Proceedings of DETC '00:
ASME 2000 International Design Engineering Technical Conferences
and Computers and Information in Engineering Conference
Baltimore, Maryland, September 10-13, 2000

DETC2000/DTM-14571

DESIGNING MODULAR AND INTEGRATIVE SYSTEMS

Manuel E. Sosa
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts
msosa@mit.edu

Steven D. Eppinger[1]
Sloan School of Management
Massachusetts Institute of Technology
Cambridge, Massachusetts
eppinger@mit.edu

Craig M. Rowles
Advanced Engine Program
Pratt & Whitney Aircraft
East Hartford, Connecticut
rowlescm@pweh.com

ABSTRACT

This paper describes a method that allows us to enhance our understanding of the difference between designing modular systems and integrative systems. Modular systems are those whose interfaces are well defined and shared with only a few other systems. Integrative systems are those whose interfaces may be more complex and shared across the product. Our approach is illustrated by analyzing the development of a large commercial aircraft engine. We document both the product's design interfaces and the technical interactions between design teams. We found statistically significant differences in the ways modular and integrative design teams handle design interfaces. We focus our analysis on studying the effects due to organizational and system boundaries, and to the existence of various types of design interfaces. By identifying modular and integrative systems and by understanding the differences in designing those systems, development organizations can improve the integration process for complex designs.

Keywords: Design interfaces, integrative systems, modular systems, product architecture, systems engineering, team interactions.

INTRODUCTION

Understanding how the development organization manages the knowledge associated with the product architecture has been recognized as a critical challenge for established firms facing architectural innovation (Henderson and Clark, 1990). This paper presents a research method intended to enhance our understanding of the coupling between the product architecture and the development organization. We are particularly interested in studying the differences in designing integral versus modular systems.

Ulrich (1995) defines several types of architectures according to how the product's functions are mapped onto its physical components. A key feature of product architecture is the degree to which it is modular or integral. Modular architectures exhibit direct mapping between functions and physical elements, and have well-defined interfaces between physical components. On the other hand, in integral architectures functions are spread across components, resulting in more complex interfaces (Ulrich and Eppinger, 1995). In very complex products, we apply these definitions at the level of the many systems (and subsystems) which comprise the product. We will refer to modular systems as those exhibiting modular architecture characteristics while integrative systems are those with features of integral architectures.

We are particularly interested in the development of complex products, such as an automobile, a computer, or an aircraft engine. These types of products usually involve the design of both modular and integrative systems. The general approach when developing complex products is to decompose the product into systems, and if the systems are still too complex, decompose these into smaller components (Alexander (1964), Pimmler and Eppinger (1994), Eppinger (1997)).

[1] Please address all the correspondence to this author.
50 Memorial Drive. E53-347. Cambridge, MA 02142
Tel: (617) 253-0468

From an organizational viewpoint, design teams in complex product development are commonly organized around the architecture of the product. In most technical products we can observe a clear mapping between the product architecture and the development organization which designs it. However, little research has focused on understanding the effects of designing various types of architectures. We call modular design teams those that design modular systems while integrative design teams are those that design integrative systems.

Design teams face two important levels of integration during the development of complex products. Function-level integration takes place within each cross-functional design team when they have to coordinate efforts in order to design their respective components. System-level integration takes place across design teams in order to integrate the components (designed by each team) to assure the product works as an integrated whole.

This paper focuses on understanding the system-level integration efforts faced by both modular design teams and integrative design teams. In particular, we want to better comprehend how modular and integrative design teams face barriers imposed by organizational and system boundaries. We also want to investigate whether design teams must apply greater effort toward addressing certain types of design interfaces.

OUR APROACH

We summarize our research method in three steps:

1. **Capture the product architecture**. We first identify how the product is decomposed into systems, and these further into components. We then document the design interfaces between them. Lastly, we analyze the distribution of cross-systems design interfaces to identify modular and integrative systems.
2. **Capture the development organization**. We first identify the design teams responsible to develop the product's components. We then survey key members of each team to capture the technical interactions between them.
3. **Compare the product architecture and the development organization**. We compare the design interfaces with the team interactions in order to study how the product architecture drives technical communication in the development organization.

AN EXAMPLE

We apply this approach to the design of a large commercial aircraft engine. The project chosen was a complex design which involved the development of both modular and integrative systems. The engine was decomposed into eight systems, and these further decomposed into 54 components. The development team was organized according to the architecture of the product, with 54 cross-functional teams assigned to design the 54 components. Figure 1 exhibits a cross-section diagram of the engine studied. For more details of the project description and data collection refer to Rowles (1999).

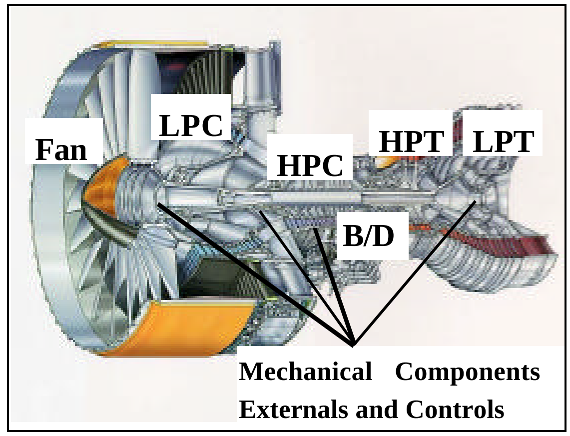

FIGURE 1: LARGE COMMERCIAL AIRCRAFT ENGINE

CAPTURING THE PRODUCT ARCHITECTURE

The engine studied was decomposed into eight systems. Each of these systems was further decomposed into five to ten components each (see Table 1).

TABLE 1: SYSTEMS AND COMPONENTS OF THE ENGINE STUDIED

System	Number of Components
Fan	7
Low-Pressure Compressor (LPC)	7
High-Pressure Compressor (HPC)	7
Burner and Diffuser (B/D)	5
High-Pressure Turbine (HPT)	5
Low-Pressure Turbine (LPT)	6
Mechanical Components	7
Externals and Controls	10

Researchers in Engineering Design (Suh (1990), Pahl and Beitz (1991)) have modeled functional requirements of product design in terms of exchanges of energy, materials, and signals between elements. Building upon the methodology presented by Pimmler and Eppinger (1994) we identified five types of design dependencies between the 54 components:

- **Spatial** dependency indicates a requirement related to physical adjacency for alignment, orientation, servicability, assembly, or weight.
- **Structural** dependency indicates a requirement related to transferring loads, or containment.
- **Energy** dependency indicates a requirement related to transferring heat energy, vibration energy, electrical energy, or noise.
- **Material** dependency indicates a requirement related to transferring airflow, oil, fuel, or water.
- **Information** dependency indicates a requirement related to transferring signals or controls.

The design dependencies were captured by interviewing design experts. These people had a deep understanding of the product architecture, but they were not directly involved in the design of the engine. We mapped the design-interface data into a square (54x54) design interface matrix. (The design interface matrix can be described as a special form of design structure matrix (DSM). For a formal introduction to DSM refer to Steward (1981) and Eppinger *et al*. (1994).) The identically labeled rows and columns of this matrix name the 54 components of the engine, and their sequencing follows the physical arrangement of the systems within the product. Indeed, the systems are sequenced following the airflow through the engine. Each cell of the design interface matrix is actually a five-component vector corresponding to the five types of design dependencies which categorize each interface.

For graphical simplicity Figure 2 exhibits the binary design interface matrix of the engine studied. The off-diagonal elements of the matrix are marked with an "X" for each pair of components that shares at least one design interface. The matrix is not completely symmetric with respect to its diagonal due to the fact that each row captures the dependencies necessary for one component's functions.

IDENTIFYING MODULAR AND INTEGRATIVE SYSTEMS

The first six systems of the design interface matrix are those in which design interfaces are primarily among adjacent components (the modular systems). On the other hand, the mechanical components system, and the externals and controls system exhibit design interfaces distributed throughout the engine (the integrative systems).

To determine the degree of modularity (and integrality) of each system we analyze the distribution of design interfaces across system boundaries. System boundaries are highlighted by the boxes around the diagonal of the design interface matrix. Marks inside the boxes represent design interfaces between components of the same system, whereas marks outside the boxes indicate interfaces between components of different systems.

Figure 3 shows the number of design interfaces between the externals and controls system and the six modular systems. Similarly, Figure 4 shows the number of design interfaces between the mechanical components system and the first six modular systems. To visually compare the difference in distribution between modular and potential integrative systems, Figure 5 shows the distribution of the design interfaces between the high-pressure compressor (HPC), which is the least modular of the six systems, and the other modular systems.

We utilize chi-square (χ^2) statistical analysis to compare observed values with expected values, given various frequency distributions (Rice, 1994). In doing so, we assume that the cells on each of the matrices shown below are statistically independent. Using log-linear network analysis Sosa (2000) validates this assumption.

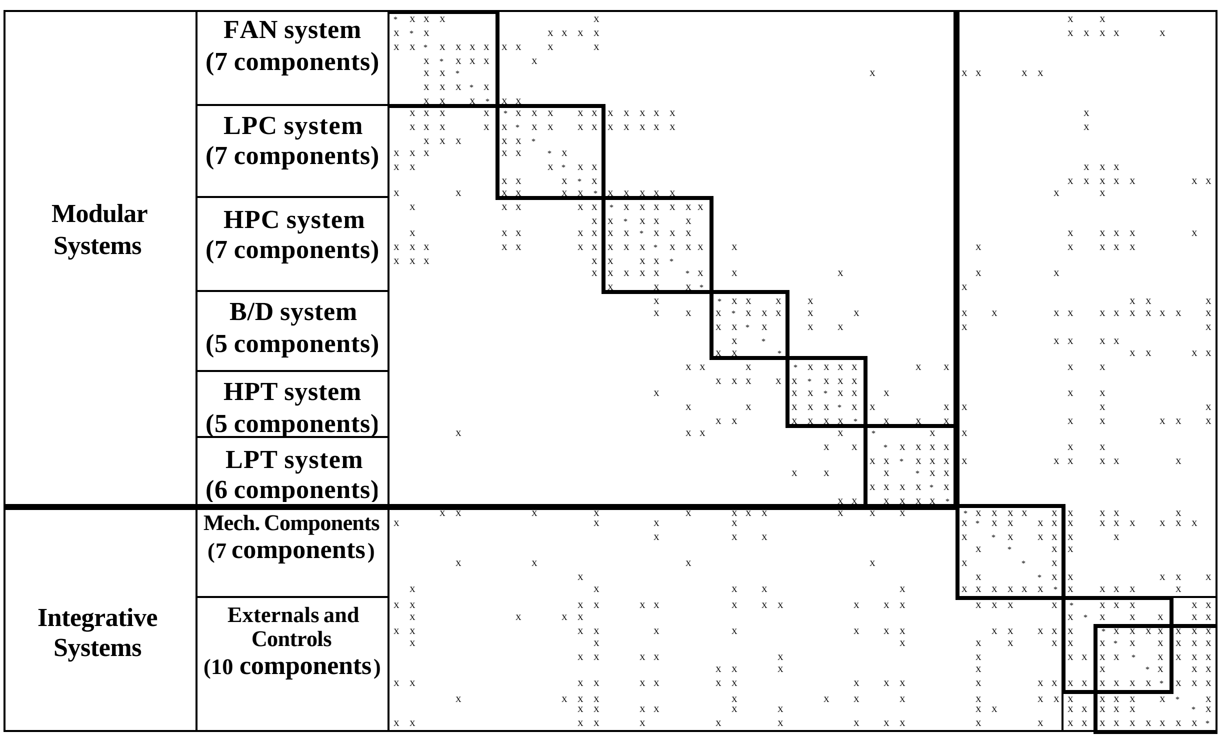

FIGURE 2: DESIGN INTERFACE MATRIX

Table 2 shows the results of the chi-square test performed to test the alternative hypothesis that "the distribution of design interfaces of the externals and controls system is statistically significant different than the distribution of the high-pressure compressor system". The test resulted in a χ^2 equal to 29.880 which is greater than the critical value of 9.488 (for α=0.05 and four degrees of freedom). The expected values shown in Table 2 are based on the distribution of the design interfaces of the externals and controls system. The actual values are the number of design interfaces of the high-pressure compressor system. Similar results were found when comparing the distribution of cross-system design interfaces of the externals and controls system and the other five modular systems. (For more details refer to Sosa, 2000.)

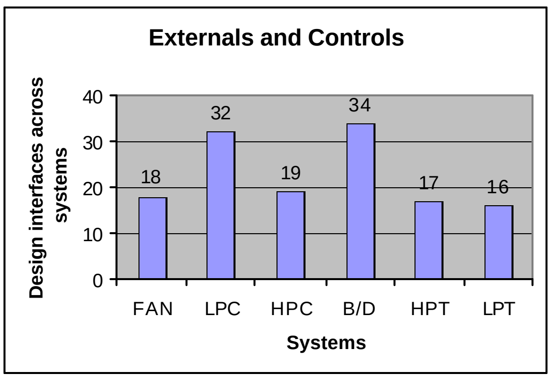

FIGURE 3: DISTRIBUTION OF DESIGN INTERFACES OF THE EXTERNALS AND CONTROLS SYSTEM

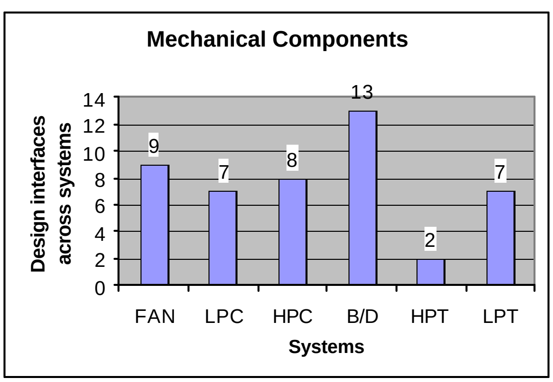

FIGURE 4: DISTRIBUTION OF DESIGN INTERFACES OF THE MECHANICAL COMPONENTS SYSTEM

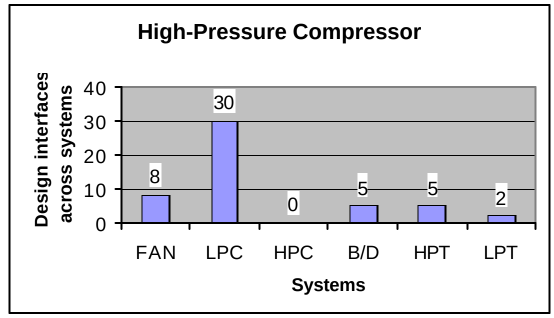

FIGURE 5. DISTRIBUTION OF DESIGN INTERFACES OF THE HPC SYSTEM

We found that the distribution of design interfaces of the externals and controls system, and the mechanical components system are similarly distributed among the first six modular systems. Table 3 shows the results of the chi-square test performed to test the null hypothesis that "the distribution of design interfaces of the mechanical components system, and the distribution of design interfaces of the externals and controls system are statistically equivalent". The test resulted in χ^2 equal to 6.237 which is smaller than the critical value of 11.070 (for α=0.05 and five degrees of freedom).

These results confirm our initial conjecture regarding modularity, that the first six systems are modular systems while the remaining two systems are integrative systems.

TABLE 2: CHI-SQUARE TEST RESULTS. COMPARING EXTERNALS AND CONTROLS SYSTEM WITH HIGH-PRESSURE COMPRESSOR SYSTEM

System	Expected fraction of design interfaces based on Ext/Controls	Expected number of design interfaces of HPC	Actual number of design interfaces of HPC	χ^2
FAN	15.38%	7.692	8	0.012
LPC	27.35%	13.675	30	19.488
B/D	29.06%	14.530	5	6.251
HPT	14.53%	7.265	5	0.706
LPT	13.68%	6.838	2	3.423
Total	100.00%	50.000	50	**29.880**

TABLE 3: CHI-SQUARE TEST RESULTS. COMPARING EXTERNALS AND CONTROLS SYSTEM WITH MECHANICAL COMPONENTS SYSTEM

System	Expected fraction of design interfaces based on Ext/Controls	Expected number of design interfaces of Mech. Comps.	Actual number of design interfaces of Mech. Comps.	χ^2
FAN	13.24%	6.088	9	1.393
LPC	23.53%	10.824	7	1.351
HPC	13.97%	6.426	8	0.385
B/D	25.00%	11.500	13	0.196
HPT	12.50%	5.750	2	2.446
LPT	11.76%	5.412	7	0.466
Total	100.00%	46.000	46	**6.237**

CAPTURING THE DEVELOPMENT ORGANIZATION

Fifty four design teams were grouped into eight system-design groups according to the architecture of the engine described above. Each of those teams was responsible for developing one of the 54 components of the engine. We capture the system-level integration effort of the organization by documenting the technical interaction between the design teams involved in the development process. We asked at least two key members from each design team to rate the criticality and frequency of their interactions with each of the other teams during the detailed design phase of the engine's development project. This method is similar to the approach illustrated by Eppinger (1997).

We organize the team-interaction data in a square (54x54) team-interaction matrix (see Figure 6). The identically ordered labels of the rows and columns of this matrix contain the names of each of the design teams. The binary team interaction matrix shows off-diagonal cells marked with an "O" to indicate each non-zero team interaction revealed.

It is important to emphasize that we documented coordination-type communication only. Morelli *et al.* (1995) and Allen (1997) identified three types of technical communication in development organizations: coordination-type, knowledge-type and inspiration-type. Since we were interested in capturing the integration effort of the development organization we focused our surveys upon the coordination-type interactions which took place during the design process to address task-related issues. Since we captured the interactions reported from the respondent's point of view, the matrix exhibits an asymmetric structure with respect to its diagonal.

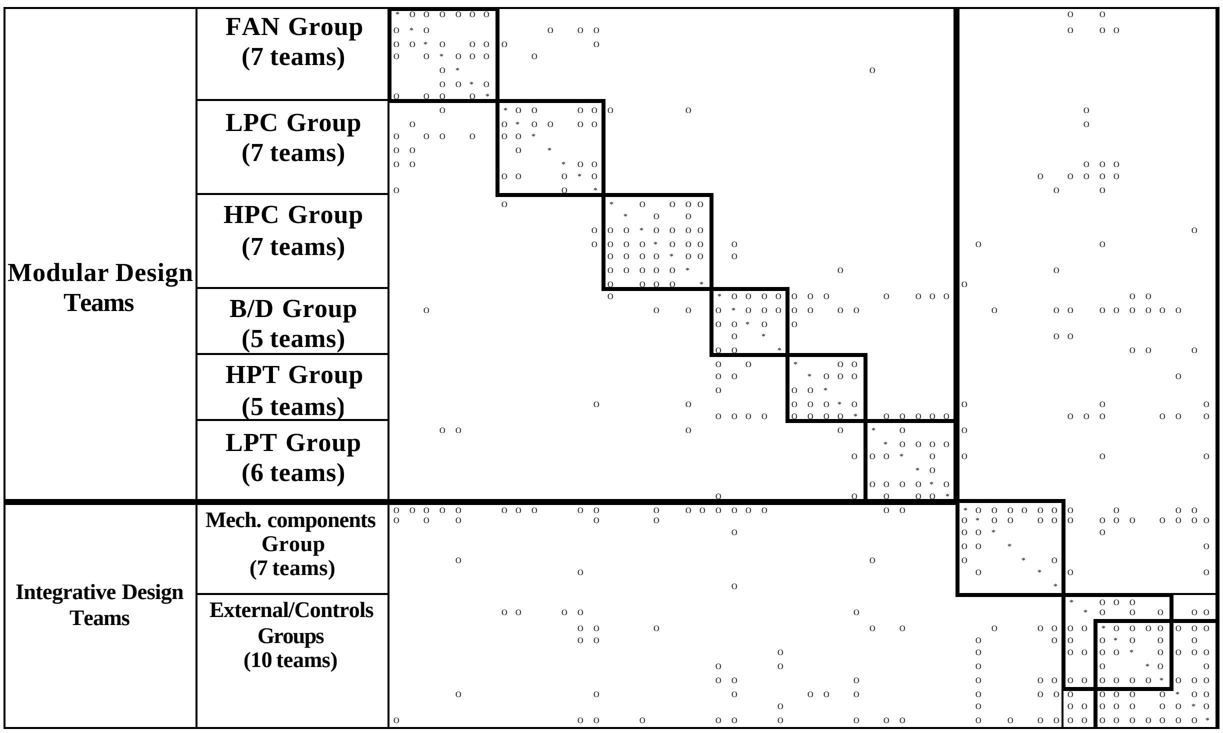

FIGURE 6: TEAM INTERACTION MATRIX

COMPARING THE PRODUCT ARCHITECTURE WITH THE DEVELOPMENT ORGANIZATION

The one-to-one assignment of the 54 components to the 54 design teams allows the direct comparison of the design interface matrix with the team interaction matrix. Figure 7 shows how, by overlapping the design interface matrix with the team interaction matrix, we obtain the resultant matrix.

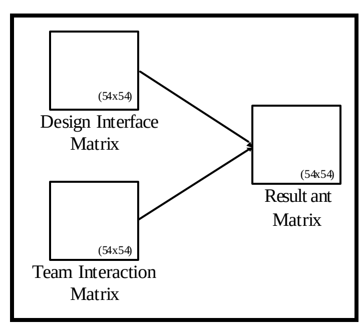

FIGURE 7: COMPARING DESIGN INTERFACES AND TEAM INTERACTIONS

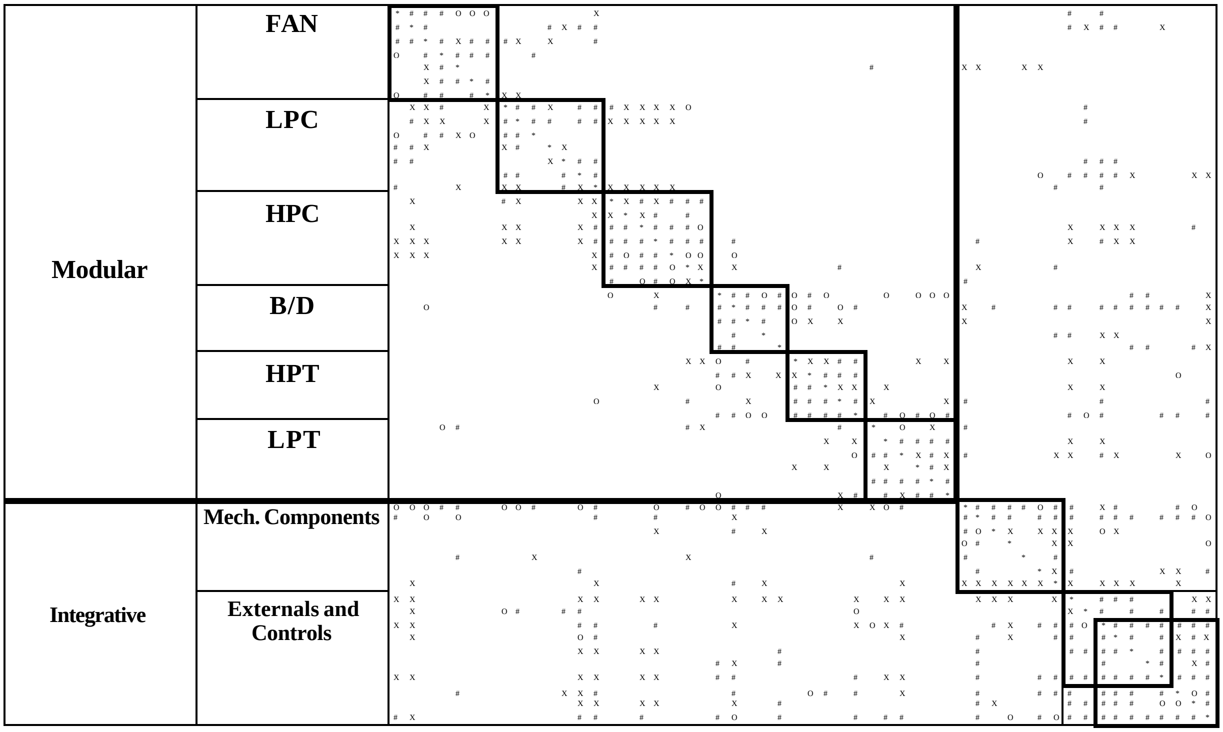

FIGURE 8: RESULTANT MATRIX

Legend:
X: Design interface with no team interaction
O: Team interaction with no design interface
#: Both design interface and team interaction
***:** Diagonal elements (meaningless)

		Design Interfaces: YES (569)	Design Interfaces: NO (2293)
Team Interactions	NO (2439)	**X** (220)	(2219)
Team Interactions	YES (423)	**#** (349)	**O** (74)

FIGURE 9: RESULTS OF THE RESULTANT MATRIX

Figure 9 shows the four possible outcomes for each cell of the resultant matrix exhibited in Figure 8. The number of cases in each category is given in parentheses.

As expected, the majority of the cells (90% of the cells) are the cases where known design interfaces were matched by team interactions (349 "#" cells), and the cases with no design interfaces with no corresponding reported team interactions (2219 blank cells). The unexpected cases accounted for 10% of the cells; those were the cases when known design interfaces were not matched by team interactions (220 "X" cells), and the cases when reported team interactions were not predicted by design interfaces (74 "O" cells). Sosa (2000) presents detailed analyses that test some of the hypotheses that explain the existence of the unexpected cases.

This paper focuses on the differences of designing modular systems and integrative systems. By analyzing the resultant matrix, we investigate the effects of system and organizational boundaries, and the effects due to the types of design interfaces on the development of modular and integrative systems.

EFFECTS OF ORGANIZATIONAL BOUNDARIES

Organizational boundaries are defined by the way the design teams are grouped into system teams. Similarly, system boundaries are defined by the way components form systems. Boundaries are highlighted in the resultant matrix by the boxes along the diagonal. Cells inside the boxes represent the interactions within (organizational and system) boundaries while cells outside the boxes represent the interactions across boundaries.

Organizational boundaries impose communication barriers which prevent design teams to interact (Allen (1977), Van den Bulte and Moenaert (1998), Sosa *et al*. (2000)).

However, given the distributed nature of integrative systems we expect integrative design teams to be less affected by this type of barrier. Since the design interfaces of integrative systems are distributed throughout the engine, we anticipate integrative design teams to be more accustomed to crossing organizational boundaries than are modular design teams. Indeed, we hypothesize that integrative design teams handle a larger portion of design interfaces across organizational boundaries than do modular design teams. Note that we are implicitly assuming that the coordination-type communication reflects handling the corresponding design interface. To test this hypothesis, we use the 569 cases in which design interfaces were identified ("#" and "X" cells of Figure 9) to perform chi-square tests.

The chi-square tests shown in Table 3 resulted in a χ^2 equal to 0.095 for the cases within boundaries, which is well below the critical value of 3.841 (for α=0.05 and one degree of freedom). On the other hand, for the cases across organizational boundaries χ^2 equaled 8.740, which is well above the critical value.

These results allow us to accept the null hypothesis that design interfaces within organizational boundaries are equally handled by teams that design modular systems as by teams that design integrative systems. However, by analyzing the cases across organizational boundaries, we found that integrative design teams handle a statistically significant higher portion of design interfaces than do modular design teams. Specifically, integrative design teams matched 53.5% of the cross-system design interfaces while modular design teams matched 36.4% of their cross-system design interfaces.

TABLE 3: CHI-SQUARE TEST RESULTS. EFFECTS OF ORGANIZATIONAL BOUNDARIES†

	Total	Expected cases of design interfaces matched by team interactions	Expected cases of design interfaces not matched by team interactions	Actual cases of design interfaces matched by team interactions	Actual cases of design interfaces not matched by team interactions	χ^2 of cases of design interfaces matched by team interactions	χ^2 of cases of design interfaces not matched by team interactions
Design interfaces within organizational boundaries (Modular systems)	137	110.9 (81.0%)	26.1 (19.0%)	110 (80.3%)	27 (19.7%)	0.007	0.031
Design interfaces within organizational boundaries (Integrative systems)	94	76.1 (81.0%)	17.9 (19.0%)	77 (81.9%)	17 (18.1%)	0.011	0.046
Total	231	187.0	44.0	187	44	**0.018**	**0.077**
Design interfaces across organizational boundaries (Modular systems)	110	52.7 (47.9%)	57.3 (52.1%)	40 (36.4%)	70 (63.6%)	3.070	2.826
Design interfaces across organizational boundaries (Integrative systems)	228	109.3 (47.9%)	118.7 (52.1%)	122 (53.5%)	106 (46.5%)	1.481	1.363
Total	338	162.0	176.0	162	176	**4.551**	**4.189**

† Expected values are determined with the pooled data which indicates that 81.0% of the 231 design interfaces within organizational boundaries are matched by team interactions while 47.9% of the 338 design interfaces across organizational boundaries are matched by team interactions.

$\chi^2_{\text{within-boundaries}} = 0.095$ $\chi^2_{\text{across-boundaries}} = 8.740$ $\chi^2_{\text{critical (0.95,1)}} = 3.841$

EFFECTS OF SYSTEM BOUNDARIES

System boundaries impose architectural knowledge barriers which inhibit design experts' understanding of certain design interfaces. This results in some team interactions that are not predicted by design interfaces. Indeed, Sosa (2000) shows that team interactions within system boundaries are more likely to be predicted than team interactions across system boundaries. A question we want to address in this paper is whether the effects of system boundaries are the same for modular design teams as for integrative design teams.

To answer such a question, we use the sample formed by the 423 team interactions ("#" and "O" cells from Figure 9) to test whether there is a statistically significant difference in the way design interfaces predict team interactions of modular versus integrative design teams. The chi-square tests shown in Table 4 resulted in a χ^2 equal to 0.483 for the cases within system boundaries, which is well below the critical value of 3.841 (for α=0.05 and one degree of freedom). On the other hand, for the team interactions across system boundaries the χ^2 equaled 9.566, which is well above the critical value.

These results allow us to accept the null hypothesis that the portion of unknown design interfaces within system boundaries is equivalent for both modular and integrative systems. However, for interactions across system boundaries we found that the portion of unknown design interfaces across modular systems is statistically significant larger than for integrative systems. Note that unknown design interfaces are characterized by unpredicted team interactions ("O" cells). Specifically, 38.5% of the modular design team interactions across system boundaries were not predicted by design interfaces whereas only 18.7% of the integrative design team interactions across system boundaries were not predicted by design interfaces.

EFFECTS OF TYPES OF DESIGN INTERFACE

According to the type of design dependency, we classify design interfaces into two major categories:

- **Spatial-type** design interfaces, which involve spatial dependencies only.
- **Transfer-type** design interfaces, which involve structural and/or energy and/or material dependencies. Information dependencies are not included in this analysis because they are not present in the modular systems.

Henderson and Clark (1990) refer to communication filters as the mechanism for screening the most crucial information. Adapting this concept to our context, we expect some design teams to handle a larger proportion of some types of design interfaces than other ones. Hence, we want to investigate whether there is a difference in the way modular and integrative design teams handle these two types of design interfaces.

TABLE 4: CHI-SQUARE TEST RESULTS. EFFECTS OF SYSTEM BOUNDARIES†

	Total	Expected cases of predicted team interactions	Expected cases of unpredicted team interactions	Actual cases of predicted team interactions	Actual cases of unpredicted team interactions	χ2 of predicted team interactions	χ2 of unpredicted team interactions
Design interfaces within system boundaries (Modular systems)	124	111.5 (89.9%)	12.5 (10.1%)	110 (88.7%)	14 (11.3%)	0.020	0.175
Design interfaces within system boundaries (Integrative systems)	84	75.5 (89.9%)	8.5 (10.1%)	77 (91.7%)	7 (8.3%)	0.029	0.259
Total	208	187.0	21.0	187	21	**0.049**	**0.434**
Design interfaces across system boundaries (Modular systems)	65	49.0 (75.3%)	16.0 (24.7%)	40 (61.5%)	25 (38.5%)	1.645	5.029
Design interfaces across system boundaries (Integrative systems)	150	113.0 (75.3%)	37.0 (24.7%)	122 (81.3%)	28 (18.7%)	0.713	2.179
Total	215	162.0	53.0	162	53	**2.358**	**7.208**

† Expected values are determined with the pooled data which indicates that 89.9% of the 208 design interfaces within system boundaries predict team interactions while 75.3% of the 215 design interfaces across system boundaries predict team interactions.

$\chi^2_{\text{within-boundaries}} = 0.483$ $\chi^2_{\text{across-boundaries}} = 9.566$ $\chi^2_{\text{critical (0.95,1)}} = 3.841$

A subset of 122 design interfaces, which could be categorized as either spatial-type or transfer-type, were used to answer this question (see Table 5). The chi-square test resulted in a χ^2 equal to 4.360 for spatial-type design interfaces, and a χ^2 equal to 6.035 for transfer-type design interfaces. Both are greater than the critical value of 3.841 (for α=0.05 and one degree of freedom). These results allow us to reject the null hypothesis that spatial-type design interfaces and transfer-type design interfaces are equally handled by modular design teams and integrative design teams.

The results obtained support the hypothesis that teams designing modular systems have a stronger preference, ability, or willingness, to deal with spatial-type design interfaces than do teams designing integrative systems. As shown in Table 5, 62.5% of the modular spatial-type design interfaces analyzed were matched by team interactions, while 29.4% of the integrative spatial-type design interfaces were matched by team interactions. Similarly, data also support the hypothesis that teams that design integrative systems are more willing to deal with transfer-type design interfaces than modular design teams. Table 5 shows that 45.0% of the integrative transfer-type design interfaces analyzed were matched by team interactions, while only 19.5% of the modular transfer-type design interfaces were matched by team interactions.

DISCUSSION

This analysis provides three important results which are summarized here:

1. The statistically significant differences in the way integrative design teams handle design interfaces across boundaries suggest that these teams are more efficient at overcoming the barriers imposed by organizational boundaries. The distributed nature of the integrative systems forces these design teams to overcome organizational barriers in order to handle design interfaces with all the systems.
2. When analyzing the effects of system boundaries, we found a statistically significant larger proportion of unpredicted team interactions ("O" cells of Figure 9) associated with modular systems. Since unpredicted team interactions represent unrecognized design interfaces, we conclude that design interfaces across modular systems are more difficult for design experts to recognize than interfaces with integrative systems.
3. The existence of various types of design interfaces and the statistically significant difference in the way they were handled by modular and integrative design teams provide empirical support to the notion of "communication filters" introduced by Henderson and Clark (1990). We found that spatial-type design interfaces are largely addressed in the design of modular systems while transfer-type design interfaces are more likely to be handled in the design of integrative systems.

By studying the coupling of the architecture of an aircraft engine and the development organization that designed it we have gained important insight about the difference between designing modular versus integrative systems. While, we cannot claim the generality of these findings before completing similar studies in other types of products in different industries. We would expect to find similar results in other projects developing complex systems and where the development teams are organized according to the product architecture.

TABLE 5: CHI-SQUARE TEST RESULTS. EFFECTS DUE TO TYPE OF DESIGN INTERFACES†

	Total	Expected cases of design interfaces matched by team interactions	Expected cases of design interfaces not matched by team interactions	Actual cases of design interfaces matched by team interactions	Actual cases of design interfaces not matched by team interactions	$\chi 2$ of cases of design interfaces matched by team interactions	$\chi 2$ of cases of design interfaces not matched by team interactions
Spatial-type design interfaces (Modular systems)	24	11.7 (48.8%)	12.3 (51.2%)	15 (62.5%)	9 (37.5%)	0.926	0.882
Spatial-type design interfaces (Integrative systems)	17	8.3 (48.8%)	8.7 (51.2%)	5 (29.4%)	12 (70.6%)	1.307	1.245
Total	41	20.0	21.0	20	21	**2.233**	**2.127**
Transfer-type design interfaces (Modular systems)	41	13.2 (32.1%)	27.8 (67.9%)	8 (19.5%)	33 (80.5%)	2.024	0.957
Transfer-type design interfaces (Integrative systems)	40	12.8 (32.1%)	27.2 (67.9%)	18 (45.0%)	22 (55.0%)	2.074	0.980
Total	81	26.0	55.0	26	55	**4.098**	**1.937**

† Expected values are determined with the pooled data which indicates that 48.8% of the 41 spatial-type design interfaces are matched by team interactions while 32.1% of the 81 transfer-type design interfaces are matched by team interactions.

$\chi^2_{\text{spatial-type}} = 4.360$ $\chi^2_{\text{transfer-type}} = 6.035$ $\chi^2_{\text{critical (0.95,1)}} = 3.841$

CONCLUSIONS AND FUTURE WORK

The research method presented in this paper provides a useful approach to investigate the coupling of the product architecture and the development organization. This paper provides two important contributions. First, we formally identify modular and integrative systems by analyzing the distribution of design interfaces across system boundaries. Second, we illustrate a method to better understand the difference between designing modular versus integrative systems.

Our research method can be summarized in three steps: 1) capture the product architecture by documenting design interfaces, 2) capture the development organization by documenting team interactions, and 3) couple the product architecture with the development organization by comparing design interfaces with team interactions. This method is particularly applicable to projects where the architecture of the product is well understood and the development team is organized around the product architecture.

The type of analysis illustrated here may outline the study of other issues related to the coupling of product architecture and organizational structure. A challenge for future research work is to extend this method to explore the evolution over time of both design interface matrix and team interaction matrix for several generations in a product family.

This study takes advantage of the direct mapping of the product architecture and the development organization in the project studied. What if this were not the case? Which types of barriers are most severe (organizational or system barriers)? Is an organizational design that mirrors the architecture of the product a good one? Studying various mappings of product architectures to development organizations may help answer these research questions.

ACKNOWLEDGMENTS

Funding of this research has been provided by the MIT Center for Innovation in Product Development. We appreciate the assistance of the engineers from Pratt & Whitney Aircraft during the development of the example presented in this paper.

REFERENCES

Alexander, C. 1964. *Notes on the Synthesis of Form*. Harvard University Press, Cambridge, MA.

Allen, T. J. 1977. *Managing the Flow of Technology*. Cambridge: MIT Press.

Allen, T. J. 1997. "Architecture and Communication Among Product Development Engineers". *Sloan Working Paper*, # 3983, MIT Sloan School of Management.

Eppinger, S. D. 1997. "A Planning Method for Integration of Large-Scale Engineering Systems". *International Conference on Engineering Design (ICED 97 Tampere)*. August.

Eppinger, S. D., Whitney, D. E., Smith, R. P. and Gebala, D. A. 1994. "A Model-Based Method for Organizing Tasks in Product Development". *Research in Engineering Design*, vol. 6, no. 1, pp. 1-13.

Henderson, R. and Clark, K. 1990. "Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms". *Administrative Science Quarterly*, vol. 35, no. 1, pp. 9-30.

Morelli, M. D., Eppinger, S. D. and Gulati, R. K. 1995. "Predicting Technical Communication in Product Development Organizations". *IEEE Transactions on Engineering Management*, vol. 42, no. 3, pp.215-222.

Pahl, G. and Beitz, W., (Wallace, K., ed.). 1991. *Engineering Design, A Systematic Approach*, Springer-Verlag, New York.

Pimmler, T. U. and Eppinger, S. D. 1994. "Integration Analysis of Product Decompositions". *Proceedings of the ASME Design Theory and Methodology Conference (DTM' 94)*, DE-Vol 68, pp. 343-351.

Rice, J. A. 1994. *Mathematical Statistics and Data Analysis*. Belmont, CA : Duxbury Press.

Rowles, C. M. 1999. "System Integration Analysis of a Large Commercial Aircraft Engine". *Master's Thesis in Engineering and Management*, Massachusetts Institute of Technology.

Sosa, M. E. 2000. "Analyzing the Effects of Product Architecture on Technical Communication in Product Development Organizations". *Ph.D. Thesis in Mechanical Engineering*, Massachusetts Institute of Technology.

Sosa, M. E., Eppinger, S. D. and Pich, M. 2000. "Factors that Influence Technical Communication in Distributed Product Development: An Empirical Study in the Telecommunications Industry". *Sloan Working Paper*, MIT School of Management.

Suh, N. P. 1990. *The Principles of Design*, Oxford University Press, New York.

Van den Bulte, C. and Moenaert, R. K. 1998. "The Effects of R&D Team Co-location on Communication Patterns among R&D, Marketing, and Manufacturing". *Management Science*, vol. 44, no. 11, pp. S1-S19.

Ulrich, K. T. 1995. "The Role of Product Architecture in the Manufacturing Firm". *Research Policy*, vol. 24, pp. 583-607 .

Ulrich, K. T. and Eppinger, S. D. 1995. *Product Design and Development*, McGraw Hill, New York.

