CHOReOS: Large Scale Choreographies for the Future Internet

Marco Autili*, Paola Inverardi* and Massimo Tivoli*

*Department of Information Engineering Computer Science and Mathematics

University of L'Aquila, Italy

Email: {marco.autili, paola.inverardi, massimo.tivoli}@univaq.it

Abstract—In this paper we share our experience in the CHOReOS EU project. CHOReOS provides solutions for the development and execution of large scale choreographies for the Future Internet. Our main involvement in the project concerns the definition of a choreography development process based on automated synthesis of choreographies out of a large scale service base and a user-centric requirements specification. By focusing on the work package WP2, whose main outcome is the realization of the CHOReOS development process, we discuss the WP2 activities by also summarizing main objectives and related achievements.

I. INTRODUCTION

The Future Internet[1] (FI) constitutes a yet-to-come Internet, whose Ultra-Large Scale (ULS) changes everything. As for any new domain, the understanding of the FI has been receiving significant attention since, there is not yet a well established characterization of it. However, the key characteristics of the FI among which ULS, heterogeneity, mobility and the consequent need for evolvability, are rarely considered in conjunction. This has specifically been the focus of the CHOReOS EU project[2], which has investigated the impact of the FI's ULS on Service Oriented Systems (SOS) realized as choreographies of services. That is, the FI for SOS development envisions a ubiquitous world of available services that collaborate to fit users' needs [1]. The trend is to build applications by reusing and assembling existing services rather than realizing stand-alone and monolithic programs [2], [3].

The Service Engineering (SE) approach adopted by CHOReOS is that of composing distributed services by considering a global specification, called *Choreography*, of the interactions between the participant services. Choreography formalizes the way participants coordinate their interactions. The focus is not on orchestrations of the work performed by the participants, but rather on the exchange of messages between them. In this respect, a choreography defines the global expected behavior between interacting participants.

CHOReOS offers a development process and a supporting Integrated Development and Runtime Environment (IDRE) that provides a comprehensive platform for the design, the deployment, the execution and the evolution of ULS service choreographies. The CHOReOS IDRE provides sophisticated model-driven techniques and tools that allow domain experts to specify requirements and to model choreographies. After user requirements have been captured via dedicated tools, the choreography is automatically synthesized into distributed coordinating logic. Choreographies are synthesized out of ULS repositories of services. Analysis mechanisms ensure scalability and quality of the synthesized choreographies. As a result, the IDRE delivers ULS choreographies of services to be deployed on the CHOReOS cloud/grid middleware. Three main business cases have been exploited to demonstrate the benefits of the CHOReOS IDRE: *Passenger-Friendly Airport*, taken from the context of air transportation, which improves services delivered in the airport as well as Air Traffic Flow Management; *Adaptive Customer Relationship Booster*, which is a sophisticated application that allows customers to receive personalized discounts and promotions on the fly and in a context-aware manner; *DynaRoute*, which provides a dynamic personal organizer that leverages services based on business infrastructure and smart phone sensor data which include location, personal calendar, and user preferences.

CHOReOS is founded by the European Community (Seventh Framework Programme - FP7) and follows a Collaborative Project funding scheme. Counting a consortium of 17 partners, the project has a duration of 36 months, with contractual starting date October 1st, 2010. The following are the participants to the project consortium: Thales, INRIA, Linagora (France), MLS, Univ. of Ioannina, Virtual Trip (Greece), ISTI-CNR, Univ. of L'Aquila, Univ. of Camerino, WIND, Cefriel, Consel (Italy), No Magic Europe (Lithuania), OW2 (Europe), City University London (UK), Vidzeme University (Latvija), and Univ. of Sao Paulo (Brasil). The total amount of funding is 8.665.785 €. According to the core FP7 objectives, the most relevant topics of the project are Internet of Services, Software and Virtualisation. Within the wide spectrum of the CHOReOS project, the University of L'Aquila (UDA) is leading partner of the Work Package WP2, and is leader of the WP2 Task concerning the synthesis.

The paper is structured as follows. Section II provides an overview of the project structure. Section III focuses on the choreography development process realized by WP2 and Section IV on the synthesis process realized by one of the WP2 Tasks. Conclusions are given in Section V.

---

[1]http://www.future-internet.eu/

[2]http://http://www.choreos.eu/

II. CHOReOS AT A GLANCE

CHOReOS is organized along 4 Research and Technological Development (RTD) work packages (WP1 to WP4), each focused on the key Scientific & Technical (S&T) aspects of the CHOReOS development process and associated IDRE.

WP1 focuses on the study of the architectural style that stems from the development of choreography-based adaptable, QoS-aware systems in the ULS FI. WP1 then analyzes and formalizes the various features of the FI and their impact on software development. These serve to elaborate a reference software architecture for the target choreography-based software systems, specifically introducing abstractions for embodied components (services), connectors (interaction protocols supported by underlying middleware) and coordination (representative choreography patterns).

WP2 builds on the preceding architecture, to elicit a dynamic choreography-centric development process and supporting methods and tools for software systems in the FI. The process builds upon well-founded abstractions and is based on Model-Driven Engineering (MDE) for rigorous and systematized development, which is key to overcome the ULS of the target environment. In effect, proposed innovations allow for requirements specifications and choreography modelling by domain experts (as opposed to IT professionals) and further synthesis of adaptable, QoS-aware choreographies out of the ULS service base together with scalability assessment.

WP3 complements the dynamic development environment with the necessary middleware support, effectively enabling the deployment of adaptable, QoS-aware choreographies in the ULS FI, accounting more specifically for ULS in the number of users and services, and related high heterogeneity of the networking environment spanning from the fixed Internet to the Internet of Things. Innovations come from integration and further evolution of latest research advances in the area of Grid and cloud computing, Enterprise Service Bus (ESB) and pervasive computing to meet the challenges posed by the ULS FI on service oriented middleware technology.

WP4 works out the concepts of Governance of services and choreographies based on thorough mechanisms for Verification and Validation (V&V) of adaptable, QoS-aware choreographies in the context of the ULS FI. Innovations come from the integration of well-proven approaches to V&V while making them shifting from a mostly design-time to a run-time activity.

Next to the above, a specific work package, WP5, has been introduced in order to cope with the integration of results of WP1 to WP4 into the CHOReOS IDRE. WP5 thus provides a clean environment for the use cases as well as high quality components to be open sourced.

In addition, DEMO work packages, WP6, WP7 and WP8, deal with the thorough assessment of CHOReOS S&T developments, from both a technological and an exploitation perspective, based on experimentation with the future real-life industry-focused scenarios introduced in Section I, as well as an actual ULS demonstration on vehicular networks.

Further, the following work packages allow for the effective undertaking of the CHOReOS research, from consortium management through assessment to spreading results beyond the consortium as well as standardisation: WP11 on project management, WP9 on technology transfer & dissemination, and WP10 on technical and socio-technical assessment and on the exploitation, which is mainly based on the preceding assessment. Hereafter, we focus on the work package WP2.

III. WP2: CHOReOS DEVELOPMENT PROCESS

The definition of the CHOReOS development process has been the main objective of WP2. Towards this goal, within WP2, a set of activities have been identified and some of the components implementing these activities have been developed in other WPs, as summarized by the following items:

- Domain expert requirements specification tool (implemented in WP2);
- ULS abstraction-oriented service base management, registration and query engine (implemented in WP2);
- Choreography synthesis processor and related model transformations (implemented in WP2);
- Choreography deployment and execution, and supporting middleware (implemented in WP3);
- Design and run-time analysis, governance V&V, monitoring and V&V configuration tools (implemented in WP2 and WP4).

UDA coordinates the contributions of 8 partners, with a total effort of 143 Person-Months (PMs), covering all the 36 Months (Ms) of the project duration, with a production of 3 deliverables (D2.1, D2.2, and D2.3), one per year (at M12, M24, and M36).

In the following we report on the history of the work undertaken in CHOReOS by the WP2 partners:

1) *Deliverable D2.1 [4]* – In D2.1 we presented a model of the CHOReOS development process by abstractly describing the "strategy" that CHOReOS uses for specifying, analysing, synthesizing and enacting, governing and monitoring ULS choreographies during the whole life cycle (from design to runtime to evolution). The dynamic development process model consists of activities that are common to other development processes, but within CHOReOS they are organized in order to fulfill the specific commitments that have been imposed to deal with ULS service choreographies. In D2.1 we gave an abstract description that characterizes the CHOReOS development process by defining the main activities that need to be performed and the artifacts manipulated by these activities without referring to specific technologies, tools, standards, models, etc.

2) *Deliverable D2.2 [5]* – In D2.2 [5], by leveraging the abstract process model defined in D2.1, we precisely described the specific set of concrete process activities, and related relationships, by using BPMN2 Process Diagrams as graphical notation. Moreover, in D2.2 we specified what standards, notations, languages, technologies, and tools we have then used to implement the tools supporting the defined process activities, and that have been then integrated into the CHOReOS IDRE.
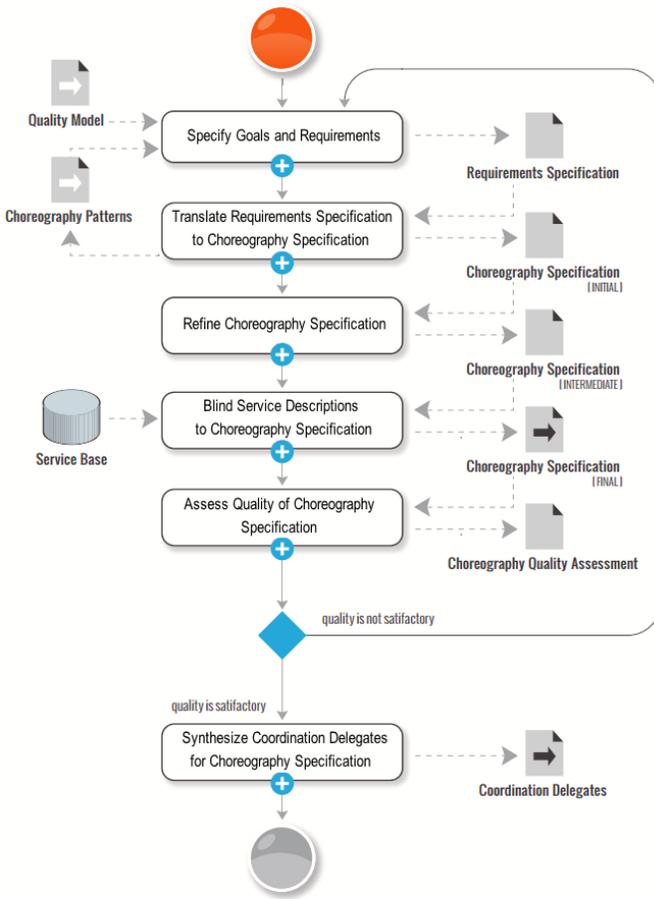
Fig. 1. CHOReOS Development Process (first level view)

3) *Deliverable D2.3 [6]* – In D2.3 [6] we describe in details the software that have been developed in WP2, and refer to the WP3-5 deliverables for the description of the software developed outside WP2[3]. In D2.3, for each tool we leverage the particular use case(s) whose characteristics best demonstrate its specific functions.

As shown in Fig. 1, the development process implemented by the IDRE consists of a set of activities, and related artifacts, that allow: (i) to specify requirements by using a dedicated tool; (ii) to derive an initial choreography specification from them, given in the Business Process Modeling Notation version 2[4] (BPMN2); (iii) to refine the initial choreography specification to allow for automation; (iv) to concretize the choreography specification by discovering services suitable to play the roles of the choreography; (v) to assess the quality of the choreography; and (vi) to automatically synthesize a distributed choreographer, to be then used to enact and execute the choreography on top of the CHOReOS middleware.

IV. CHOReOS SYNTHESIS PROCESS

This section summarizes the synthesis approach that has been developed within the dedicated WP2 Task, and highlights how it has been used in CHOReOS for solving the problem of automatic distributed coordination, service protocol adaptation, and choreography evolution. A detailed description of the approach can be found in [3], [6], [7], [8].

The synthesis process uses dedicated model transformations to generate, out of a BPMN2 choreography diagram, an automata-based specification of the coordination logic "implied" by the choreography. Specifically, an extension of Labelled Transition Systems (LTSs), called *Choreography LTS* (CLTS), is generated to explicitly describe the coordination logic that must be applied to enforce the choreography. CLTSs represent the mean to precisely describe the complex coordination logics implied by BPMN2 choreography specifications. For the choreography to be externally enforced, the coordination logic modeled by the CLTS is distributed between additional software entities, whose goal is to coordinate (from outside) the interaction of the participant services in a way that the resulting collaboration realizes the specified choreography. To this aim, our method automatically derives these software entities, called *Coordination Delegates* (*CD*s), and interpose them among the participant services according to the CHOReOS architectural style (see Fig. 2). CDs perform pure coordination of the services' interaction (i.e., *standard communication* in the figure) in a way that the resulting collaboration realizes the specified choreography. To this purpose, the coordination logic is distributed among a set of *Coordination Models* that codify coordination information. Then, at run time (after being deployed on the CHOReOS middleware), the CDs exchange this coordination information (i.e., *additional communication*) to prevent possible undesired interactions. The latter are those interactions that do not belong to the set of interactions allowed by the choreography specification and can happen when the services collaborate in an uncontrolled way.
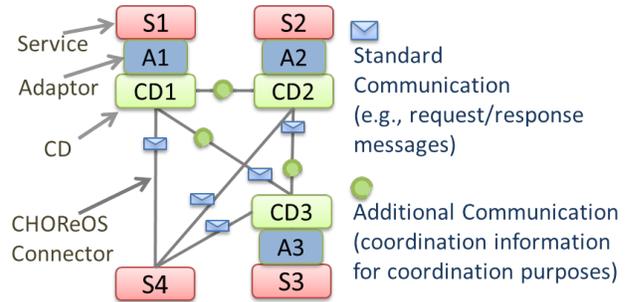

Fig. 2. CHOReOS Architectural Style

However, the coordination logic performed by the CDs is service-independent since it is based on the expected behavior of the participants as specified by the choreography, rather than on the actual one obtained after discovery. Within CHOReOS,

this is done to consistently realize separation of concerns, i.e., to separate pure coordination issues (i.e., undesired interactions) from adaptation/mediation ones (e.g., syntactic mismatches, data incompatibilities at the service interface level). The latter can arise whenever a service discovered as a participant does not exactly match the role to be played. To tackle the adaptation/mediation problem, a modular adaptor (e.g., A1 in Fig. 2) is automatically synthesized. Its modularity is achieved by synthesizing it as a composition of *mediators*.

Adaptors are synthesized by performing a 2-phase method. In this paper we do not go into the details of the two phases that are rigorously described in [8], we rather give an overview of them. The first phase takes as input a domain ontology DO, for services (indeed, for service behavioural descriptions) P and R, and automatically synthesizes a set, W, of *Communication Mediators* (CMs). The goal of CMs is to solve *communication mismatches*. They concern the semantics and granularity of the service protocol actions. To solve these kind of mismatches it is necessary to assume and use ontology knowledge in order to align the two protocols to the same concepts and language. In particular, the CMs in W are used as wrappers for P and R so to "align" their different alphabets to the same alphabet. In other words, the goal of CMs is to make two heterogeneous service protocols "speak" the same language. To this aim, the synthesized CMs translate an action from an alphabet into a certain sequence of actions from another alphabet (e.g., through the merge of messages). However, despite the achieved alphabet alignment, *coordination mismatches* are still possible (e.g., some message reordering is needed); the second phase is for synthesizing *COordination Mediators* (COMs) whose goal is to solve such mismatches. Coordination mismatches concern the control structure of the protocols and can be solved by means of a COM that mediates the conversation between the two protocols so that they can actually interact. The synthesis of COMs is carried out by reasoning on the traces of the "wrapped" P and R. As detailed in [8], for all pairs of traces, if possible, a COM that makes the two traces interoperable is synthesized. The parallel composition of the synthesized COMs represents, under alphabet alignment, the correct adaptor for P and R.

V. Conclusions

CHOReOS is related to others international projects and research works described in the literature. For space reasons, it is not possible to report here a discussion about related works and projects. We refer to the work described in [9] and in Deliverable D1.1 [10] for a comparison with related works and projects in the state of the art.

The main goal of CHOReOS was to support the development of choreography-based service-oriented systems. To this aim, within CHOReOS we precisely defined a development process and implemented a supporting Integrated Development and Runtime Environment, called IDRE. In particular, focussing on WP2, as main achievement, we tackled the problem of realizability enforcement, which so far has been receiving little attention by the SE community. Furthermore,

the definition of the CHOReOS process and its synthesis sub-process required the exploitation of state-of-the-art languages, systems, and techniques that have emerged in different contexts including SOA, model-transformations, and distributed coordination. Their integration and interoperability within the same technical space gave us the opportunity to harness the power and individual capabilities of different tools as part of a tool chain to support the systematic development of choreography-based systems which has thus far been missed.

In the following we summarize some of the Core Objectives (CO) of WP2 and their Achievements (A).

- **CO1**: From requirements to service choreography. **A1**: Requirements specification tool for domain experts, accounting for both functional, and non, properties.
- **CO2**: Scaling up to ultra-large service bases. **A2**: Scalable hierarchically structured registry based on service (functional and non-functional) abstractions.
- **CO3**: Enable service discovery in the FI, based on the concept of service abstractions. **A3**: Implementation of different clustering algorithms for the extraction of service abstractions.
- **CO4**: Enforcing scalable choreographies. **A4**: Implementation of an automatic synthesis processor of evolvable decentralized choreographies and related generation of coordination delegates.
- **CO5**: Handling of BPMN2 Choreography Diagrams complex behaviors. **A5**: M2M transformation able to transform most of the Choreography Diagram constructs.
- **CO6**: Enabling adaptable and evolvable choreographies in the ULS Future Internet. **A6**: Decentralized synthesis of choreographies, based on distributed coordination and modular adaptation of services.

References

[1] ERCIM News, "Special Theme: Future Internet Technology. Number 77," April 2009.

[2] P. Poizat and G. Salaün, "Checking the Realizability of BPMN 2.0 Choreographies," in *Proceedings of SAC 2012*, 2012, pp. 1927–1934.

[3] M. Autili, D. Di Ruscio, A. Di Salle, P. Inverardi, and M. Tivoli, "A model-based synthesis process for choreography realizability enforcement," in *FASE'13*, ser. LNCS, 2013.

[4] C. P. Team, "CHOReOS dynamic development model definition - Public Project deliverable D2.1," September 2011.

[5] CHOReOS Project Team, "Definition of the Dynamic Development Process for Adaptable QoS-aware ULS Choreographies," 2012.

[6] CHOReOS Project Team, "CHOReOS Dynamic Development Process: methods and tools," 2013.

[7] M. Autili, A. Salle, and M. Tivoli, "Synthesis of resilient choreographies," in *Software Engineering for Resilient Systems*, ser. LNCS 8166, 2013.

[8] P. Inverardi and M. Tivoli, "Automatic synthesis of modular connectors via composition of protocol mediation patterns," in *Proc. of ICSE'13*.

[9] V. Issarny, N. Georgantas, S. Hachem, A. Zarras, P. Vassiliadist, M. Autili, M. A. Gerosa, and A. B. Hamida, "Service-oriented middleware for the future internet: state of the art and research directions," *Journal of Internet Services and Applications*, vol. 2, no. 1, pp. 23–45, 2011.

[10] CHOReOS Project Team, "CHOReOS State of the Art, Baseline, and Beyond," December 2010.