A Systematic Survey report on Various Frameworks and Models for Verification of Choreography in SOA

Choudhury Bandana Das
School of Computer Engineering
KIIT University, Bhubaneswar, India, 751015
bandanadas.11@gmail.com

Prachet Bhuyan
School of Computer Engineering
KIIT University, Bhubaneswar, India, 751024
pbhuyanfcs@gmail.com

ABSTRACT: **Service- Oriented Architecture (SOA) is a mainstream strategy for improving IT performance. Web Services are self-contained internet enabled applications which are capable of performing business activities. Web Services are software services which are advertised by providers and invoked by customers using web service composition which is composed of orchestration and choreography. Orchestration refers to an executable business process that can interact with both internal and external web services. Choreography is the process that tracks the message sequences among multiple parties and sources. To carry out workflows and business processes based on web services: Business Process Execution Language (BPEL) follows the orchestration paradigm and Web Service Description Language (WS-CDL) follows choreography scenario.In this paper we present a complete survey report on choreography based frameworks and models to show message interaction in an organization and also focused on verification of choreography in SOA. After reviewing we summarizes by identifying some loopholes which causes loss of messages while being tracked by choreography.**

KeyWords- SOA, Web Service, Choreography, Orchestration, BPEL, WS-CDL, Choreography Frameworks, Choreography Models and Verification of Choreography.

I. INTRODUCTION

In the past decades, the evolution of software systems observes continuous growth of the complexity levels, in order to resolve this issue, Service-Oriented Architecture (SOA) is an architectural approach which supports service composition to ensure reusability and productivity. SOA is an architectural approach which supports service composition to ensure reusability and productivity. SOA is an approach for organizing by using Information Technology to match and combine needs with capabilities in order to support the overall mission of an enterprise. Currently, computer science is in a new version or period of abstraction. Generation ago, we had learned to abstract from hardware but now it is totally changed, we have learned to abstract from software in terms of SOA. SOA has become quickly the leading software paradigm. SOA has been widely adopted through the web services approach. So, SOA together with web service technology, concentrates on the definition of a software system as a complex distributed business application. Today, more and more organizations focus towards web based services in order to have a smooth business process. Services are communicated through network and messages are transmitted through interfaces in a platform neutral, standardized format like Extensible Mark up Language (XML). Web services are self- constrained and perform business activities. The specification, design and implementation of web service requires basic three major aspects such as orchestration, conversation and choreography. Web service orchestration and choreography are both concerned with the web service composition in order to meet the needs of business processes. BPEL is for orchestration and WS-CDL is for choreography. Other standards such as WSCI mainly supports choreography. Orchestration defines the internal behaviour of a business process (Process Model). Choreography addresses interactions that implement the collaboration between services. Choreography focuses on the external perspective that is process interaction (Interaction Model). Choreography has altered much interact in the research field and researchers have worked on its model, analysis and implementation issues. In order to create business processes for an organization, effective message passing between clients is necessary. To strengthen the business processes, security is the major concerned for every organization's authentication and authorization. In today's business environment "Choreography" is an established concept but is less well researched. During the design process of choreography, decentralized control brings a set of challenges and issues which causes problem in tracking messages effectively. To overcome this issue a utility tool can be implemented to support verification for choreography in order to filter out the messages which are highly secured and health of the messages. This paper focused on complete systematic survey on various approaches for choreography and different methods for verification of choreography in SOA. Next, we have identified different loopholes or lacunas which causes barrier in effective message passing scenario in choreography.

The rest of the paper is arranged as follows: Section II describes the related work. Section III we have presented the comparison table for various choreography frameworks and models with different verification methods for choreography in SOA. Section IV describes our objective based on our detailed literature survey on choreography in SOA. Section V presents the conclusion and future work.

II. RELATED WORK

Peltz[1] introduces that web service composition mainly depends on two aspects i.e. orchestration and choreography. The author describes orchestration as an executable business process which can interact with both internal and external web services. Choreography is defined as tracking message sequences among multiple parties and sources. The author focused on the work to design a business process by composing web services including web service languages and standards. Standards such as XLANG focus on the creation of business process and message exchange behaviour among web services, Web Service Flow Language (WSFL) describes both public and private process flows and then WSFL exposes Web Service Description Language (WSDL) interface which allows recursive composition by supporting exception handling. The author introduces Business Process Execution Language 4 Web Services (BPEL4WS) specification which is used for modelling behaviour of web services for short models, it is also called BPEL which provides support for both executable and abstract business processes. The author also gives idea about Web Service Choreography Interface (WSCI) which describes only the observable behaviour of participants in a specific business interaction. The author depicts a clear picture of orchestration and choreography which is described in the Table 1.

O**RCHESTRATION**	CHOREOGRAPHY
Refers to an executable business process.	Create business processes from composite web services.
Include business logic and task execution order.	Involve parties to describe it's part in the interaction.
Represent control from one party's perspective.	Tracks the message sequences among multiple parties.
Less collaborative.	More collaborative.

Table 1 Depicts the difference between Orchestration and Choreography.

Foster et al. [2] discusses verification and implementation of web service compositions by proposing a model- based approach. The approach adds semantics to the specification models in order to carry out verification approach. The expected verification result is given by designer and implementer through UML in the form of Message Sequence Charts (MSCs) and then compilation is done by Finite State Process notation (FSP). FSP model transition of workflow processes through a modelling tool such as Labelled Transition System Analyser (LTSA), which provides compilation of FSP. Implementations provide trace equivalence for verification process.

Uchitel et al. [3] have discussed a model- based approach for verification of web service interactions . The author provides a precise definition of compatibility and extend an approach for modelling and analysing the behaviour of web service compositions. The aim of this analysis focuses on the compatibility of processes within the composition environment of web services specification. The compatibility of composition are defined as interface, behaviour and input- output data. The approach concentrates on Finite State Machine representations of web service orchestrations and add semantics to the existing process interactions.

Brogi et al.[4] presented the formalization based on WSCI web service choreographies by using Process Algebra approach (CCS) and discussed the benefits gained from such formalization. The aim is to check that two or more web services are compatible to interoperate or not, and if not, then aims to check whether the specification of adaptors are automatically generated or not with the help of Promela tool.

Magee et al. [5] describes model- based approach for verification of web service compositions and implementations with the help of supporting tool. The proposed approach consists of a designer, providing necessary requirements, uses MSCs which define the usage and modelling of services. Then the scenarios are composed and synthesized to get a behavioural model, by the use of Finite State Process(FSP) algebra and then implemented to the tool called Labelled Transition System (LTS).

Zhu et al.[6] proposed an ontology based framework for web service processes which provides techniques for web service composition, description and matching. They also discussed a description logic knowledge representation and reasoning framework which provides foundations and they based this ontological framework on an operational model for behaviour of service process and composition.

Gomez-Gutierrez et al.[7] introduces web service composition and highlights that these compositions requires advanced programming skills and knowledge about certain technologies. The authors proposed a framework for the smooth composition of web services that focus on government services requirements for effective interaction. The goal is to give permission to government agencies to smoothly register simple services or create new services based on already earlier registered services. The authors defines the additional goal as to reduce and minimize overhead by using mapping methods. The authors presented in details of a ongoing project that supports the proposed framework which overview the concept of easy registration of services. This framework could be useful for other entities for sharing choreographed services along with government agencies.

Zongyan et al. [8] proposed algorithm and a formal model of WS-CDL to get initial channel sets. The algorithm is used for implementation and verification of choreographies with channel passing. The authors proposed an approach to detect design defects such as logic mistakes in choreographies. The authors also discussed choreography model which discards properties like exception handling, choreography performing, multiple participant instances, information driven control flow, service channels and session channel.

Jiang et al. [9] describes behaviour of choreography more elaborately than orchestration. The proposed approach called Semantic Interface for Mobile Services(SIMS) concentrates on validation of web service choreographies. The validation is performed for interacting participants and for individual participants. Conformance checking is performed for individual participants against the choreography specification. In details, SIMS provides methods for validation which assures of usefulness of composition.

Madani et al.[10] discussed about web service compositions, mechanisms, formal methods and different verification issues. The main aim is to check the correctness of choreography behaviour. The authors presented a method using first order logic notation based on partial-order planning problems. This method is introduced to model the variable specification of the system and also check the properties of the model such as reachability of ideal and safe states.

Kozyura et al.[11] presented the relations between local enforceability and inconsumable messages with respect to local model is obtained from the global model. The authors addresses the formal verification of message choreography models with an example of MCM approach, developed at SAP Research. The main aim is to address that whenever a choreography model has no inconsumable messages, it is local enforceable, no matter how the behaviour of global model is connected with the behaviour of local partner model.

Foster et al. [12] discussed behaviours of orchestration and choreography and presented model based on service orchestration and choreography. The authors presented properties which analyse and check service models for orchestration and choreography. Lastly, the authors also describes mechanical support based on integrated workbench to address verification and validation of web service compositions.

Yoon et al.[13] proposed a novel distributed service choreography framework to overcome the issues and challenges related to loosely coupled interactions and semantic conflicts. The authors discussed deployment of safety constraints to decrease overhead and increase conflicting free, more reliable service interactions. The process is achieved through federated publish/subscribe messaging format.

III. COMPARISION

TABLE

SL No	Paper Title	Aim of the paper and Proposed Framework or Model	Web Services Interaction, Languages Used, Tools used for Implementation.	Results	Remarks
1	Web Service Orchestration and Choreography[1]	Present overview on various standards for business process to model interaction. "A Conversational Model" is used for BPEL4WS process flow and WSCI collaboration.	Asynchronous interaction. BPEL, WSDL, BPEL4WS specification and WSCI interface is used. Implementation is not done.	Defines Orchestration as an executable business process and Choreography creates business process from composite web services. Provides Conventional Model which is flexible and dynamic .Disadvantage- standards used for orchestration and choreography do not support for security.	This paper mainly found the difference between orchestration and choreography, propose BPEL4WS process flows for structured and basic activities . WSCI shows

					observable behaviour between web services.
2	Model-based Verification of Web Service Compositions [2]	To verify web service composition for web service implementation by proposing an approach called " Model- Based Approach". This approach supports verification against specification models.	BPEL4WS is used to execute workflow specification for web service invocation. Specification of the designs modelled in UML in the form of MSCs, compiled to FSP in LTSA tool to describe concurrent programs.	Provides Verification Approach to verify service workflows and any workflow processes . Assigns semantics to confirm expected results. Disdavantage lacks a proper framework which can simplify modelling, verification and implementation.	This paper mainly describe a early verification through model checking process to check violations and deadlocks to get trace results of FSP model checking.
3	Compatibility Verification for Web Service Choreography[3]	To verify process interactions for coordinated web service compositions by "Model- Based Approach". This approach discusses compatibility analysis of web service compositions.	Synchronous message interaction. BPEL4WS used to provide interaction activities of web service compositions. FSP is used in LTSA tool for implementation.	Provides compatibility as interface, behaviour, input-output data. Provides Combined Disadvantage - does not concentrate on private models, absence of full verification of choreography at implementation and coordination layers.	This paper mainly focus on Safety and Progress checks by gathering all orchestrations, model activities based on BPEL4WS, translate it to FSP and finally compile processes in LTSA tool.
4	Formalizing Web Service Choreographies [4]	To check two or more web services are compatible to interoperate or not. Present Formalization based on "WSCI Global Model" to check compatibility and specification of adaptors on web service compositions.	WSCI, BPEL4WS, WSFL and WSCDL notations are used. Algorithm for building adaptor is used to eliminate deadlock in PROMELA tool.	Shows how WSCI formalization uses process algebra approach to check compatibility, interoperability, specification of adaptors and deadlock free messages. Disadvantage-system fails unreportedly, checking of correlations, transactions and properties are omitted.	This paper mainly focus on WSCI formalization using (CCS) to reduce complexity of web service choreographies. Algorithm is used to build adaptors to check deadlock, safety and liveness properties.
5	Tool Support for Model-based Engineering of Web Service Composition [5]	Describes tool support to verify web service composition and implementation. " Behavioural Models" describes modelling of service interactions.	Synchronous message interaction. BPEL4WS is used, scenarios modelled in UML in form of MSCs, compiled to FSP algebra for modelling behaviour and verification is implemented in Eclipse environment.	Provides approach to generate first behavioural model to check scenario acceptable to end-user. BPEL4WS implementation generate 2nd behavioural model to check deadlock and liveness properties. Disadvantage- Absence of clarification on notational and semantic correctness. Absence of service	This paper mainly focus on giving early feedbacks to developers to gain greater assurance on both design and implementation. verification, check compatibility and policy. Validation, assure

				workflow for all service actors.	requirements for designer and implementer.
6	A Semantical Framework for the Orchestration and Choreography of Web Services[6]	To support (service-based) software development on and for the web. "Formal Operational Model"- formalize orchestration and choreography to develop semantical framework that supports composition, description and matching.	Synchronous communication.WS-BPEL used for orchestration and WS-CDL used for choreographies. Ontology used represent knowledge about services.	Provides WSF platform based on discovery and invocation. Provides Service process composition as orchestration(process model) to check sequence, choice, iteration, control flow, execution order and choreography (interaction model) to check request/response action, loop, concurrency, channels and data flow. Disadvantage- absence of predictable assembly, maintainance change and evolution mangement.	This paper mainly focus on service reuse. Ontological framework work as a development approach for distributed platform as web based on Description logic.
7	A Framework for Smooth Composition of Choreographies of Web Services[7]	Introduces web service composition and highlights that these compositions requires advanced programming skills and knowledge on web service composition. "Smooth Composition Framework" allow registration of services by government agencies.	WSDL file is used to access the service. SOAP/ XML is used for message exchange interactions. Implementation is not done.	Framework provides web-based infrastructure, accessed by govt agencies to register services from new or already registered ones. Main Components of Framework-gateway, store information by WSDL file about entity and supports composite services. Disadvantage- absence of sharing of services, absence of issuing choreography to check deadlock and design flaws.	This paper mainly focus on registration of composite services. The composition of services- specify usage and ordering of services at specific condition.
8	An Approach to Check Choreography with Channel Passing in WS-CDL[8]	Proposed algorithm and a formal model of WS-CDL to get initial channel sets."Formal Model of WS-CDL"- interaction of web services through proper channels.	Interaction through channels, static, sequential, parallel and choice communication structure. WS-BPEL with pie calculus- model channel as location and WS-CDL - invoke channel variable, types not instances. Chor(c)-subset of WS-CDL. Java Implementation of algorithm on Pi4SOA tool.	Provide approach to logic mistakes ,cyclic dependency and duplicate declaration. Algorithm derive initial set of channels. Verification is to check channel absence and channel redundancy. Disadvantage- Exception handling not presented, finalization of choreographies not presented, multiple participants instances absent, service and session channels absent.	This paper mainly concentrates on developing a Formal Model of WS-CDL and propose algorithm to create initial channel sets. Verify choreographies along with channel passing mechanism in Pi4SOA tool.

9	Modelling and Validating Service Choreography with Semantic Interface and Goals[9]	Describes behaviour of choreography more elaborately than Orchestration. Applied "Semantic Interface for Mobile Services" (SIMS)-concentrates on validation of web service choreographies.	Interfaces based on semantic platform and semantic dependencies with stating proper goals and goal states. Implementation - UML 2.0 Activity diagram.	SIMS check well-formedness and design flaws. Validation check consistency of services, two interfaces, and behavioural description of semantic interface. Disadvantage-compositions are safe but are not useful, desired behaviour not expressed, interface dependencies not sufficient and fails to check whether participants acquire desired behaviour.	This paper mainly gives idea on elementary and composite collaborations-specify choreography structure and specify relationship between exactly two participants in semantic interface.
10.	A Logical Formal Model for Verification of Web Service Choreography [10]	Discussed web service composition mechanisms, related formal methods and verification issues. "Choreography Model"- captures interaction at global perspective. " Rule-based Model"- map choreography to logical formal model.	Synchronous interaction. WS-CDL- provide means for tools to validate conformance to choreography descriptions. Implementation-predicates written in PROLOG tool and verification of choreography is done.	Present a method using first- order logic notation to check properties - ideal, reachable and safe states. Choreography model- treat services equally, allow control and data flow dependencies, message correlation and check time constraints. Rule-based model- map choreography with rules, actions, relations, goal state to check safeness and liveness. Disadvantage-fails to check channel absence, time constraints	This paper mainly represent how to transfer choreography structure to new model based on partial order planning and predicates using PROLOG tool.
11.	Local Enforceability and Inconsumable Messages in Choreography Models[11]	To study the relations between local enforceability and inconsumable messages ."Message Choreography Model(MCM)"- describe message exchange between global and local partner models.	MCM language is used for interaction. Implementation- LTS tool developed at SAP Research.	Provide MCM to check local enforceability, check absence of inconsumable messages between global and local models. Implementation addresses formal verification of MCM. Disadvantage- fails to check deadlock, fails to construct a simulation relation between the traces of global and local model. Fails to express exponential time.	This paper mainly focus on getting choreography model free from inconsumable messages that is to check whether messages are changed while being transferred through LTS tool at SAP Research.
12.	An Integrated Workbench for Model-Based Engineering of Service Compositions [12]	To model service orchestration and choreography behaviour, check service models and describe mechanical support for web service compositions. " Formal Behavioural Model"- model checking techniques.	WS-BPEL for orchestration and WS-CDL for choreography. Implementation- Eclipse IDE plug-in[WS-Engineer] is built on top of LTS modelling by using the Finite State Process notation.	Present formal models using process algebra to check safety and correctness properties. In Interaction model MSC design models are synthesized to FSP models to check message sequencing. Algorithms are used to check fault	This paper mainly concentrates on modelling service orchestration and choreography behaviour for checking choreography issues by providing mechanical

		"Interaction Design Model"- represent service composition behaviour.		handling, compensation recovery actions, compatibility, choreography policy and concurrent interactions for two clients. Disadvantage- Arise of deadlock situation and conflicting with server resource usage.	workbench for verification and validation of web service compositions.
13.	A Distributed Framework for Reliable and Efficient Service Choreographies [13]	Resolve semantic conflicts by proposing " Novel Distributed Service Choreography Framework"- forms service bus and flexible message interaction.	Implementation- PADRES- federated content based publish/ subscribe messaging system. Web Service attached to choreography agent- interprets and translates local processes into pub/sub constructs.	Provides framework to reduce extra overheads, pub/sub messages to other partners and deploy conflicting interaction to safety constraints. Algorithms used for arrangement of coordinated messages and mapping between global and local task. Disadvantage- wrong pick decision making by malicious services and latency less flexible.	This paper mainly focus on model which addresses conflicting interactions of web services and propose algorithms to overcome safety constraints.

Table 2 Depicts Comparision Table.

IV. DISCUSSION

This paper mainly concentrates on SOA and Web Services. A Systematic review is made on the web service composition which depends on orchestration and choreography. The detailed literature survey provides insight on SOA rapidly adopting choreography for effective message passing scenario between web services. Different verification methods and tools have been discussed which support choreography in SOA. So, we have strengthen our ideas through survey to identify different lacunas or loopholes which causes barrier in effective message tracking scenario in choreography and presented these loopholes in Table 3. In future we plan to propose a service choreography model and also propose verification algorithm to overcome these issues in order to get healthy choreographed messages.

SL.NO	ISSUES OF CHOREOGRAPHY IN SOA
1.	Deadlock situation and Leak
2.	Problem in crossing of messages.
3.	Channel Passing problem.
4.	Inconsumable messages.
5.	Correctness properties.
6.	Design flaws and less flexibility of Latency.
7.	Rules Governing problem and Global view message exchange problem.

Table 3 depicts issues of choreography in SOA.

V. CONCLUSION AND FUTURE WORK

A systematic review was made on the web service composition which depends on mainly two factors i.e. orchestration and choreography. Our paper mainly focused on detailed literature survey which provides insight on SOA adopting rapidly choreography for effective message passing scenario between web services. Some verification methods such as Machine State Charts(MSCs), UML2.0, Conformance Mapping and Model Checking for choreography were identified. Further, different verification tools such as LTSA, Eclipse environment with plug-in tool as Eclipse Modelling Framework(EMF), FSP, Pi4SOA, Ontology concept have been analysed which support effective message tracking in SOA. Also, we have strengthen our ideas through survey for identifying lacunas or loopholes during message tracking by choreography in SOA. After identification of loopholes or issues in future, we plan to propose a "Service Choreography Model" which will capture the messages in a sequence and queue format by introducing the concept of choreography engine and channel passing mechanism. Then we plan to develop some verification algorithms to sense the issues and overcome the loopholes in order to produce healthy message interaction for choreography in SOA and implement with a supported tool by a case study.

REFERENCES

[1] C. Peltz, "Web Services Orchestration and Choreography". IEEE Computer Society ,2003, pp. 46-52.

[2] H. Foster, S. Uchitel, J. Magee and J. Kramer, "Model-based Verification of Web Service Compositions", in Proceedings of the 18th IEEE International Conference on Automated Software Engineering , 2003.

[3] H. Foster, S. Uchitel, J. Magee and J. Kramer, "Compatibility Verification for Web Service Choreography", in Proceedings of the IEEE International Conference on Web Services, 2004.

[4] A. Brogi, C. Canal, E. Pimentel and A. Vallecillo, "Formalizing Web Service Choreographies", Electronic Notes in Theoretical Computer Science, 2004, pp. 73–94.

[5] H. Foster, S. Uchitel, J. Magee and J. Kramer, "Tool Support for Model- Based Engineering of Web Service Compositions", in Proceedings of the IEEE International Conference on Web Services , 2005.

[6] C. Pahl and Y. Zhu, " A Semantical Framework for the Orchestration and Choreography of Web Services, Electronic Notes in Theoretical Computer Science, 2006, pp. 3–18.

[7] J. A. Gómez-Gutiérrez and P. I. Rivera-Vega, "A Framework for Smooth Composition of Choreographies of Web Services", IEEE International Conference on Web Services ,2007.

[8] C. Chao and Q. Zongyan, "An Approach to Check Choreography with Channel Passing in WS-CDL", IEEE International Conference on Web Services, 2008, pp.700-707.

[9] S. Jiang, J. Floch, and R. Sanders, "Modeling and Validating Service Choreography with Semantic Interfaces and Goals", IEEE International Symposium on Service-Oriented System Engineering, 2008, pp. 73-78.

[10] Z. Madani and N. Nematbakhsh, "A Logical Formal Model for Verification of Web Service Choreography", in Proceedings of 12th International Conference on Computer and Information Technology, December , 2009, pp. 448-453.

[11] V. Kozyura, A. Roth and W. Wei, "Local Enforceability and Inconsumable Messages in Choreography Models", Fourth South-East European Workshop on Formal Methods 2009, pp- 10-16.

[12] H. Foster, S. Uchitel, J. Magee and Jeff Kramer, "An Integrated Workbench for Model-Based Engineering of Service Compositions", IEEE Transactions On Services Computing, VOL. 3, NO. 2, APRIL-JUNE 2010, pp.131-144.

[13] Y. Yoon, C. Ye, H.- A. Jacobsen, "A Distributed Framework for Reliable and Efficient Service Choreographies", International World Wide Web Conference Committee (IW3C2), March 28–April 1, 2011.