Grid Workflow Service Composition Based on Colored Petri Net

Zeng-gang Xiong, Zheng-li Zhai, Xue-min Zhang, Xue-wen Xia

Abstract

This paper presents a model of grid workflow service composition based on colored Petri net that can represent complex session protocols and the component service composition, using token of color of colored Petri net to simulate the different events and message types of business processes. The service session agreement and the grid workflow service composition are converted to colored Petri net, which can make use of the specialized tools of existing Petri net to analyze and verify system performance and behavior properties, thereby enhancing the reliability of the grid workflow service composition.

Keywords: Grid Workflow, Server Composition, Colored Petri Net

1. Introduction

Workflow management systems and workflow modeling and processing techniques have been studied and used for complex business processes [1]. Web service composition problem share many common features with workflow systems. With the putting forward of WSRF, grid service and Web service have gathered together. First, since the grid service is cross-platform, heterogeneous, dynamic, and the practical application are generally very complex, in order to decentralize and simplify the application logic and improve the reusability of services, a single elementary service or component service can not be too complicated, requiring combining many simple grid services to meet the complex applications in practice. Second, different grid services are created based on different heterogeneous systems that may be created in a different way, using different programming languages, and implemented by different suppliers. In order to organize all kinds of loosely coupled, decentralized grid services into a usable system, to achieve the full advantage of grid services and be accepted more widely, there must be combine the existing grid service to create new value-added grid services to meet the application requirements. In the grid service community, the other problems that researchers face are about the lack of proper methodology of design grid services and their relevant composition. Any relevant strategies of the business rules have not yet been got in the service composition. The service composition in grid workflow environment has become an important topic.

There has been several related works that use different approaches and architectures for web service composition [1][2][3][4][5]. But those papers did not give formal description in details on the service composition.

This paper presents a model of grid workflow service composition based on colored Petri net, which provides a new grid service integration to create an application mode of new value-added services, so as to provide effective solution for complex grid workflow application.

The remainder of this paper is organized as follows. Basic compositions of the colored Petri net are presented in Section 2. Section 3 is devoted to An example of grid workflow service composition. Finally, Section 4 provides some concluding remarks.

2. Basic compositions of the colored Petri net

Firstly, the operations (activities) of service are modeled using the colored Petri net. The operation is defined as a colored Petri net with input and output, namely interface information is
described by its input and output token, its internal logic is described by transitions; then the colored Petri nets of several basic composition modes between operations are described.

**Definition 1** (Colored Petri net) Colored Petri net is an eight-tuple $\text{CPN} = (\Sigma, P, T, F, C, G, E, I)$, where:

1. $\Sigma$ is the non-empty finite set of types, also called color sets;
2. $P$ is a finite set of tokens;
3. $T$ is a finite set of transitions;
4. $F \subseteq P \times T \cup T \times P$;  
5. $C$ is the color function, $C: P \rightarrow \Sigma$;
6. $G$ is a guard function, $G: T \rightarrow \text{Boolean expression}$, and meet $\forall t \in T: \text{Type}(G(t)) = \text{Boolean} \land \text{type}(\text{var}(G(t))) \subseteq \Sigma$, where $\text{var}(G(t))$ represents the set of variables contained in function $G(t)$;
7. $E$ is an arc function, $E: F \rightarrow \text{Boolean expression}$, and meet $\forall f \in F: \text{Type}(E(f)) = C(p)_{\text{MS}} \land \text{type}(\text{var}(E(f))) \subseteq \Sigma$, where $p$ is the connected token off;
8. $I$ is the initialization function, $I: P \rightarrow \Sigma$ which generates the initial identification $\text{MS}$ for each color value assigned each token, ie $\forall p \in P: \text{Type}(I(p)) = C(p)_{\text{MS}}$.

The set of types determines data values, operation and function can be used in an expression (such as arc expressions, guard and initialization expressions). We assume that there is at least one element in each type. Tokens, transitions and arcs are three pairwise disjointed finite sets.

Color function $C$ maps a type $C(p)$ for each token $p$. Intuitively, this means that each token on $p$ must have a data value belong to type $C(p)$.

Guard function $G$ maps a Boolean expression for each transition, in which the type of all variables belong to $\Sigma$. If the guard expression is omitted, its value is always true in the drawing.

Arc expression function $E$ maps a $C(p)_{\text{MS}}$ type expression on each arc. This means that the value of each arc expression must be type multiple sets of the token $p$ which connected to this arc.

Initialization function $I$ maps a $C(p)_{\text{MS}}$ type expression for each token $p$. In the drawing, the initialization expression is omitted.

Here is the colored Petri net model of basic activity.

**Definition 2 Basic activity**

A formal definition of basic activity $A$ is a colored Petri net $A=(\Sigma, P, T, F, C, G, E, I)$ (see Figure 1), where:

$\Sigma = \{\text{STRING}\}$;

$P = \{\text{In}, \text{Out}\}$;

$T = \{\text{activity A}\}$;

$F = \{<\text{In}, \text{activity A}>, <\text{activity A}, \text{Out}>\}$;

$C = \{C(\text{In})=\text{STRING}, C(\text{Out})=\text{STRING}\}$

Guard function $G$ is determined based on the semantics of the specific operations, which decides whether the operation can be executed for different input.

$E = \{E(<\text{In}, \text{activity A}>) = x, E(<\text{activity A}, \text{Out}>) = x\}$

$I = \{I(\text{In})=\text{'Go'}, I(\text{Out})=\text{NULL}\}$
Grid Workflow Service Composition Based on Colored Petri Net
Zeng-gang Xiong, Zheng-li Zhai, Xue-min Zhang, Xue-wen Xia

Figure 1. The colored Petri net model of basic activity

The form of a service may be composed of a number of basic activities (operations) in accordance with certain rules, and then the corresponding colored Petri net model is composed of a number of transitions and tokens according to its current relationship.

Transitions fall into two classifications. The activity transition implements specific activities (operations), as the activity A shown in Figure 1. The control transition organizes activities into a service, indicated by the names CTi(i=1,2,...) when drawing. In general, the control transition executes in the end point of its source activity, the next state can be calculated through the occurrence of the corresponding transition. The activity transition executes by calling the specific operation, the next state is decided by the operation results.

The basic activities constitute basic service through composition, and basic services form more complex services through some composition. The following are colored Petri net models come from some basic compositions.

**Definition 3** Sequence composition

For the two activities A1 and A2, and A1=(Σ1,P1, T1, F1, C1, G1, E1, I1), A2=(Σ2, P2, T2, F2, C2, G2, E2, I2), the colored Petri net model shown in Figure 2 A), then A1 and A2 in sequence compose activity A=A1 ⊙ A2=(Σ, P, T, F, C, G, E, I), its colored Petri net model is shown in Figure 2 B).

**Figure 2.** Colored Petri net model by sequence composition
**Definition 4** Concurrent composition

For the two activities $A_1$ and $A_2$, concurrent activities $A_1 \lor A_2$ composed of $A_1$ and $A_2$, its colored Petri net model is shown in Figure 3.

In Figure 3 the role of control transition $CT_1$ is generating the initial conditions of activities $A_1$ and $A_2$ according to the input. The role of the control transition $CT_2$ is aggregating the output of activity $A_1$ and $A_2$, then generating total output, and synchronizing $A_1$ and $A_2$.

**Definition 5** Selection composition

For the two activities $A_1$ and $A_2$, selection activities $A_1 \oplus A_2$ composed of $A_1$ and $A_2$, its colored Petri net model is shown in Figure 4.

![Figure 3. Colored Petri net model by concurrent composition](image1)

![Figure 4. Colored Petri net model by selection composition](image2)

Figure 4 in the role of control transition $CT_1$ generates selection condition based on input, and then by guard expression value of activity transition $A_1$ and $A_2$ to select which one activity should be implemented. The role of control transition $CT_2$ and $CT_3$ respectively is sending the corresponding activities $A_1$, $A_2$'s output to the token $Out$ as the final output of this selection structure.
**Definition 6** Iteration composition

For activity A1 loop in a given condition until the conditions no longer satisfied, the colored Petri net model implementing iteratively this activity (denoted as $^\mu A1$, $^\mu$ is iteration operator) is shown in Figure 5.

In Figure 5, the role of the control transition CT1 is generating the initial conditions and control conditions of iteration based on input and stored these conditions into token p₁ and p₂; the role of the control transition CT2 is to take the current results (originally, the initial conditions) compared with the control conditions and store the results into token p₃. If met the condition (c is true), when meet the guard expression of control transition CT4, control transition CT4 occurs and generates the occurring conditions of activity A1, and then activity A1 can perform. Then the operating results of the activity A1 return to the token p₁through control transition CT5, and then to judge and continue to loop; if not met the conditions, when meet the guard expression of control transition CT3, control transition CT3 occurs and the loop ends.

![Figure 5. Colored Petri net model of iteration structure](image)

For clarity, the above is aimed at only two activities to define four basic compositions: sequence, concurrence, selection, and iteration. By the nesting method can be easily extended to multiple activities composition, and hence can construct any compositions. In short, the compositions of various structures can be expressed with colored Petri net.

3. An example of grid workflow service composition

With the composition of the session protocols and services specified in a unified model, to integrate completely a session to a greater service process becomes possible. Here we use a unified model to create a travel agency service of t workflow service composition in grid environment.

Example 1, many travel agents need to be based on the following business process solutions to create a travel agent service composition to provide a value-added services for travelers, combining air services and credit card services.

Travelers submit a travel order (TripOrder) to the travel agent and hope to get a travel plan (TripPlan). TripOrder includes the place of departure, destination, departure date and time, return date and time (for the round trip), the largest number of interchanges, the number of travelers and other information.

Travel agents find the best travel plans for destination based on traveler’s standards (such as the lowest cost, the least accumulated flight distance, the minimum number of interchanges, etc.). Before putting forward travel plans to the travelers, travel agents call checkTicketAvailability operation of air service to verify the availability of flights. In the case of no flights, travel agents inform travelers and wait for travelers to submit the revised TripOrder.
If the flight had tickets, travel agents sent travel plans to the travelers for confirmation. Travelers decide to book the tickets and tell their contact information to travel agents so that air lines would send them tickets.

Next, travel agents and airline service interactively complete reservation. Air service is assumed to maintain this reservation T time units.

At this time, travelers can order or cancel the reservation. If they decided to accept the travel plan, they will send the order application to the travel agents, including their credit card information. Travel agents call the credit card service to charge. If approved charge, travel agent call bookTicket operation of air services to complete the ticket order. The result is air lines ordered tickets to the proposed travel plan of travel agency and sent the tickets to the travelers. On the other hand, if the credit card charge is rejected, travel agents inform travelers and are waiting for travelers to provide additional payment information. If there is no order or cancel the order application received in T time units, it should cancel the reserved ticket and inform the travel agents. Travel agents send the result message to travelers to confirm.

According to several Petri net models described in Section 2, we can easily obtain the service composition model of Petri net shown in Figure 6.

Figure 6. Colored Petri net model of iteration structure

In the initial design process, to check whether the service process performs as expected, whether the service process and the session of component services are consistent, the designer can design an original sign and run simulation with the help of the CPN Tools. Then: (1) to detect process composition errors using the analysis techniques offering by CPN Tools; (2) correct the error; and repeat steps (1) and (2) until the design specifications are validated. As early as possible to find and correct errors, process inspection can be carried out at the design stage.
4. Conclusions

This paper presents a model of grid workflow service composition based on colored Petri net model. Colored Petri net model of grid workflow service composition can express clearly the composition logic of component service and describe precisely the relationship between services. On the one hand, the user can fully or partially analyze the logical relationship between the component services; on the other hand, users can also utilize reachability graph of colored net, invariant and other tools to simulate the dynamic behavior of service composition and analyze its dynamic nature.

5. Acknowledgement

This research is partially supported by the National Natural Science Foundation of China (No.60873192, No.60873193), Key project in Hubei Provincial Department of Education (D20102701, Q20102705), “Eleventh Five-Year Plan” issues of Hubei Province(2009B105).

6. References