CROSS-ORGANIZATION PROCESS INTEGRATION in DESIGN-BUILD TEAM

Min-Yuan Cheng, Ming-Hsiu Tsai

Abstract—Combining the design-build and fast-tracking delivery methods, this study aims to reengineer the fast-tracking processes across both organizations of designer and constructor, so that fast-tracking processes in the design-build team can span across organizational boundaries and are composed of cooperating workflows executed in different organizations. For this purpose, two subjects are described in this study, namely, (1) fast-tracking model creation, and (2) cross-organizational process reengineering. Consequently, this research addressed a mechanism to facilitate the flexibility of cross-organizational process integration, which may assess alliance of design and construction companies for one design-build project.

Index Terms—design-build, fast-track construction, business process reengineering (BPR), virtual enterprise (VE), intelligent agent (IA).

I. INTRODUCTION

DESIGN-bid-build (DBB) was the project delivery process of choice for most of the twentieth century. By the beginning of 2000, the so-called traditional design-bid build process was still used on nearly two-thirds of the projects in the United States [1], and similar situation arose in Taiwan for last decades. Due to linear structure of DBB, with the production of substantial design required before at-risk construction pricing and construction work begins, design-bid-build tends to be slower than other delivery systems. Moreover, several problems, such as “differing goals of designer and constructor”, “defects of lowest bid contracting”, and “segregation between the work of designer and input of constructors”, bring anxiety while applying DBB method for major construction projects with great complexity. For these reasons, design-build (DB) has been considered an appropriate method of acquisition in Taiwan.

The design-builder is both the A/E and the at-risk constructor, so the owner who employs design-build delivery has only one single point of contact for all questions regarding the design and delivery of the facility. Besides, under a rapidly changing business environment, the fast-tracking delivery method has received considerable attention over the last decade because of its time saving feature depending on overlapping of phases of a project [1]-[3]. Therefore, DB is considered to be the fastest project delivery system as it encourages fast-tracking of design and construction phases. However, unfamiliarity with DB process of both owners and constructors decreased the benefits of DB method in Taiwan. Not only the front-end definition process of defining user needs and translating those needs to a facility program and technical performance requirement is a wrinkle for owners, but also changing adversarial processes to collaborative processes for designers and constructors is critical for the designers and constructors. Furthermore, due to limitation of laws and regulations in Taiwan, A/E and construction companies keep themselves within their own spheres, so that the functional interval between them brings difficulty in cooperation in a DB team. For this reason, to increase the efficiency of cooperation for fast-tracking of DB team, this study aimed to reengineer the business processes across both organizations of designer and constructor, so that processes of team can span across organizational boundaries and are composed of cooperating workflows executed in different organizations. Hence, this research applied the BPR philosophy and extended it to the cross-organizational process to integrate fast-tracking processes of designer and constructor.

Two subjects are described in this study, namely (1) fast-tracking model creation, and (2) design-build cross-organizational process reengineering. Firstly, to create the fast-tracking model, this research applies the axiomatic design (AD) to decompose DB project into design-build modules (DBMs) and to analyze the dependency among them. Based on the AD method, only overlapping relations of coupling, decoupling and uncoupling between design-build modules can been delineated. However, details of process activities for further process integration are necessary. Therefore, this study secondly addressed a cross-organizational process reengineering method to integrate processes of designer and constructor, where the processes are involved in design-build modules. This study evaluates activity similarities between designer and constructor to identify the redundant and irrational activity items, and then creates collaborative fast-tracking process model to perform each DBM, thus the processes of designer and constructor could be executed smoothly as in a virtual enterprise.
II. INTEGRATED DESIGN-BUILD FRAMEWORK

By combining the fast-tracking and the design-build delivery methods, and reengineering processes of design and construction companies leagueing in a DB team, this study aimed to create an Integrated Design-Build Framework (IDBF) to be the foundation for performing the DB project. Two models are included in the IDBF, namely (1) fast-tracking model, and (2) process integration model as shown in Figure 1. The fast-tracking model presents the overlapping relations among all design-build modules (DBMs) which are sub-construction items decomposed from the specific design-build project. Each DBM is assigned a sub goal of the project, which should be achieved by the specified design and construction tasks gathering in the process integration model. The process integration model consists of process and activity sets and is treated as the infrastructure of the IDBF being responsible for performing all DBMs. Moreover, due to separation of designer and constructor in a DB team, an integrated area to coordinate workflows of designer and constructor should be created by the cross-organizational process reengineering method generated in this study.

To build the both models of IDBF, firstly, this study applied the axiomatic design (AD) methodology for mapping a DB project into customer, functional, physical and process domains by zigzagging decomposition and creating the dependency matrices. Figure 2 shows the mapping relationships in AD, in which each domain consists of its specific characteristic vectors such as customer attributes (CA), functional requirements (FR), design parameters (DP) and process variables (PV). Meanwhile, the dependency between two neighbor vectors can be realized through determining the dependency matrices as shown in Figure 3.

Based on the dependency matrices, a concurrent matrix can then be produced to reveal the dependency and overlapping relationships among all DBMs. Hence the fast-tracking model can be a guide for fast-track scheduling of a DB team.

However, because the implementation of each DBM relies on coordination of design and construction processes, details within process are necessary for further process integration. Therefore, this study secondly addressed a cross-organizational process reengineering method to create the Process Integration Model to be the foundation for implementing design-build modules (DBMs). To this aim, two sets, namely process set and activity set, are necessary for creating fast-tracking process model (see Figure 1). The process set includes business processes related to all design-build modules, and each process model owns its corresponding activities summarized in the activity set. With these two sets, this study can then evaluates activity similarities between designer and constructor’s processes to identify the redundant activities, and meanwhile, determines the cooperation linkages among processes by comparing the input/output entities of all activities. Therefore, the process integration model can be built due to elimination of redundant activities and connection of design/build processes.

III. FAST-TRACKING MODEL CREATION

Effective fast-tracking in a construction project can help reduce the duration and lower the cost of the project [3]. Therefore, this study combines the design-build delivery method with fast-tracking approach to enhance efficiency of design-build team united with designer and constructor. However, the time and cost savings are not always realized due to the adhoc approach taken on developing a fast-track plan [6]. To overcome with this defect, this paper focuses on the process aspect to create the IDBF for planning and performing the fast-tracking construction through process integration between designer and constructor.

The fast-tracking model is the preliminary for creating IDBF due to its essentiality for a design-build team and for determining the process integration model. Hence, the purpose of fast-tracking model is addressed to create an idea plan for determining fast-tracking phases and for delineating overlapping relationships of them.

Based on the Axiomatic Design methodology, this study addresses the following steps, shown in Figure 4, to determine the fast-tracking design/build construction phases.
described as below. The Independence Axiom [4]. The details of the procedure are addition, during the mapping process, the design must satisfy decomposition. The FRs and DPs must be decomposed until the design goals. Therefore, decompositions of FRs is required, and of a project, and they are not detailed enough to provide the description of project design goals.

A. Define the highest level FRs

The first step of designing a DB construction project is to determine the customer needs/attributes (CAs) in the customer domain that the building must satisfy. Then the highest level FRs of a DB project in the functional domain can be determined to satisfy the CAs, where the FRs are defined as the minimum set of independent functional requirements which are the description of project design goals.

B. Decompose FRs and Design Parameters (DPs)

The highest level FRs can only express abstract requirements of a project, and they are not detailed enough to provide the design goals. Therefore, decompositions of FRs is required, and Figure 5 shows the decomposing procedure which follows the concept of zigzagging between two domains in the Axiomatic Design, and the repeating constrain for completing the entire decomposition. The FRs and DPs must be decomposed until the design can be implemented without further decomposition. In addition, during the mapping process, the design must satisfy the Independence Axiom [4]. The details of the procedure are described as below.

C. Generate construction process variables (PVs)

Process variables (PVs) are the primary elements in the process domain which characterize the execution process corresponding to DPs. For a design-build project, DPs can be treated as the design solutions, and PVs can be treated as the construction process that can fulfill the design solutions. Unlike the zigzagging decomposition between FRs and DPs in the step 2, just a direct mapping DPs of the physical domain to PVs of the process domain is needed to perform here. Based on the hierarchy of DPs, A PV could be set while designers choose a specific DP and derive its corresponding construction process or operation method.

1. Mapping at the current level FRs to DPs
   Think of all the different ways of fulfilling each of the FRs by identifying plausible DPs.

2. Evaluating the design matrix [A]
   To keep the independence among the FRs, a design matrix [A] must be evaluated after FRs and DPs were generated. The mapping process between the domains can be expressed mathematically in terms of the characteristic vectors that define the design goals and design solutions. At a given level of the design hierarchy, the set of functional requirements that defines the specific design goals constituted the FR vector (denoted as {FR}) in the functional domain. Similarly, the set of design parameters in the physical domain that has been chosen to satisfy the FRs constitutes the DP vector (denoted as {DP}), the relationship between these two vectors can be written as

\[
\{FR\} = \{A\}\{DP\}
\]  (1)

where [A] is called the building design matrix that characterize the project design of a design-build project.

Moreover, to satisfy the Independence Axiom, the design matrix [A] must be either diagonal or triangular matrix. When the design matrix [A] is diagonal, each of the FRs can be satisfied independently by means of one DP. Such a design is called an uncoupled design. When the matrix is triangular, the independence of FRs can be guaranteed if and only if the DPs are determined in a proper sequence. Such a design is called a decoupled design. Any other form of the design matrix is called a full matrix and results in a coupled design which violates the Independence Axiom.

Based on the form of the design matrix [A], the designer can determine whether the FRs satisfy the Independence Axiom or not.

3. Modifying FRs and DPs
   As the design matrix [A] is a full matrix, the DPs or the FRs need to be modified to satisfy the Independence Axiom. Because dependences among the FRs would depend on the selected DPs, replacing the ill-fitting DPs might cause FRs to satisfy the independent axiom.

4. Zigzagging DPs to next level FRs
   As the current level FRs satisfy the Independence Axiom, the designer can go back to the functional domain and decompose the next level FRs.

Fig. 4. Procedure for determining the concurrent design/build construction phases

Fig. 5. The procedure of decomposing FRs and DPs
D. Evaluate the construction matrix \([B]\)

After PVs were completely mapped from the physical domain, the set of process variables that fulfill the design parameters constituted the PV vector (denoted as \(\{PV\}\)) in the process domain. Like (1) shows the relationships between FRs and DPs, the relationship between DPs and PVs can be written as

\[
\{DP\} = \{B\} \{PV\}
\]

where matrix \([B]\) is defined as a construction matrix which presents the relationship between DPs and their corresponding PVs.

E. Produce the concurrent matrix \([C]\)

To make fast-track possible, this study packages design items (DPs) and their corresponding construction processes (PVs) into a design-build module (DBM), so that overlapping relationships among DBMs can be determined. To this aim, the concurrent matrix \([C]\) is applied to summarize the dependence relations of DPs and PVs. As (3) shows, matrix \([C]\) is the product of matrix \([A]\) and \([B]\).

\[
[C] = [A][B]
\]

Consequently, due to product of design matrix \([A]\) and construction matrix \([B]\), a mathematical relation of (4) exists between FRs and DBMs.

\[
\{FR\} = [C]\{DBM\}
\]

\[
\text{TABLE I}
\]

<table>
<thead>
<tr>
<th>Type No. (symbol)</th>
<th>([A])</th>
<th>([B])</th>
<th>([C])</th>
<th>Overlapping relations</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (FO)</td>
<td>([I])</td>
<td>([I])</td>
<td>([I])</td>
<td>DEM1 | Dem1 - Construction 1</td>
<td>Full Overlapping : Two DBMs are independent. 100% overlapping can be fulfilled</td>
</tr>
<tr>
<td>2 (OC)</td>
<td>([I])</td>
<td>([LT])</td>
<td>([LT])</td>
<td>DEM1 | Dem1 - Construction 1</td>
<td>Sequence Construction : Overlapping could be applied to the design phase, while sequence construction phase due to decoupled overlapping</td>
</tr>
<tr>
<td>3 (SD)</td>
<td>([LT])</td>
<td>([I])</td>
<td>([LT])</td>
<td>DEM1 | Dem1 - Construction 1</td>
<td>Sequence Design : Sequence design was necessary, and construction could only start as its corresponding design was completed</td>
</tr>
<tr>
<td>4 (S)</td>
<td>([LT])</td>
<td>([LT])</td>
<td>([LT])</td>
<td>4 Triangular - triangular | |</td>
<td>Sequencing : Unidirectional sequence from DEM1 to DEM2 occurred both to design and construction phase</td>
</tr>
<tr>
<td>5 (OC)</td>
<td>([I])</td>
<td>([X])</td>
<td>([X])</td>
<td>DEM1 | Dem1 - Construction 1</td>
<td>Interlocked Construction : Interlocking occurred between construction processes in two DBMs. Overlapping could be applied to the design phase. The final complete time depends on the integration of two building processes</td>
</tr>
<tr>
<td>6 (IL)</td>
<td>([LT])</td>
<td>([UT])</td>
<td>([X])</td>
<td>DEM1 | Dem1 - Construction 1</td>
<td>Counter sequences occurred to the design and construction phases, so that no overlapping space between two DBMs</td>
</tr>
<tr>
<td>7 (F)</td>
<td>([X])</td>
<td>([X])</td>
<td>([X])</td>
<td>DEM1 | Dem1 - Construction 1</td>
<td>Full Feedback : Interlocking occurred between two DBMs. No overlapping space was allowed</td>
</tr>
</tbody>
</table>

\[
\text{Fig. 6. Example of concurrent matrix } [C] \text{ and its corresponding junction structure diagram.}
\]

After the modules of a project are defined, a flow diagram can be used as another means of representing the system architecture. To construct the flow diagram, this study adopts a specific means of representing the relationship between modules as shown in Figure 7.

\[
\text{Fig. 7. Representation of the design at each junction of a flow chart. (1) Summing junction (uncoupled module, S). (b) Control junction (decoupled module, C). (c) Feedback junction (coupled module, F) }
\]

According to the defined relationship, then, a flow diagram of the system can be completed. Following this procedure, a real design-build case was analyzed and its structural system design-build module (part of fast-tracking model) was shown in the Figure 8.
IV. CROSS-ORGANIZATIONAL PROCESS REENGINEERING

Each design-build module in the fast-tracking model is fulfilled through executions of designer’s and of constructor’s business processes; that is, the designer and the constructor need to cooperate to achieve the functional requirement of each DBM. Therefore, this study subsequently addresses a cross-organizational process reengineering method to integrate processes in a DB team, since design-build process flows across two strategic business units.

Interoperability of process is the main focus of the cross-organizational process reengineering. Hence, the purpose of this phase is to integrate the business processes of designer and of constructor from an information processing viewpoint based on the input-operation-output paradigm.

Moreover, the integrated process models in this phase are general ones for all DBMs in the fast-tracking model; i.e., they can be applied to all DBMs after being specified data entities of specific process.

Figure 9 shows the procedure to reengineer the designer’s and the constructor’s processes.

A. Process Modeling

To integrate both processes of designer and of constructor, a process model providing formal representation of characteristics of processes is necessarily constructed from the beginning of process reengineering [6]. Since this study aiming at the process information integration, a data-oriented modeling process was applied including three tasks, namely, (1) graphic process model creation, (2) semantic process model identification, and (3) ER model mapping. Figure 10 shows the mapping relationship of graphic and of semantic process models. The graphic process model is represented with eEPC diagram of ARIS modeling language [8]. The semantic process model is mapped from graphic model into four subsets, namely, \( f_1: \{\text{process name}\}, f_2: \{\text{process input data}\}, f_3: \{\text{process output data}\} \) and \( f_4: \{\text{activity set}\} \) [7]. Meanwhile, each activity set is also composed of its name, input and output subsets.

![Fig. 9. Cross-organizational process reengineering procedure](image)

![Fig. 10. Subcontractor Management Process model: (a) Graphic Process Model; (b) Semantic Process Model](image)

Each process has its corresponding ER model derived from the graphic and textual process model. Process ER model provides a conceptual level of description, independent of the support on which data are stored, to facilitate data analysis and integration for reengineering purposes. Figure 11 shows an example of ER model derived from process model in Figure 10.

![Fig. 11. Example of ER model (Subcontractor Management Process)](image)

B. Semantic Hierarchy Creation

This study evaluates similarities of data and of activities in the process model based on the affinity of semantic concepts of two entities. Thus, a semantic hierarchy corresponding to the process models is created to depict the concept relationships of data and of activities entities, so that the semantic distance between two entities can be identified, and their semantic affinity can then be calculated.

1. Create data thesaurus of process model

   To present the semantic relationships among data entities, the data thesaurus is organized as an associative network in which
nodes correspond to names of data entities and labeled linkages between nodes present terminological relationships. Figure 12 shows the example of data thesaurus derived from ER model in Figure 11. The \( n_i \) and \( n_j \) are names of data entities, and \( l_{ij} \) is the labeled linkage of \( n_i \) and \( n_j \). A labeled linkage is a triplet \( l_{ij}=(r, a, k) \), where \( r \in \{\text{SYN}, \text{BT}/\text{NT}, \text{RT}\} \) is the type of semantic relationship; \( a \) is the affinity associated with the relationship, and \( k \) is the number of occurrence of the relationship type for the considered pair of names. SYN, BT/NT, and RT are three types of semantic relationships addressed by Castano and Antonellis [7]. Their definitions are described as following.

- SYN (SYNonymy): two entities are synonyms while they can be replaced each other in all schemas without changes in meaning. The relationship affinity \( a_{\text{syn}} = 1 \).
- BT/NT (Broader/Narrow Terms): two names are defined as BT (or NT) relationship while one has a more (or less) general meaning than the other. The relationship affinity \( a_{\text{BT}} = a_{\text{NT}} = 0.8 \).
- RT (Related Terms): two names \( n_i \) and \( n_j \) if an entity \( e_i \) with name \( n_i \) participates in a relationship with an entity \( e_j \) with name \( n_j \) in some schema. The relationship affinity \( a_{\text{RT}} = 0.5 \).

These three relationships are defined as explicit relationships because they can be determined directly according to relations in the ER model. However, since some degree of similarity exists between two entities due to their participation in a relationship with a third entity, the implicit relationships between any two entities are necessary to determine. The implicit relationships can be derived from the explicit ones; that is, implicit relationships correspond to paths of explicit relationships between names in the thesaurus.

\[
S(n_i, n_j) = S_{\text{hom}}(a_{ij}) = a_{ij}^{1/4}
\]

were \( S(n_i, n_j) \) is the similarity of \( n_i \) and \( n_j \), \( S_{\text{hom}} \) is the homogeneous semantic similarity function; \( a_{ij} \) is the affinity of relationship of \( n_i \) and \( n_j \), and \( k \) is the number of occurrence of the involved relationship.

\[
S(n_i, n_j) = S_{\text{het}}((S_{\text{hom}}(a_{ij-BT}), S_{\text{hom}}(a_{ij-RT}))) = (S_{\text{hom}}(a_{ij-BT}) + S_{\text{hom}}(a_{ij-RT})) - (S_{\text{hom}}(a_{ij-BT}) \cdot S_{\text{hom}}(a_{ij-RT}))
\]

were \( S_{\text{het}} \) is the heterogeneous semantic similarity function; \( a_{ij-BT} \) is the affinity of BT relationship of \( n_i \) and \( n_j \), and \( a_{ij-RT} \) is the affinity of RT relationship of \( n_i \) and \( n_j \).

\[
S(n_i, n_j) = S_{\text{imp}}(a_{ij}, a_{hi}) = a_{ij} \cdot a_{hi}
\]

where \( S_{\text{imp}} \) is the similarity function of implicit relationship; \( a_{ij} \) is the explicit relationship strength of \( n_i \) and \( n_j \); \( a_{hi} \) is the explicit relationship strength of \( n_h \) and \( n_j \).

Applying these equations, the semantic similarity matrix can then be created based on the data thesaurus. Figure 13 shows an example of semantic similarity matrix derived from Figure 12, which represents semantic similarities of all pairs of data entities.

3. Data semantic hierarchy creation

The semantic hierarchy is the foundation for analyzing similarities of processes and activities. It presents the similar concept clusters of data of the process model. The concepts at the top of the hierarchy are either more general than the lower level concepts or are composed of them. Each concept at the bottom level of the hierarchy has associated a cluster of names
corresponding to entities that are semantically similar. Names contained in clusters are entity names used in the process model. Based on the semantic similarity matrix, the data concept hierarchy can then be created with clustering analysis. Figure 14 shows the partial semantic hierarchy of data.

\[ ISARC 2005 \ 7 \]

Moreover, the name set affinity parameter denoted by \( A(X,X') \) can be derived from name affinity parameter. Equation (9) shows the function of name set affinity parameter.

\[ A(X,X') = \sum_{n_k \in n_x} A(n_i,n_j) \quad \forall n_i \in n_x , \forall n_j \in n_x \]  

\[ (9) \]

where the \( n_x \) is the name set \( X \) and \( n_x' \) is the name set \( X' \), and \(|n_x| \) and \(|n_x'| \) are numbers of elements respectively in two sets.

By calculating all pairs of name entities belonging to two sets, the name set affinity parameter presents the affinity strength of two name sets.

Based on the name set affinity parameter, three types of process similarity are applied in this study, namely, Activity Similarity (ASim), Process Information Similarity (PISim), and Process Functional Similarity (PFSim). The process information similarity denoted by \( PISim(P_i,P_j) \) is the measure of affinity degree of input and output information sets corresponding to several processes \( P_i \) and \( P_j \).

Equation (10) shows the mathemetic function of information similarity.

\[ PISim(P_i,P_j) = \sum_{f_j \in D_j} A(D_i,f_j,D_j,f) \]  

\[ (10) \]

where the \( D_i \) and \( D_j \) are respectively the semantic process model of process-i and process-j; \( \zeta = f_2,f_3 \) of \( D_k \), \( f_5 = IN(P_k) \), \( f_7 = OUT(P_k) \), and \( 0 \leq ISim(P_i,P_j) \leq 2 \).

High process information similarity only provides a macroscopic evaluation factor which implies that two processes have high potential for integration due to their similar information features, and vice versa. Therefore, a microcosmic view form activities of a process is necessary for advanced similarity analysis. Equation (11) expresses the function of activity similarity denoted by \( A\text{sim} \).

\[ A\text{Sim}(A_{hi},A_{kj}) = A\text{Sim}(A_{hi},A_{kj}) + A\text{Sim}(A_{ki},A_{kj}) + A\text{Sim}(A_{hi},A_{kj}) + \]  

\[ A\text{Sim}(A_{hi},A_{kj}) + (11) \]

where the \( A_{hi} \) is the \( h \)th activity in the process-i; \( A_{kj} \) is the \( k \)th activity in the process-j; \( A\text{Sim}(A_{hi},A_{kj}) \) is activity similarity of \( A_{hi} \) and \( A_{kj} \); \( A\text{Sim}(A_{hi},A_{kj}) \) is the input set of \( A_{hi} \), \( OUT(A_{hi}) \) is the output set of \( A_{hi} \), and \( 0 \leq A\text{Sim}(A_{hi},A_{kj}) \leq 3 \).

Not only the information similarity \( A\text{Sim}(A_{hi},A_{kj}) \), \( A\text{Sim}(A_{hi},A_{kj}) \), but the functional similarity \( A\text{Sim}(A_{hi},A_{kj}) \) is also concerned in the activity similarity function. Therefore, high activity similarity expresses that two activity are highly recommended for integration.

Moreover, summarize all activity similarities of two processes by (12), the Process Functional Similarity can finally
be calculated. Similar to $PISim$, the process functional similarity provides a macroscopic evaluation factor for determining two process need to be integrated or not.

$$PFSim(P_i, P_j) = \sum_{i,j} P_i \cdot \sum_{i,j} P_j$$

where $|P_i|$ and $|P_j|$ are numbers of activities of process-$i$ and of process-$j$, and $0 \leq PFSim(P_i, P_j) \leq 3$.

D. Cross-organizational process integration

According to the process similarity factors, this study addressed an application procedure for process integration which shown in the Figure 16.

Following the procedure in Figure 16, a general integrated design-build process model can be created. In the process integration model, cooperation linkages will be determined to demonstrate the cooperative operations between designer’s and constructor’s processes. Figure 17 shows a partial integrated process model (from detailed design to purchasing) of an experiment on a case study. Two cooperation linkages were determined with the process similarity analysis, namely, “select subcontractors” and “budget inspection”. For the “select subcontractors” linkage, the constructor is responsible for providing information of qualified subcontractors according to the development design, and several subcontractors are selected to be involved in the detailed design works. For the “budget inspection” linkage, the architect provides the preliminary budget, and the constructor needs to adjust the budget according to the estimation information derived from price negotiation/comparison or estimation database.

V. DISCUSSION

The Integrated Design-Build Framework is a research project, whose long-term goal is to develop a process integrating method and finally to create a multi-agent system to support the cooperation in a design-build team. This study is firstly aimed at creating the fast-tracking model and integrating designer’s and constructor’s processes in a design-build team from the information processing viewpoint. On the one hand, the fast-tracking model is created for enhancing the speed of a design-build project and it provides a dependency relation map of all design-build phases for process management in high velocity, complex projects. On the other hand, the process integration model demonstrates the links of processes of designer and of constructor for facilitating the process cooperation in a design-build team. Based on design-build phase relations and process/activity information, a multi-agent system could possibly be developed in the next step of this study.

REFERENCES