execution engine that extends the Spark execution engine with support for full recursion and a scalable, distributed scheduler.
5. An *Online Monitoring and Recalibration Platform* that enables workflow designers to adapt executing workflows.
6. *Information Visualization* (InfoViz) services presenting the results of analytics computations to the users.

Each component can be used as a standalone tool and has its own interface as a web application. However, all components are fully integrated to communicate and exchange metadata about workflows, so that they can be easily deployed as the full ASAP workflow management system.

Workflow Management Tool

The Workflow Management Tool (WMT) is a component of the ASAP system architecture that provides a GUI for managing workflows. It is used for workflow creation, modification, analysis and optimization. The model underlying WMT combines simplicity of expression of application logic and adaptation of the level of description of execution semantics. It enables the separation of task dependencies from task functionality. In this way, WMT can be easily used by many types of users, with various levels of data management expertise and interest in the implementation.

Users can use the WMT/PAW module to also analyze workflows, detect errors, simplify or optimize phases, perform editing operations such as substitutions and metadata annotations, as well as apply a novel technique for multi-workflow optimization first developed during the ASAP project. A workflow created in PAW is prepared for execution in three steps: First, the tasks are analyzed and the workflow is augmented with associative tasks; the new version of the workflow, which we call the analyzed workflow. Second, workflows are manipulated by swapping, composing/decomposing and factorizing/distributing transitions, in order to achieve workflows that have equivalent outputs with their original state, but have a form that can result in optimized execution. Third, PAW schedules the execution of a set of workflows following the novel technique of multi-workflow optimization. This technique is based on the joint execution of the common parts of two or more workflows.

The resulting workflows and the operators they include are described using a system-wide metadata language and communicated in this way to the rest of the components in the ASAP system.

Operator Definition Language

Data analytics workloads are consuming large amounts of computation time. As such, a detailed study of their computational patterns and properties is merited. One of the goals of the project was to study these workloads in detail and to develop a programming language that enables programmers to express analytics workloads in such a way that they can be executed at high performance.

We developed the Swan language, an extension of the Intel Cilk parallel programming language, that facilitates high-performance execution of data analytics workloads. Swan adds three features to Cilk, namely: (i) data-flow extensions to express arbitrary parallel patterns and enable virtualization of memory; (ii) a scheduling hint for fine-grain parallel

loops, and (iii) a scheduling hint to exploit locality-awareness in Non-Uniform Memory Access (NUMA) systems.

To a large extent, this work applies to shared memory systems, i.e., it is concerned with a single node in a data center. It should be noted, however, that servers with terabyte-scale main memory exist and offer a highly competitive alternative for workloads exhibiting frequent synchronization. The work is however not limited to shared memory systems as we apply some of our ideas also to Spark. Moreover, data-flow extensions may be used to orchestrate parallelism also in distributed memory systems without affecting the principles of the programming model. More importantly, when used within the rest of the ASAP system, Swan can be one of the many possible execution engines that offers high performance when the data is of an applicable size, so that IReS will select it as the most efficient execution engine for problems that fit in a single node.

We demonstrated the usefulness of Swan for the data analytics problem by applying it to several analytics workloads. In first instance, we apply Swan to Map-Reduce workloads. Map-reduce workloads are conceptually simple and are easily implemented using parallel loops with reducers. Contrary to popular frameworks such as Hadoop, our reducers have clearly defined semantics and do not require the commutativity property.

Next, we applied the Swan language to the problem of graph analytics. Graph analytics differ from map-reduce problems in many respects. Most importantly, graph analytics problems involve irregular computations and have a high dynamic range of parallelism. In our study, we found that graph analytics are highly sensitive to the organization of the memory system in a server. We demonstrate how NUMA-aware scheduling has a significant impact on the performance of graph analytics.

Furthermore, we demonstrated how several of the map-reduce workloads as well as the graph analytics workloads benefit from the fine-grain scheduling hint. The fine-grain scheduling hint expresses that certain parallel loops have very low operational intensity, i.e., they perform very few computations per byte transferred through the memory system. This property appears in several map-reduce workloads and, due to the high dynamic range of parallelism, also in a significant number of parallel loops in the graph analytics workloads.

Next, we investigated text analytics problems. Text analytics problems again have the property of low operational intensity. In our case study of term frequency/inverse document frequency, however, this property implies that performance is highly sensitive to the organization of the data structures, the memory management of intermediate data and efficiently managing parallelism. On the basis of this, we propose a number of operators and a library that are generally useful in text analytics. Finally, we demonstrated that our proposals for HPTA are also applicable to Spark.

Overall, these case studies demonstrated that Swan is an appropriate programming language to express a variety of data analytics workloads. Rigorous performance evaluation, involving a comparison against state-of-the-art solutions for each of the problem

domains, demonstrates that the goal of high-performance analytics is achievable with Swan, especially when relying on the IReS scheduler to recognize problem sizes and instances that can take advantage of its performance.

Intelligent Resource Scheduling

Big Data analytics have become indispensable to organizations worldwide as a means of extracting significant value out of the enormous amounts of data that stream into their businesses. That, in turn, offers organizations an unprecedented competitive advantage: The ability to identify new opportunities, take educated decisions based on historical facts, render their operations faster and more cost efficient and keep customers satisfied. The volume, velocity and variety of Big Data pose new challenges to analytics, entailing a high degree of parallelism in both storage and computation: Modern data centers host huge volumes of data over large numbers of nodes with multiple storage devices and process them using thousands or millions of cores.

In the landscape of Big Data analytics, multiple and diverse execution engines and data stores have emerged as platforms of choice for specific computation types and data formats (e.g., Apache Hadoop, Spark, Hama, Hbase, etc.). To alleviate the burden of building and maintaining such systems, many of them are currently either offered as-a-service by the most prevalent Cloud providers (e.g., Amazon EMR, Google Cloud, Microsoft Azure HDInsight) or packaged in pre-cooked VM or container images for ease of deployment (Docker Hub). Still, although many approaches in the relevant literature manage to optimize the performance of single engines by automatically tuning a number of configuration parameters (Herodotou 2011, Lim 2012), they bind their efficacy to specific data formats and query/analytics task types.

However, one size does not fit all: No single execution model is suitable for all types of tasks and no single data model is suitable for all types of data. Indeed, modern workflows have evolved into increasingly long and complex series of diverse operators, ranging from simple Select-Project-Join (SPJ) and data movement to complex NLP-, graph- or custom business-related tasks, with varying data formats (e.g., relational, key-value, graph, etc.) and shrinking delivery deadlines. Time constraints aside, analysts may be equally interested in other execution aspects, such as cost, resource utilization, fault-tolerance, etc., and thus need to be able to impose various — and often multi-objective — optimization policies, adding another degree of complexity to an already convoluted problem.

Multi-engine analytics have recently been proposed as a promising solution that can optimize for this complexity (Tsoumakos 2014) and are gaining ground ever since. Cloud vendors currently offer software solutions that incorporate a multitude of processing frameworks, data stores and libraries to facilitate the management of multiple installations and configurations (Cloudera CDH, Hortonworks Sandbox, AWS Databases). This is where the ASAP project comes into place: it leverages the power and opportunities offered by

multi-engine environments to harvest Big Data through complex analytics workflows.

One of the most compelling, yet daunting challenges in such a multi-engine environment is the design and creation of a meta-scheduler that automatically allocates tasks to the right engine(s) according to multiple criteria, deploys and runs them without manual intervention. IReS takes over that role exactly within the ASAP project.

IReS is an open-source Intelligent Multi-Engine Resource Scheduler that integrates multiple execution engines and data stores into the optimizing, planning and execution of complex analytics workflows. IReS adopts a black-box approach on the analytics operators. This facilitates the handling of any kind of task, ranging from low- (e.g., join, sort, etc.) to higher-level operators (e.g., machine learning, graph processing, etc.) that run on any state-of-the-art, centralized or distributed system (e.g., Map-Reduce, BSP, RDBMSs, NoSQL, distributed file-systems, etc.). Moreover, the engine-agnostic approach allows for easy addition of new operators and engines. All that IReS requires is a description of the analytics tasks and data via an extensible meta-data framework, as well as a model of the cost and performance characteristics of the required tasks over the available platforms. Consequently, utilizing a DP-based, state-of-the-art planner, the platform is able to map distinct parts of a workflow to the most advantageous store, indexing and execution pattern and decide on the exact amount of resources provisioned in order to optimize any user-defined policy. The resulting optimization is orthogonal to (and in fact enhanced by) any optimization effort within an engine. Moreover, IReS can efficiently adapt to the current cluster/engine conditions and recover from failures by effectively monitoring the workflow execution in real-time.

IReS functionality and components developed within project ASAP include:

- A modeling methodology that provides performance and cost metrics of the available analytics operators for different engine configurations. The resulting models are utilized in multi-engine workflow optimization. The models of the available operators/engines are constructed using the metrics collected from actual executions of the operators both offline (training phase) and online (refinement phase). Thus, the models are refined with every workflow execution, achieving higher accuracy and capturing temporal performance degradation.

- A multi-engine planner that selects the most prominent workflow execution plan among existing engines, data stores and operators, based on a dynamic programming (DP) algorithm. The planner does not only choose the (near) optimal execution plan but also elastically provisions the correct amount of resources, consulting the cost and performance models of the various operators.

- An extensible meta-data description framework for operators and data, which allows IReS to automatically discover all alternative execution paths of an abstractly described workflow by matching operators that perform similar tasks.

- An execution layer that enforces the selected multi-engine execution plan. The execution layer actively monitors the selected multi-engine execution plan, allowing for fine grained resource allocation control and fault tolerance.
- Our open-source prototype has been extensively evaluated over various real-life and synthetic workflows chosen to include diverse datasets and computation types under realistic conditions. The results attest the ability of IReS to efficiently decide on the optimal execution plan based on the optimization policy and the available engines within a few seconds, even for large-scale workflow graphs, adapt to changes in the underlying infrastructure and temporal degradation with minimal overhead and, most importantly, speed-up the fastest single-engine workflow executions up to 30% by exploiting multiple engines.

Spark-Nesting Execution Engine

Modern analytics queries consist of complex computations operated on massive amounts of data. Those queries are impossible to execute on a single node, due to limitations in the CPU frequency and the memory capacity. Thus, the data have to be distributed across a cluster of nodes and processed in parallel. Conventional execution engines are not aware of cluster parallelism, and message passing runtimes like MPI offer precise control and great performance benefits, but the API they provide is very primitive to express complex applications. By restricting the programming model to only map and reduce, or equivalent operators, Map-Reduce clusters scale out because they do not need to track task dependencies, have simpler communication patterns, and are tolerant to executor and even master node failures. However, this simplified programming model cannot easily express some applications, including applications with nested parallelism or hierarchical decomposition of the data. When faced with such algorithms, programmers often develop iterative versions that translate recursion into worklist algorithms. This may be inefficient as it introduces unnecessary barriers from one iteration to the next, and can be unintuitive and complicated to code.

Project ASAP extended the Apache Spark Map-Reduce engine to directly support such nested and recursive computations. Spark is an implementation of the Map-Reduce model that outperforms Hadoop by packing multiple operations into single tasks, and by utilizing the RAM memory for caching intermediate data. We target Apache Spark because it is a widely used, efficient, state-of-the-art platform for data analytics, and currently the fastest-growing such open-source platform.

We performed an extensive evaluation of the resulting Spark engine on 3 different cluster environments and compared our extensions against built-in operators implemented without nesting. We adapted several operators from the WIND application to use our Spark and show measurable benefits to performance and scalability. To demonstrate the generic usability of the programming model extension beyond ASAP applications, we im-

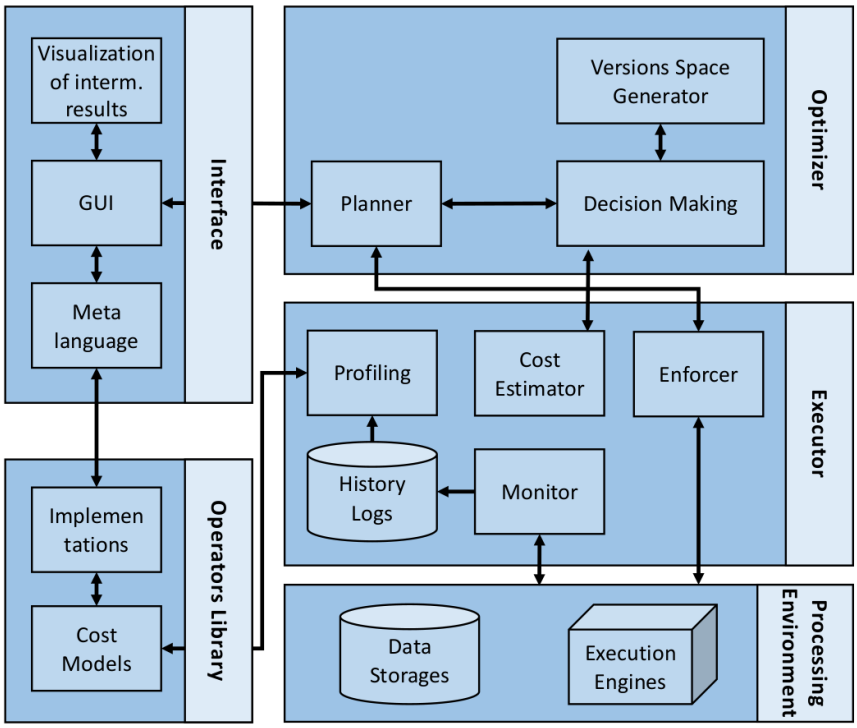

Figure 1.1: The architecture of PAW and its interaction with IReS

plemented an N-Body particle simulation using the nested RDD mechanism. Moreover, we modified the default Spark task-scheduling mechanism so that it can support many parallel light schedulers. We measured its performance against the default Spark scheduler, and found a speedup of up to 2.1$\times$ for computations using fine-grain tasks.

Online Monitoring and Recalibration

The Online Monitoring and recalibration module, integrated with WMT/PAW, enables the analytics expert to change the original task or workflow by altering the task parameters or infusing new tasks, while they monitor the progress of processing in terms of data accessing and resource utilization based on input from the runtime machines or the integrated ASAP visualization tools. The module includes novel techniques for (a) manually changing a workflow at runtime and re-executing it avoiding repeated computations, called recovery and monitoring points technique; (b) automatically changing a workflow at runtime based on conditional structures if-then-else and goto statements that enable the construction of dynamic and adaptable analytics workflows within the ASAP system.

PAW implements a novel workflow model. A workflow is a directed, acyclic graph

(DAG); the vertices represent data processing tasks and the edges represent the flow of data. Each task is a set of inputs, outputs and an operator. Data and operators need to be accompanied by a set of metadata, i.e., properties that describe them. Such properties include input data types and parameters of operators, the location of data objects or operator invocation scripts, data schemas, implementation details, engines etc. PAW is an integrated part of the ASAP system, but it can also stand as an independent tool for workflow management and optimization. PAW enables workflow design by users with various expertise, the automation of workflow analysis in order to clarify and specify execution semantics, single and multiple workflow optimization with respect to time efficiency, over a diverse collection of data stores and processing engines, monitoring of workflow execution and manual and automatic workflow recalibration. Figure 1.1 depicts the architecture of PAW, as well as its interaction with the rest of ASAP. PAW consists of four layers: Operators Library, Interface, Optimizer, and Executor. These provide for workflow design, optimization, and execution dispatch, respectively. Workflows are executed on a set of execution engines and storage repositories of the multi-engine environment.

Operators library This library contains operators, and their corresponding implementations with cost functions. The operators are classified as, either logical operators, which perform the core analytics jobs over the data, or the associative operators, which serve as "glue" between different engines and perform move and transformation operations. The recalibration module has supplemented the library with several operators, called recalibration points.

Interface The GUI allows users to interactively create and/or modify a workflow, and add new operators to the Library. The user designs a workflow graph in the interactive tool and describes data and operators in the Tree-metadata language, which captures structural information, operator properties (e.g., type, data schemas, statistics, engine and implementation details, physical characteristics like memory budget), and so on. The metadata tree is user extensible. To allow for extensibility, the first levels of the metadata tree are predefined. Users can add their ad-hoc sub-trees to define their custom data or operators. Furthermore, the interface allows users to observe the process of execution and intermediate results of a workflow for the recalibration needs.

Optimizer The orchestration of the optimization process is performed by the Planner. It takes as an input a workflow from the Interface and sends it to the Decision Making module, which returns an optimized version of a workflow. All possible versions are produced in the Versions Space Generator and their costs are estimated by the Cost Estimator. The Decision Making module chooses the version with the minimal cost as an optimal one.

Executor The executor performs several tasks. The Enforcer schedules workflows for execution, generates executable code and dispatches workflow fragments to execution engines. The Monitor observes the system state, tracks the progress of executing workflows and stores History Logs of runs. These logs are used to construct more precise cost functions of operators through the Profiling module. As an execution system, PAW uses IReS.

InfoViz Visualization Services

The interactive visualizations of the InfoViz module are intended to support free insight generation without prior modeling of a domain, embracing both unstructured (Web intelligence) and structured (linked data) sources. Shneiderman et al. (1996) present a taxonomy of data types in the context of visualization, including temporal and multivariate data.

ASAP dynamically combines such data types on the fly, taking into account the use case specifics and current user tasks. Time is a particularly important dimension to consider, and was as such a focus of the work conducted. The resulting interactive visualizations should (i) reveal complex patterns and evolving trends, (ii) provide flexible mechanism to select appropriate timescales, and (iii) are capable of rendering a considerable amount of data spanning multiple sources and significant timescales — without relying solely on aggregation, which might hide important facts.

Interactive Visualization of Heterogeneous Data from Multiple Sources

Visualization is an effective means to help analysts make sense of the current data deluge. Quite often it is not enough to design a single visualization, but rather a set of visualizations to get a better sense of hidden patterns and trends, as different images might send different signals to the user. If this discovery process is to be effective, visualization components need to be able to use large quantities of data from heterogeneous data sources. A telecommunications analyst who wants to visualize call metadata from various cities, for example, might require additional information besides call metadata; e.g.:

- news or social media coverage about the observed cities to correlate peaks in the number of calls with co-occurring events such as music concerts, sports events or political campaigns;
- population statistics, GDP data or statistical indicators from the respective cities, assuming such datasets are available in an RDF or JSON format;
- patterns that correlate call metadata (aggregated and anonymized) with statistical data and/or news and social media coverage.

In order to be able to cope with such requirements, a state-of-the-art visualization engine and dashboard needs to include not just a set of appropriate visual methods, but also components that support:

- the parallel processing of a wide variety of data types, including semantic data types like geolocation, dates, etc.;
- the remix of data from a wide variety of data sources regardless of domain, structure (structured or unstructured), or provenance;
- the possibility to extract various types of aggregated statistics or the most important entities, and means to select, sort and summarize the data.

Data Matching and Integration

Without an integrated back-end that provides such services, any visualization service will not reach the scale and depth needed to create on-the-fly visualizations from heterogeneous data sources. Among the most complex problems that such a back-end needs to solve is the matching of data sources to different visualizations. Several solutions have been proposed, but there is still no widely accepted standard to flexibly integrate and visualize heterogeneous data sources. In general the problem of resolving the entities from different datasets to the same common real-world entities is known as data matching or record linkage since the 1960s (Koudas et al., 2006). More recent solutions to address the problem of flexible integration for automated visualization include schema matching (Cammarano et al., 2007), ontology based information extraction and integration (Buitelaar et al., 2008), data wrangling (Kandel et al., 2011) or proactive wrangling (Guo et al., 2011) via interactive data transformation scripts, scalable data curation (Stonebraker et al., 2013), human data wrangling following a crowdsourcing approach (Clow, 2014), as well as visual embedding and visual product spaces (Demiralp et al., 2014).

In addition to these automated visualization models, a number of models mostly focused on automated visualization of structured data (and especially Linked Data) include Ontology Based Data Access (Giese et al., 2013), Linked Widgets for exploiting governmental Linked Data (Trinh et al., 2013), LDVM or Formal Linked Data Visualization Model (Brunetti et al., 2013), universal data cube concordance model (Kelleher, 2014), and OLAP4LD — which is a generalization of OLAP for various types of linked data (Harth et al., 2014).

Despite the availability of various models and related tools, the field of data matching is still in its infancy. This also suggests that there is still a lot of work to be done in order to get to a place where automated visualization systems are mature and flexible enough to visualize any type of heterogeneous data source on the fly. Interoperability, speed and scalability are obviously other factors that need to be taken into account when designing such a system.

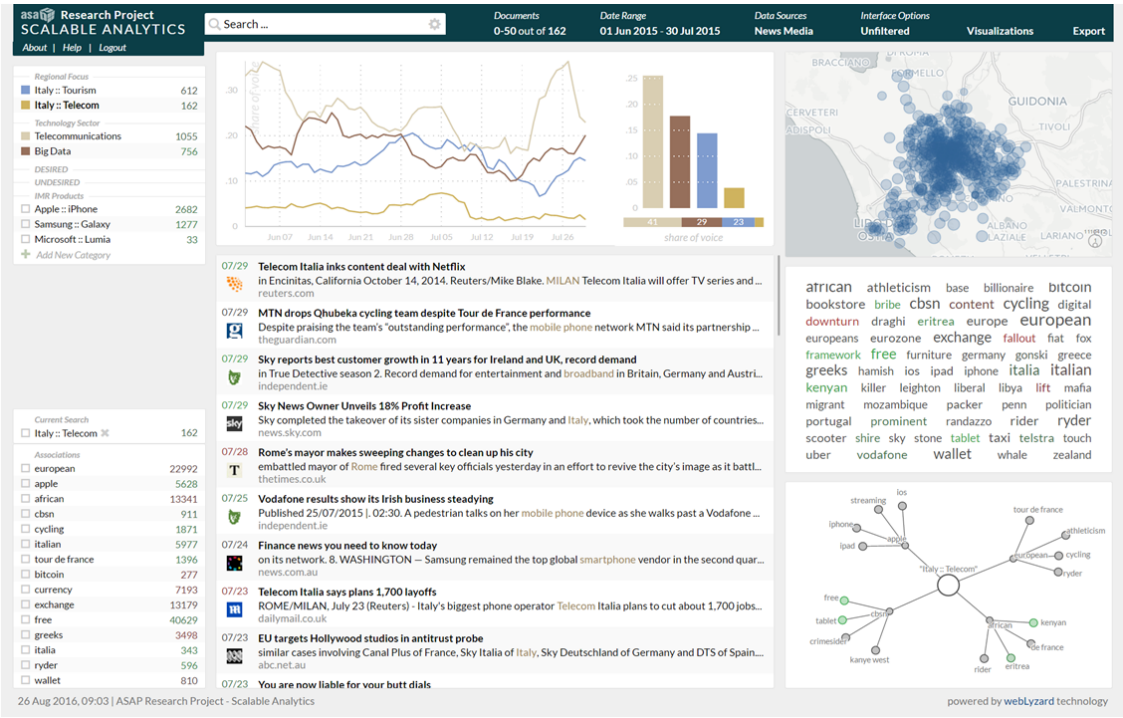

Figure 1.2: Screenshot of the ASAP dashboard prototype

Visual Dashboard to Explore Contextualized Information Spaces

ASAP integrates real-time data feeds from multiple sources via an open API, which allows uploading structured and unstructured datasets, searching for specific sets of indicators or documents, and embedding individual visualizations in Web-based applications, to be rendered in real time.

The ASAP dashboard combines several visualizations to represent the contextualized information space using a multiple coordinated view approach. Synchronized widgets show the various metadata dimensions (see screenshot in Figure 1.2; the lightweight look and feel reduces complexity and highlights the actual content). These widgets help explore the storytelling potential of big data visualization and address calls for methods to support the complementary relationship between the explorative and analytical dimensions of information visualization. The ASAP dashboard builds on insights gained from the Media Watch on Climate Change, which initially served as a rapid prototyping platform to develop the widget synchronization, as a means to gather feedback from non-expert users, and as an outreach channel. Figure 1.2 shows a screenshot of the actual ASAP dashboard prototype.

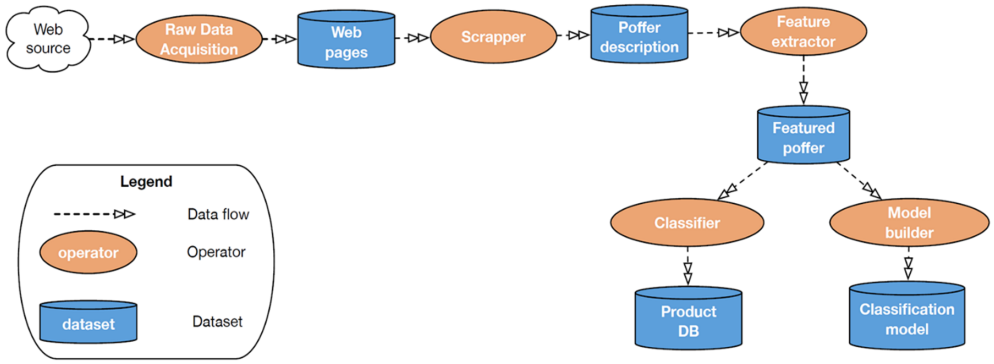

Figure 1.3: Workflow of WP8 Web Content Analytics

Web Content Analytics

The use case is centered on the information services of Internet Memory Research (IMR) as part of the Mignify platform. IMR collects, cleans and classifies data from Web, and uses the results to support its online services.

Analytic Goals The general goal of the IMR data collection, extraction and classification processes is to build and maintain a catalog of product references, and to discover product offers related to this catalog on public marketplaces.

- A Catalog is a tree of categories and subcategories. An example of category is Coffee machines, and a sub-category is Espresso machine.
- A Product Offer is an online proposal to sell one or several items of a product, with specific conditions such as price, delivery, etc. If, for instance, an electronic marketplace proposes 100 items of the coffee machine xxP34, at a given price YY, this constitutes a product offer for product xxP34.

Figure 1.3 shows the overall workflow of WP8. IMR maintains a Web map of classified sites that references hundreds of thousands of marketplaces. Our crawler scans the pages and identifies those that contain lists of products. Thanks to a semi-supervised approach, we then analyze the page structure and produce a wrapper to extract a product offer record. This integrates a rich set of product-related information including brand, type, price, textual description, and user comments. Once product information has been extracted from the page, we run a classification process to predict the category of the product. The product matching operation associates product offers with product categories, given the description of offers extracted from e-marketplaces.

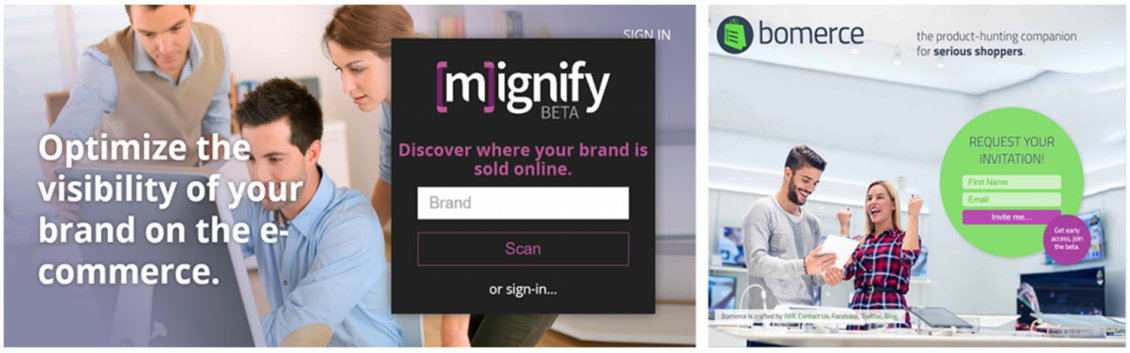

Figure 1.4: WP8 Applications Presence and Bomerce

Applications IMR maintains and expands a large database of classified product offers. This ProductDB database supports several services which can be split in two main categories depending on their target users:

- *Presence* is a B2B service that takes advantage of the ProductDB database to provide competitive intelligence. The service offers a public interface that lists of eCommerce sites where a specific brand can be found. Sellers can get specific data on their brand and analyze their main competitors.

- *Bomerce* is a price comparison application for Web and mobile devices. When exploring online offers for products or services, users are confronted with heterogeneous offers from proprietary eCommerce sites. Bomerce helps to compare such offers and seek third-party advice, sends out of notifications in the case of promotions, and checks the reputation of an eCommerce site.

Both Presence and Bomerce depend on the quality of the ProductDB database, and therefore on the data acquisition, extraction and classification workflow. We modeled and implemented this workflow with ASAP. First, an abstract workflow has been defined as a high-level view of the various steps involved in the transformation of raw Web pages into structured and classified product descriptions. This abstract workflow is then implemented through concrete operators taken from the ASAP library.

Interactive Exploration The ASAP Dashboard shown in Figure 5 demonstrates the potential of big data technologies in conjunction with advanced visual analytics to automatically transform noisy and unstructured Web content into valuable repositories of actionable knowledge. Processing dynamic content streams from multiple sources and extracting metadata attributes from the product offers, it extends IMR's price comparisons by:

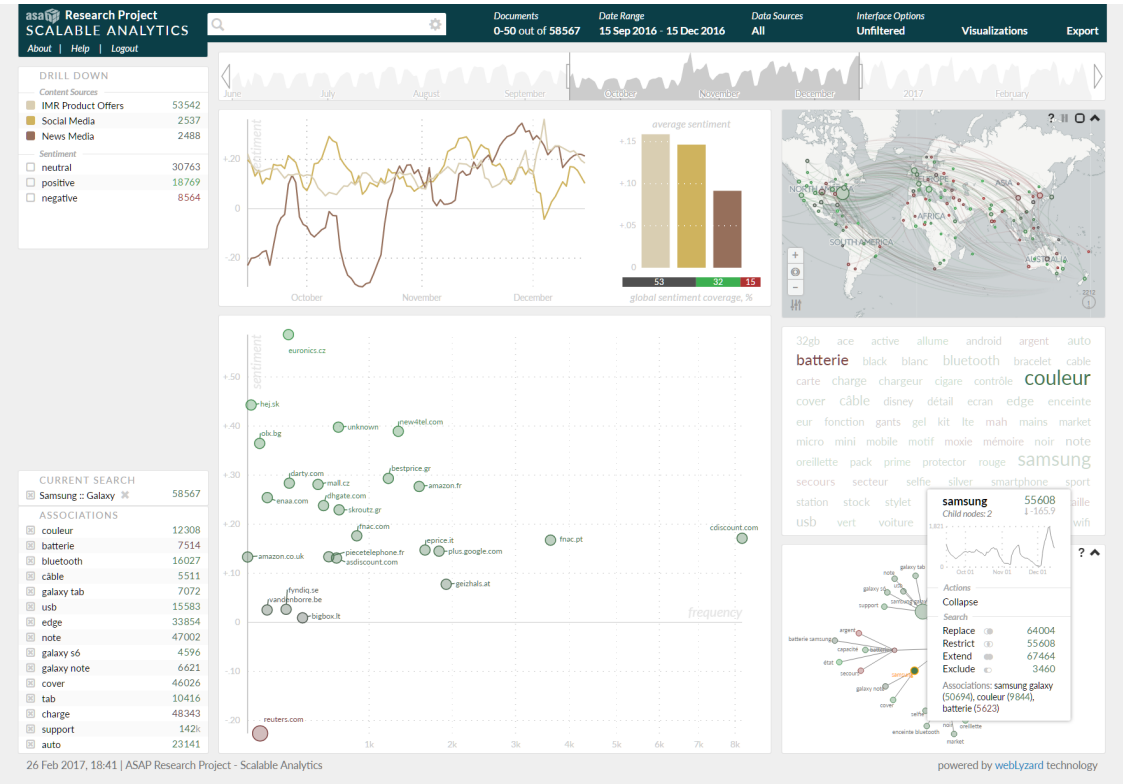

Figure 1.5: Screenshot of the ASAP dashboard — drill down sidebar to compare product sentiment for the Samsung Galaxy series by source; scatterplot for cross-media analysis; geospatial projection of referenced locations; tooltip for on-the-fly query refinement

- Visualizing aggregated keywords computed from noisy textual descriptions contained in the product offers collected by IMR; identifying the leading sources of these offers, including an analysis of keywords that e-commerce sites associate with specific products or an entire product category.

- Exploring product features that impact the perception of a product in online media coverage (news channels vs. social media vs. product offers), creating additional value for sales and marketing decision makers.

- Providing metadata dimensions such as sentiment, which indicates whether a feature is mainly perceived as a unique selling proposition that causes satisfaction, or a hygiene factor that causes dissatisfaction. This distinction is an important source of feedback to guide strategic marketing decisions.

For a business intelligence tool to complement price comparisons, such metadata

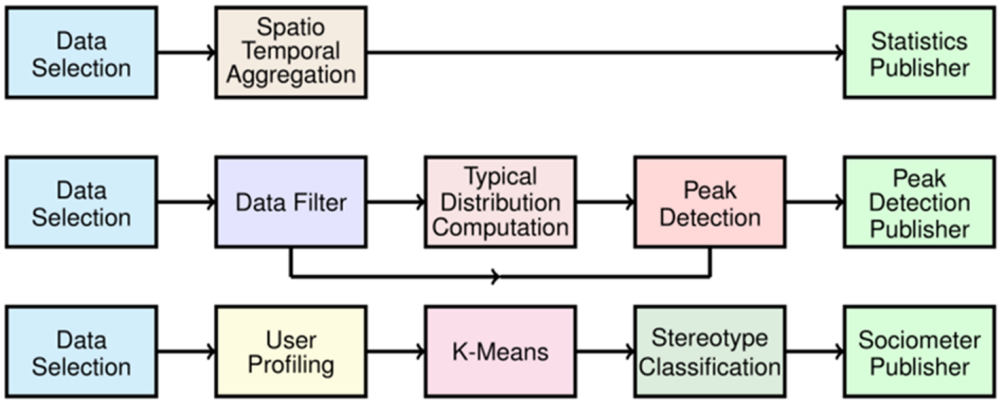

Figure 1.6: Workflow of WP9 – Telecommunication Data Analytics

dimensions are particularly important. The drill-down sidebar of the ASAP dashboard shown in Figure 1.5 helps to better understand the temporal distribution of metadata attributes. The line chart in the shown example compares the average sentiment for the Samsung Galaxy series by source (product offers, social media, news media). The bar chart presents the same data in aggregated form, and the scatter plot maps the frequency vs. sentiment matrix of the major content sources. The geographic map projects the entire set of search results. The adaptive tooltip in the lower right corner enables on-the-fly query refinements — either to Replace the search query with a new term, or to apply the Boolean operators AND (Restrict), OR (Extend) and NOT (Exclude). Using the dashboard's view synchronization mechanism, the tooltip is tightly coupled with the tag cloud — which highlights the product features "color" and "battery" as the strongest associations with the hovered keyword "Samsung".

Telecommunication Analytics

The usage of mobile phones and the resulting datasets stored in Call Data Records (CDR) represent a high-quality proxy to better understand human mobility behavior in different application scenarios such as smart cities, transportation planning and environmental monitoring.

Analytic Goals Mobile phones communicate to antennas covering specific local areas. The active connection between a phone and an antenna, e.g. a phone call or a text message, represents spatio-temporal information that, once collected and aggregated, reflects the distribution of users in an area covered by mobile services. The analytic goals of ASAP focused on better understanding such aggregated data — applying strict data anonymization procedures and addressing the involved computational challenges.

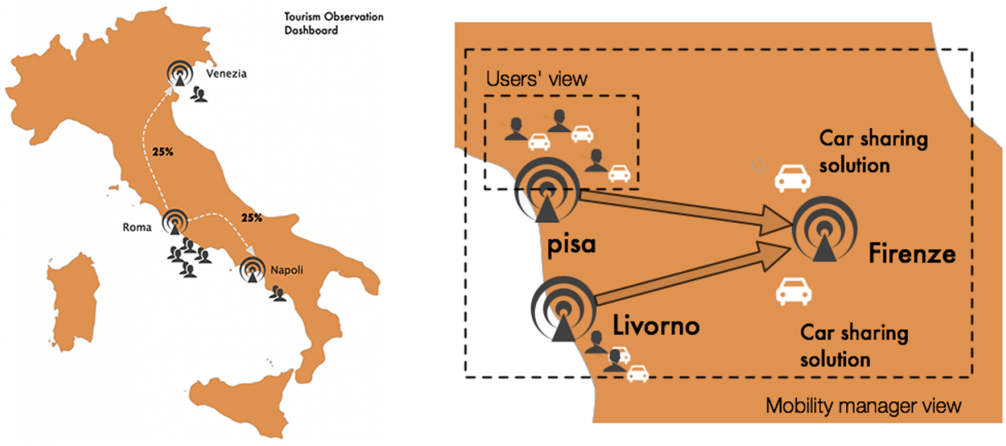

Figure 1.7: WP9 Applications – Tourism Observation Board and Mobility Manager

Within ASAP, a telecommunications application was designed to demonstrate how analytical services based on human mobility data can be provided during routine mobile network operation. Figure 1.6 shows the workflow of selected modules of this application, including user profiling, sociometer and peak detection.

Applications The sheer volume of CDR data poses computational challenges when collecting, storing, mining, and visualizing specific indicators. In this context, ASAP investigated the following applications (see Figure 1.7 for two conceptual diagrams):

- *Event Detection* analyses the different features of an event, including its spatio-temporal characteristics, social aspects, and statistical properties. By controlling input parameters such as time interval, spatial area and additional CRM attributes, analysts gain a detailed understanding of evolving events.

- *Ridesharing* provides functions for mobility managers and individual drivers alike, e.g., the visualization of routine trips in a specific area, together with an optimized car sharing solution for managing such trips. A driver can use this application as a recommender system to identify ridesharing opportunities.

- *Tourism Observation* The analysis of dynamic tourist flows allows mobility managers to identify common movement patterns of visitors, using a map-based dashboard to provide spatio-temporal constraints as input. Along this line, mobile phone data, despite their limited spatial precision compared to other location data such as GPS, are of interest due to their global availability across countries, and their independence from specific means of transportation. We can use this kind of data to reconstruct mobility and traffic flows by using origin / destination matrices, complemented by

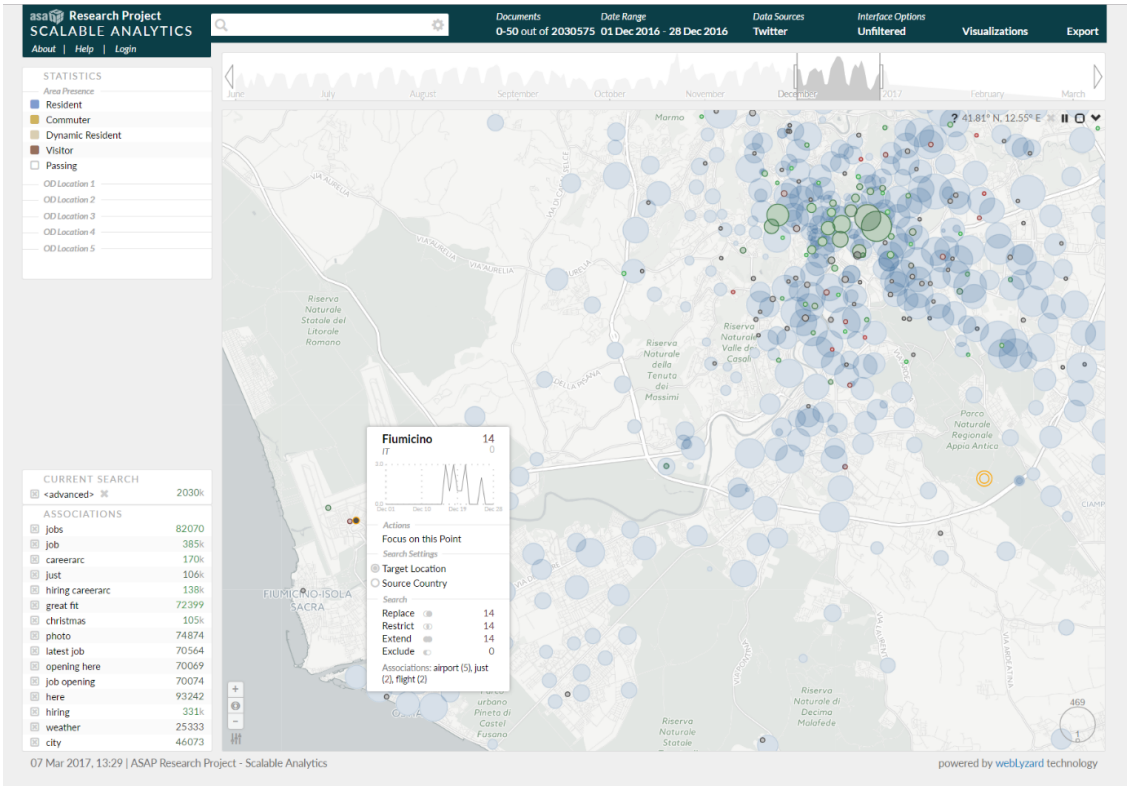

Figure 1.8: Geographic map of the ASAP dashboard, showing anonymized Call Data Records (blue markers) for Rome, and an overlay of sentiment-annotated Twitter content (green = positive; red = negative coverage) as of December 2016

other sources such as CDR-based traffic intensity, traffic migration from handovers, and individual movements of volunteer participants.

Interactive Exploration The ASAP dashboard illustrates how actionable knowledge can be visualized from the ingested statistical indicators — to understand weekly patterns of user movement, for example, or to visualize the interplay between multiple indicators by generating an overlay of social media content on top of aggregated CDR data.

To index and visualize the statistical data produced by the ASAP workflow —area presence by user type (resident, dynamic resident, commuter, visitor or passerby), O/D matrices, etc.— a Statistical Data API was developed following the RDF Data Cube Vocabulary approach. It supports JSON to enable rapid visualization of large datasets. The desired slices of a dataset, e.g., the residents who visit a certain area, do not need to be defined within the dataset, but can be specified at runtime.

Tested with queries that delivered up to 100 million documents, the geographic map

of the ASAP dashboard has been optimized for large datasets. Figure 1.8 shows the maximized version of the map including a tooltip for on-the-fly query refinements.

Users can zoom to street level and visualize anonymized cell tower activity data from the City of Rome (blue markers), and overlay this information with geotagged Twitter postings (green = positive sentiment; red = negative sentiment) —e.g., to identify communication hotspots during an event. The examples are intended as a proof of concept how semantic technologies in conjunction with advanced visual tools can transform statistical data into valuable repositories of actionable knowledge.

Overall Foreground

The ASAP consortium has developed a wide array of open source tools and libraries that make the management of analytics applications that are structured as workflows, easier to develop, faster, less error prone, cheaper, and overall better to quickly adapt to big data that is evolving fast. We delivered well documented, open source tools on public repositories, along with use-case examples of two industrial-scale applications that demonstrate the applicability and impact of ASAP technology on workflow analytics applications.

1.4 Potential Impact

The main objectives of the dissemination strategy adopted in project ASAP are:

- To maximize visibility of the project and all members of the consortium among key stakeholders in the big data analytics field.
- To raise awareness of the problems that the project targets and project the project results and solutions to a wide audience of potential end users.
- To engage key groups such as graduate students and junior engineers that have a high probability of trying new skills and transferring technology to businesses in the near future.
- To influence the industry in terms of reducing the cost and improving effectiveness of the solutions and solution methods they adopt.
- To influence de-facto standards and widely-used applications and introduce code developed within project ASAP to existing open-source solutions.
- To exploit the project outcomes with the industrial partners.

The ASAP consortium dissemination effort has tried to attract the attention of the following audiences to the general issues addressed by the ASAP project:

- ASAP Consortium partners: Researchers, Graduate Students, Engineers, Senior Management, Marketing Experts
- Scientific Community
- Research Organizations
- Business audience: Management, Research and Production Engineers, Sales
- Cloud computing providers
- Analytics tools open-source developer community
- Related EU project consortia
- Policy makers
- General public

Competitive Analysis

The ASAP project is one among many research project consortia, companies, research labs, and other organizations currently exploring solutions to the increasing complexity of the analytics ecosystem. By attending conferences and networking events related to the community, literature review, and monitoring social media, the ASAP consortium maintains an up-to-date view of the state-of-the-art, as has been updated –per area– in the corresponding deliverables of the research Work Packages.

Overall, there have been attempts for the development of analytics workflow solutions; CloudFlows is a Horizon 2020 project that provides a UI for designing analytics workflows, although without support for dynamic workflow adaptation or integration with a cost-modeling scheduler. Zoe is a project of the Eurecom Research Lab that integrates Spark and Tensorflow with support for workflows, although it does not use adaptive scheduling to optimize resource costs or select alternative execution engines. R AnalyticFlow is an open source module for the R language that manages analytics workflows in a single node. Apache Oozie and Python Luigi are open source tools for workflow execution; both assume workflows as DAGs of operators and schedule them on available resources, although without optimizing for data sources or supporting cost models for optimal scheduling. Companies such as Composable Analytics, IBM MobileFirst offer similar solutions to custom workflows; in this case changes to the analytics application may require costly external support by the solution providers or consultants. Project ASAP offers a state-of-the-art solution for the design, optimization, cost-efficient scheduling, execution, monitoring, adaptation, and visualization of complex analytics workflows that combine multiple execution engines.

Strategic Impact

The ASAP project is strategically positioned for maximum impact on the analytics business and research in Europe and internationally.

Major realized impact points of the project are:

- Better European positioning on the existing analytics ecosystem. Project ASAP software targets existing analytics projects using open source code contributions to build up credibility, attract users that require new features (such as irregular graph analytics) offered by ASAP tools, and gain expertise.

- Improvements on the state-of-the-art in performance. Project ASAP has produced peer reviewed publications that measure performance improvements over the state-of-the-art in high-visibility venues.

- Creation of open-source ecosystem. Project ASAP software modules already are referenced in Horizon 2020 proposals and future consortia. We expect the develop-

ment and improvement of the individual tools will continue and plan to maintain their integration as a complete workflow management platform.

Societal Impact

Project ASAP has influenced the following societal domains:

- Analytics Services. Project ASAP enables better efficiency of workflow execution by service providers, enabling economies of scale for co-hosted analytics applications. We expect industry to uptake ideas, solutions, and software developed and published by the project.
- Transportation planning. The telecommunications analytics application developed and deployed using the ASAP system has already produced useful information regarding traffic and behavioral patterns of residents in Rome.

Industrial Impact

Project ASAP has created a valuable set of tools and made them open-source and available to the industry. Numerous industries can benefit from the produced foreground:

- Telecommunications Analytics. Project ASAP has developed an open-source library of operators for the analysis of telecommunication information. Within the project these were used for the industrial use case, but could have a wider application in telecommunication analytics.
- Graph Analytics. Project ASAP has created a reusable library of graph analytics operators that outperform the state-of-the-art in many cases. With the rise in graph-like big data applications such as social networking business applications, we expect the graph algorithms developed —already integrated with popular engines— to find use beyond the ASAP application use cases.
- Finance. Financial applications often target low-latency performance and may involve streaming data and static data sources, creating complex workflows with different data sources and computations. The IReS scheduler is adaptable to cost function modeling, and thus can optimize such complex workflows for latency instead of resource cost or total execution time.

Path to exploitation

Due to the generic nature of the project results, especially the ASAP unified programming model and distributed computing engine, exploitation activities went and will go beyond

a specific industry and beyond the defined uses cases. Tailored exploitation actions are targeted at companies already collaborating with partners of the ASAP consortium (in various domains, not exclusively those covered by the two use cases), working groups and standardization bodies, and other stakeholders with an interest in big data technologies such as policy makers and NGOs.

Specific exploitation plans by academic and business partners include future research projects, new product development, and development of internal analytics processes.

FORTH

The CARV laboratory of the Institute of Computer Science at FORTH has expertise in runtime systems, memory management, distributed systems, and languages, and brought this expertise to a focus on the problem of distributed Analytics computations within the ASAP project. With ASAP, FORTH has strengthened its expertise on several aspects of Big Data computations (scheduling, placement, programming models for expressing these computations, as well as compilation and execution of big data queries). The CARV laboratory plans to continue its research on related problems and apply ASAP technology to new Big Data application domains.

We plan to maintain and continue improving on Spark-Nesting, the recursive-query execution engine developed within ASAP that is based on Spark. We have already located several other applications from astrophysics, biology, and natural language processing that will benefit from the extended programming model developed. Moreover, we plan to use the other ASAP modules internally, as we found they facilitate workflow management for domain experts that are not computer scientists and have assisted us in collaborations outside the ASAP project.

We plan to exploit expertise and collaborations developed within ASAP in the future. Specifically, we would like to look into application domains other than bulk/batch analytics computations as we expect that many of the optimizations developed within ASAP will also be applicable in streaming and low-latency/real-time domain applications. We will explore collaborations with the ASAP partners, targeting problems in embedded systems (Big Data from IoT) and financial applications (mixing big data-warehouse with continuous stream processing).

UNIGE

Working in the ASAP project gave the collaborators from UNIGE the opportunity to realize their innovative vision in terms of research and engineering for a powerful platform for the management workflows on Big Data analytics. UNIGE intends to continue working in extending and improving this platform, as well as test it on real workflows of new use cases; the latter may come from the domain of bioinformatics and astrophysics, as UNIGE has a close relationship with the Swiss Institute of Bioinformatics (SIB) and the European

Organization for Nuclear Research (CERN). Furthermore, UNIGE intends to exploit the acquired research and engineering experience from ASAP, in order to initiate new efforts for the creation of project proposals on Big Data management, to be submitted to the EU and to the Swiss National Science Foundation (SNSF).

ICCS

The Computing Systems Laboratory of ICCS has a strong expertise in the areas of Big Data and Cloud Computing, particularly focusing on Big Data management, performance aspects of Big Data Analytics and elasticity of Cloud Computing platforms. Within ASAP, the ICCS team exploited this expertise to design and implement the Intelligent Resource Scheduler (IReS), a platform that optimizes, plans and executes complex Big Data analytics workflows in multi-engine environments. More specifically, in the course of the project issues of elastic resource provisioning, management of large volumes of intermediate data, scheduling and execution of tasks in Cloud environments have arisen and prior work of ICCS members in these areas has helped to effectively resolve them.

Moreover, thanks to ASAP, ICCS has expanded its expertise in the areas of (a) performance modeling of distributed runtimes and data-stores, (b) management of Big Data analytics workflows throughout their lifetime and (c) multi-engine environments. The lab members have already expanded their research activities to these areas and will continue conducting research in these fields after the end of the project. ICCS has already appointed 4 ASAP-related diploma theses, which have been successfully completed, allowing undergraduate students to familiarize themselves with the related research areas. ICCS plans to maintain and continue improving IReS, extending it to support streaming engines and their special characteristics. We also plan to enhance IReS with more advanced optimization capabilities for special types of data/queries (e.g., Sparql queries over RDF data). The first step towards this direction has already been made with MusQLE, a side system that specifically caters for the optimization and execution of SQL queries over multiple engines.

ICCS will exploit the work of ASAP in the following ways: (a) by conducting research in the new fields of expertise gained through ASAP, (b) by collaborating with the project partners in research activities as well as future potential research projects, (c) by internally using parts of the ASAP platform both for our research activities and for managing Big Data related laboratory exercises conducted by undergraduate students as part of the “Distributed Systems” and “Advanced Topics on Databases” courses of the National Technical University of Athens and (d) by exploiting the newly gained expertise in the ongoing projects SELIS and ACTiCLOUD.

QUB

The High-Performance and Distributed Computing group at QUB has expertise in parallel computing, programming languages and runtime systems. In recent years, it has expanded its core research from the high-performance computing domain into data analytics, cloud computing, and recently fog computing as modern areas where computing at the physical limits of the system (e.g., peak performance, energy constraints) allow application of our core knowledge. Concomitant with this move, the group has become part of a newly created Center for Data Science and Scalable Computing in the ECIT-2 institute at QUB. The ASAP project, but also related EU projects such as CACTOS, RAPID, UniServer and VINEYARD, have been effective vehicles for QUB to expand its knowledge in the area of data analytics and data center applications. ASAP in particular, has provided new knowledge to us around the computational and data management challenges in various types of data analytics workloads. It has provided us with a vehicle to further develop our existing work on parallel programming languages and runtime system support, compilation and runtime scheduling algorithms.

In the future, QUB will build on its acquired knowledge to develop a position as a worldwide leader in high-performance analytics. It will feed its acquired knowledge into running projects and also develop new research projects in this area. It will continue development of the Swan programming language using use cases from high-performance computing and data analytics. It will moreover disseminate its experience through undergraduate teaching. In particular, 4 final-year projects investigating the technologies developed in ASAP have been completed and 4 are in progress. QUB will sustain research collaborations with ASAP partners aiming to further develop the ASAP technologies.

Internet Memory Research

During the project, IMR has reorganized its business activities towards structured data extraction at large scale. Crawling, wrapping and classification have been focused on eMarketPlaces, and the product-related information obtained from the process are currently used as a support for B2B and B2C services. Deliverable 8.4 introduces two such services that have been used during ASAP to evaluate the data processing modules produced by our partners. They are based on complex data analytics workflows whose design and evaluation has been greatly helped by the cooperative work in ASAP. The high-level, platform-agnostic approach adopted by the project has encouraged our big data engineering team to (i) abstract their task at the appropriate level, (ii) view a workflow definition under a modular perspective, and (iii) systematically inspect the behavior of each component, in terms of performance, robustness, and quality. Using these disciplined conception and implementation guidelines has numerous benefits. First, it saves time and efforts when some of the components have to be replaced. It is well known that maintenance and evolution of software is one of the biggest cost in the application de-

velopment process. We decided for instance to change the classification algorithm which gave poor precision results. With a well-defined workflow made of nearly independent component, the task turned out to be much easier than if we had kept a monolithic conceptual perspective. Second, the ASAP interfaces supply a clear and understandable view of a workflow definition that helps non-expert users to understand what is at stake in a workflow execution, and can be used to exploit various materializations of the workflow that aims at other goals than the core objective of performance optimization. We used the materialization mechanism for instance to implement a testing version of the classification workflow that can be used to evaluate a new component before introducing it in the catalogue of alternatives.

In the future, we will benefit from the project's lessons to continue the enhancement of our workflows management. One of our next projects is the adoption of Flink as a distributed execution engine, at least for parts of our data processing workflows. This will involve the replacement of some operators with a materialization based on Flink, and their evaluation to check the benefits of the new platform. We plan to generalize the principle of maintaining a catalogue of independent materialization of a given operator, either for choosing the best alternative in a specific evaluation context, or as a possibility to backtrack to a tested and validated implementation if a new one appears to be not satisfying enough. Overall, we will build on the ASAP lessons to improve our technical approach and reduce the cost of our big data processing tools.

WIND — Tre

WIND contributes to communicating the project results via internal corporate channels as well as through the Web-based channels of the Company's group. Furthermore, WIND exploitation activities are dealing with the impact of the ASAP project in the area of privacy-aware mobility mining to improve our portfolio of services, taking into account the expertise gained. These activities will be enhanced through the use of the ASAP platform, enabling easier large scale data analysis aiming at trend discovery and smart decision making. WIND as a TLC Operator is also interested in Privacy-Aware Mobility Mining to improve its portfolio services according to the legal context: in the new ecosystems the right to the protection of the private sphere must coexist with the right to access to knowledge and to services as a common good. This evaluation may be useful for different situations like the transportation (things/people) reconstruction and optimization and the Urban/Country Territory promotion for a new smart customized solution.

The results obtained from ASAP will allow WIND to define a roadmap to introduce big data innovations. The project's platform will be useful to analyze different scenarios and realize customized solutions according to both B2B and B2C adopted models. This is in line with the privacy-proven approach that enabled new services using a prototyping approach, thereby reducing initial investments. The experience gained from ASAP has allowed WIND to define a number of possible applications areas:

- Beginning with the Telecommunications Data Analytics Application (TDA) and enriching it with the use of CRM data in order to offer to our Business Market solutions that meet the needs of an emerging market such as in the case of tourism applications with digital solutions that contribute to improving the mobile services offer.

- The capability to better configure the customers offer and differentiate itself from other telco operators (user targeted applications/personalized applications and services).

Using the TDA it would be possible to obtain:

- Forecasts of traffic flows linked to events and/or places (Transportation/Urban Mobility/Tourism).

- Improved customer experience since the traffic forecasts for each event signal the need to improve network capabilities to offer better perceived quality (Transportation/Urban Mobility).

- New “digital assistant” which will be offered to enhance customer interaction.

- A competitive advantage by leveraging Big Data, WIND’s “mine”, to expand the range of solutions for an easy “Digital Life”.

- An important innovation in the Italian digital market by using the “privacy proved” data.

- Contributions to the development in the Italian and European markets; big data skills and new professions like data scientists.

As an added value, such data can be mined and correlated in order to help business analysts predict patterns and define efficient marketing and business strategies taking into account the specific context of “smart” tourism management. The use cases give WIND the opportunity to create different components for innovative applications: most efficient solutions to offer good services and a good perceived quality of the connections to gain trust from connected customers. This will be done also by improving and expanding the network capacity in specific areas where a higher user concentration and services usage has been observed as a result of the ASAP framework use.

WIND plans on evaluating the effect of the tourist trend application in order to further invest on similar analytics applications that cover more business sources and also fuse them with the “wisdom of the crowds” (i.e., social data). These are some details on how ASAP with its specific orientation can offer innovative implications to industrial performance.

The strategic and advanced use of Big Data will help WIND to gain information/insight on the success rate of an event and to improve the effectiveness of WIND released services in the context of the selected event.

For example, as part of WIND exploitation activities in ASAP there was a collaboration with the Mobility Agency of the City of Rome in the area of tourism analysis. The collaboration with this entity will allow in the future a practical application of the ASAP framework and of the results coming from the tests done with the ASAP Telecommunication Data Analytics (TDA) application.

The services targeted as a result of this collaboration will be mainly on the discovery of people's preferred activities in specific touristic areas in the City of Rome (POI – Points of Interest). The TDA would be useful to improve the customer experience according to their potential interests while various planned events in their cities happen. In addition, the TDA may be customized for other unexplored commercial contexts.

Feedback from WIND's Big Data Department "Customer Experience and Big Data Analytics" suggested that the use of "thermal" diagrams for the visualization of the activity level of users might be a desirable feature in a series of future applications (similar to the GROOVE visualization reported by webLyzard in Deliverable D6.2). Also concerning other forms of visualization to support some possible applications in WIND there is the so called "O/D" (Origin/Destination) Matrix as explained in the TDA application (refer to Deliverable D9.2 and D9.4). Both visualization techniques in future applications are one of the exploitation results of the ASAP platform in WIND's upcoming services based on Big Data analytics.

Finally, from the point of view of the company, the ASAP activities have allowed to test the ability to work with a multi-disciplinary team to create an organized critical mass capable of handling new services during the phases of design, deployment and management of the solutions offered by placing the focus on learning by doing.

webLyzard technology

In terms of visibility in the relevant research communities, and to attract overseas clients (a crucial factor for an SME focusing on large-scale applications of semantic technologies), the communication activities of webLyzard target government agencies and research centers in Europe and the United States, as well as large business-to- consumer brands. Such brands represent highly valuable assets. They are among the primary exploitation targets and essential for the continued growth of webLyzard, and for achieving its long-term commercial goals.

Consumers who discuss brands on social media not only respond to brand communication, but also play a pivotal role in shaping a brand — e.g., when repeating or commenting on a story via Twitter or Facebook. A deep understanding of this process helps to increase brand performance. Given the volume and complexity of the underlying dataset, visual methods such as those developed in WP6 of ASAP are the best way to convey such

an understanding. Embedded into the ASAP dashboard (D6.4), the visualization components support ad hoc exploration of dynamic datasets (the comparison is not restricted to brands, but can also include other entities such as products, persons and organizations).

Word-of-mouth and active collaboration with large agencies such as the National Oceanic and Atmospheric Administration (NOAA) or international organizations such as the World Bank and the United Nations Environment Programme (UNEP) helped to disseminate ASAP results to an international audience. Improved scalability increases the WLT knowledge base, attracts new clients and represents an important competitive advantage, particularly in conjunction with the new visualization methods of WP6. An early exploitation of developed technologies in real-world applications complemented the ongoing evaluation efforts reported in D6.5.

In Year 1 of ASAP, this included the U.S. Climate Resilience Toolkit 44 developed in response to President Obama's Climate Action Plan. webLyzard provided the Toolkit's search function to help visitors quickly locate the most relevant content across U.S. federal government's Websites. The visualization modules of T6.1 were integrated into the analytics view of this application, which enables communication experts at the National Oceanic and Atmospheric Administration (NOAA) to monitor Web content streams and continuously improve the toolkit's knowledge repository.

In Year 2, our work concentrated on modularizing the core platform and providing a REST API (Application Programming Interface) to upload, annotate, retrieve and visualize structured and unstructured data. The capability to integrate and visualize third-party data will enable joint exploitation together with the other ASAP industry partners, and increase the dissemination potential of international collaborations.

In Year 3, we (i) completed the system and interface integration to prepare the public release of the ASAP dashboard (reported in D6.5), 45 (ii) extended our collaboration with the United Nations Environment Programme (UNEP), testing new methods developed within ASAP as part of the UNEP Live Platform, 46 and (iii) launched the US Election 2016 Web Monitor as an independent initiative. 47 Such showcases demonstrate technological leadership and increase international visibility, and in turn help attract additional clients who license the developed technologies. The media interest in these showcases has been used not only to promote innovative information services, but also to point towards the ASAP project as a driver of innovation behind a number of enabling technologies.

Looking ahead, the release of a publicly available version of the ASAP Dashboard will convey the significant technical progress achieved, supporting both individual and joint exploitation activities:

- *Individual Exploitation* will be pursued both in terms of using the new API framework to attract customers who require semantic and visual search services for their in-house data assets, as well as extending existing platforms such as the U.S. Climate Resilience Toolkit. Its comprehensive portfolio of interactive visualization services is a core value proposition of webLyzard. The availability of a REST API that serves

as an interface to the repository as well as access to these visualization services will unlock new exploitation potential, following a Visualization-as-a-Service (VaaS) approach, or even a Container-as-a-Service (CaaS) approach when deployed in conjunction with the Docker platform.

- *Joint Exploitation.* The integration and visualization of structured and structured content, as demonstrated through the joint processing of telecommunications data (WIND) and Web content metrics (webLyzard, IMR), also paves the way for joint exploitation, for example in the form of business intelligence services that relate actual human behavior (e.g., phone calls and SMS messages during a public event) with peaks of online coverage and the aggregated perceptions of online communities. Both use cases present opportunities to develop specific products after the successful completion of the ASAP project, including Web intelligence offers for (i) telecommunications and tourism companies, initially targeting the Italian and Austrian markets (WIND, webLyzard; WP9), and (ii) large B2C brands operating internationally and using e-commerce marketplaces as distribution channels for their products (IMR, webLyzard; WP8).

Promotional Material

Business Cards

webLyzard organized the printing of 4,000 business cards using the corporate identity established through the WordPress theme of the ASAP Website. The cards are available to all project partners to increase project visibility at conferences and workshop. They serve as a cost effective, environmentally sustainable and often more accepted alternative to regular printing material.

T-Shirts

FORTH organized the printing of t-shirts with the ASAP logo as an additional promotional item to increase project visibility, for example when attending conferences or showcasing

the project at various events. T-Shirts were also distributed in attending graduate students in the HiPEAC computing systems week 2014 and at the FORTH BigData summer school 2017.

Video

Project ASAP has created and distributed a set of tutorial videos regarding individual tools and the platform as a whole.

- `https://www.youtube.com/channel/UCndqDS--1SZCWVl3yj253Fw`

1.5 Contact Details

Foundation for Research and Technology – Hellas
Coordinator
Contact person: Dr. Polyvios Pratikakis
E-mail: polyvios@ics.forth.gr

Institute of Communication and Computer Systems
Contact persons: Prof. Dimitrios Tsoumakos, Dr. Katerina Doka
E-mail: dtsouma,doka@cslab.ece.ntua.gr

Université de Genève
Contact person: Dr. Verena Kantere
E-mail: Verena.Kantere@unige.ch

Queen's University Belfast
Contact persons: Prof. Hans Vandierendonck, Prof. Dimitrios Nikolopoulos
E-mail: h.vandierendonck,d.nikolopoulos@qub.ac.uk

Internet Memory Research
Contact person: Prof. Philippe Rigaux
E-mail: philippe.rigaux@internetmemory.net

Wind Telecommunicazioni
Contact persons: Roberto Bertoldi, Maria Rita Spada
E-mail: roberto.bertoldi,MariaRita.Spada@wind.it

webLyzard technology
Contact persons: Arno Scharl
E-mail: scharl@weblyzard.com