Are Your Engineers Talking to One Another When They Should?

Cost overruns, schedule slippage, and quality problems often result from a failure to provide timely information or resources. Here’s a way to help prevent that from happening.

by Manuel E. Sosa, Steven D. Eppinger, and Craig M. Rowles

Companies that design complex, highly engineered products all have their horror stories. Ford and Bridgestone Firestone lost billions of dollars after their failure to coordinate the vehicle design of the Ford Explorer with the design of its tires. Similarly, Airbus’s development of the A380 “superjumbo” suffered major delays and cost overruns because of late emerging incompatibilities in the design of the electrical harnesses of various sections of the plane’s fuselage. These mistakes probably contributed to the loss of Airbus’s CEO and to important changes in the management of the A380 program.

What’s striking about these stories and many others like them is that in virtually every case, the people involved all agreed, in hindsight, that they could have avoided their expensive mistakes by making sure that the different teams responsible for
If a company finds that a lot of planned communications were not planned, it should revisit its product development process. To help manage the communications breakdowns associated with complex projects, we propose the following approach: (a) identify unattended interfaces, areas where communication should be occurring but is not, and (b) look for unidentified interfaces, areas where communication is occurring but has not been planned. The result of implementing this approach is what we call an alignment matrix, which reveals mismatches between the communications and exchanges that are supposed to occur and those that actually do. It also demonstrates, therefore, how well the project has been planned and executed.

To see how the approach works, let’s suppose that we plan to develop a product with four components, each of which requires its own specialized design team. (This approach may be used when the organizational structure maps directly to the product architecture—that is, component X is designed by team X—which is the case in most of the complex development projects in the automobile and aerospace firms we have studied. For cases in which the organizational and product structures are not aligned, the approach may still be applicable.)

#### Catching Missed Interfaces Before They Occur

The first thing a project team does when faced with a complex development challenge is break the project down into manageable pieces that are then assigned to dedicated subteams. In the context of developing a product like a jet engine, this results in a large number of specialized cross-functional teams, each working on a different component or subsystem of the engine. Of course, these teams cannot work in isolation; in addition to designing their assigned components, they must also integrate their designs with those of the other components to ensure that the entire product or system functions as a whole. It is critical, therefore, in planning a complex product development process that the project managers specify just which resources and information different teams will need from each other at particular stages of the project.

In the Pratt & Whitney jet engine development project we studied, the engine was divided into eight subsystems, each of which was further decomposed into five to ten components, for a total of 54 engine components. Typical for such projects, the development organization was correspondingly structured into 54 cross-functional design teams responsible for each component, plus six integration teams responsible for managing engine-level requirements in areas such as fuel efficiency. These teams had to interact a lot: There were several hundred interfaces among the engine components, many of which would have experienced significant problems without proper communication among the relevant design teams.

To help manage the communications aspect of such projects, we propose the following approach: (a) identify unattended interfaces, areas where communication should be occurring but is not, and (b) look for unidentified interfaces, areas where communication is occurring but has not been planned. The result of implementing this approach is what we call an alignment matrix, which reveals mismatches between the communications and exchanges that are supposed to occur and those that actually do. It also demonstrates, therefore, how well the project has been planned and executed.

To see how the approach works, let’s suppose that we plan to develop a product with four components, each of which requires its own specialized design team. (This approach may be used when the organizational structure maps directly to the product architecture—that is, component X is designed by team X—which is the case in most of the complex development projects in the automobile and aerospace firms we have studied. For cases in which the organizational and product structures are not aligned, the approach may still be applicable.)

Companies that design complex products all have their horror stories. Yet, they can all avoid mistakes by ensuring that the different teams responsible for developing the components of the products communicate more effectively.

A new application of an established project management tool, the design structure matrix (DSM), can help a company identify where failures in planned communications could occur as well as recognize when project teams engage in unplanned technical communications. We also analyze communications between project teams that take place both within and outside the formal project structure. We conclude by discussing how managers should handle communications problems that are revealed in the process. While we do not pretend to offer a definitive solution to the design communication problem, we do believe that managers who use our tools over succeeding generations of products will improve the quality of their development processes.

### Article at a Glance

<table>
<thead>
<tr>
<th>Company</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pratt &amp; Whitney</td>
<td>Helped with the development of the PW4098, which powers the Boeing 777.</td>
</tr>
<tr>
<td>Insead</td>
<td>In Fontainebleau, France.</td>
</tr>
<tr>
<td>MIT Sloan School of Management</td>
<td>In Cambridge, Massachusetts.</td>
</tr>
<tr>
<td>Emegear</td>
<td>Medical device company based in East Hartford, Connecticut.</td>
</tr>
</tbody>
</table>

Manuel E. Sosa is an assistant professor of technology and operations management at Insead in Fontainebleau, France. Steven D. Eppinger is a professor of management science and engineering systems and deputy dean at MIT’s Sloan School of Management in Cambridge, Massachusetts. Craig M. Rowles was a Module Center manager at Pratt & Whitney, based in East Hartford, Connecticut; he is now CEO of Emegear, a medical device company in Carpinteria, California.
Assessing the Fit Between Design and Communication

The first step in analyzing the communication problem among design engineers is to have the system architects identify technical interfaces between components and record their responses on a design interface matrix. Next, have component and subsystem design team members identify the technical interactions they have had, are having, or expect to have with other teams, and present those responses on a team interaction matrix. Then merge the two matrices to generate an alignment matrix that reveals matches and mismatches between interfaces and interactions. The matrices below are based on a product with four components.

In the design interface matrix below, a shaded cell indicates the presence of a technical interface between two components. Thus, the first row of the matrix indicates that component A will require some input from components B and D but nothing from C. The first column shows that A will be expected to provide input to (or impose technical constraints on) B and D but not C.

In the team interaction matrix, each row indicates from which other teams a particular team expects to need information and resources, and each column shows where a team will be expected to provide the information and resources. A shaded cell indicates where the teams expect to interact.

1. Interview the system architects. We begin by requiring the senior engineers responsible for the product’s overall function and layout – the system architects in engineering lingo – to identify the technical design interfaces among the four components. Do components need to be spatially connected with each other? Do some components transfer forces, material, energy, or information to other components to enable them to work properly? Answers to such questions are used to identify the interfaces among all the components of the product.

Armed with this data, the project managers can present the responses on a four-by-four design interface matrix (a type of DSM used to map the network of component interfaces), such as that illustrated in the exhibit “Assessing the Fit Between Design and Communication.”

2. Survey the component design teams. In the second step, we identify the technical communications that the people working on each component design team expect to take place with people from the other teams. Specifically, we ask members of each team whether they anticipate the need for technical information or resources from other teams. In surveying them, we need to make sure they are all familiar with the function and specifications of the component or components they are developing. (We do not share with them the matrix produced in the first step, as this would bias the teams’ responses.) Using these survey data, we can represent the technical interaction patterns of the teams in another four-by-four matrix (corresponding to the
four design teams) that we call a team interaction matrix, also shown in the exhibit “Assessing the Fit Between Design and Communication.”

3. Combine the results. In the third and final step, we overlay the two matrices to obtain the alignment matrix (again, see the exhibit “Assessing the Fit Between Design and Communication”). This matrix shows the matches and mismatches between the product’s

state where they actually received the information and resources they needed. Overlaying the new team interaction matrix on the original design interface matrix would reveal whether the mismatches uncovered at the start of the project had persisted and whether other mismatches had appeared. A postmortem of this kind yields valuable insights into future product development projects, especially for companies that expect to develop similar products in the future or further generations of the same product.

We conducted one such analysis of Pratt & Whitney’s development of the PW4098, the engine that, at the time, set new standards in the aviation industry for development speed and cost. (It was also the first commercial jet engine ever certified for 180-minute extended-range twin operations from its first day of service.) We began by interviewing the engine’s architects, who identified 569 interfaces among the 54 main components of the engine. Many of these interfaces were critical and complex because they not only involved physically adjacent components but also the transfer of material (air, fuel, or oil), energy (vibration or heat), structural forces, or signals used by the control system of the engine. The design interface matrix drawn from this information is shown in the exhibit “Creating an Alignment Matrix for Pratt & Whitney.”

We then asked at least two members of each of the 54 teams involved in the project how often they received technical information from other teams during the detailed design phase of the project and how critical they perceived this information to be. The results of this survey documented 423 interactions among component teams, which appear on the team interaction matrix shown in the exhibit. We finally computed an alignment matrix by merging the two first matrices.

Readers armed with a magnifying glass would count 220 unattended design interfaces that were not matched by team interactions and 74 unidentified interfaces in which teams exchanged technical information even though there was no identified design interface between the components. Although it would be naive to expect a perfect alignment of design interfaces and team interactions – and, in this case, many of the 220 unattended interfaces were unproblematic or not critical – misalignment on this scale indicates that Pratt & Whitney was subject to considerable risks involving cost overruns or other problems with this project.

Why Mismatches Occur

Mismatches do not occur randomly in a product or organization. Rather, they are the result of product design and organizational factors. Planned key communication points may remain problematic for several reasons, including the presence of organizational boundaries (cross-boundary interfaces are more likely to be missed than interfaces with a team belonging to the same group), the lack of interface criticality (complex and critical interfaces receive more attention than noncritical ones), the use of indirect communica-
Creating an Alignment Matrix for Pratt & Whitney

This exhibit shows the design interface, team interaction, and alignment matrices we developed for Pratt & Whitney after the PW4098 project. It is striking how many mismatches (294 in all) turned up even in a post hoc analysis. Conducting this exercise before or during the project would probably have revealed many additional mismatches that got fixed during the course of the project. Note that components (in the design interface matrix) and teams (in the team interaction matrix) that belong to the same subsystem are clustered together so that we can easily distinguish (with the boxes along the diagonal) between interfaces (and interactions) that occur within boundaries versus those that fall across boundaries.

Design Interface Matrix
System engineers identified 569 design interfaces among engine components

Team Interaction Matrix
Design teams reported the occurrence of 423 technical interactions among them

Alignment Matrix Key
- Matched interface: (349 instances) design interface that is matched by an actual team interaction
- Unattended interface: (220 instances) design interface identified by system architects that lacks corresponding team interaction
- Unidentified interface: (74 instances) team interaction that takes place without a corresponding design interface identified by system architects
- Lack of interdependence: (2,219 instances) components that do not share an interface or involve design team interactions


The impact of these unattended interfaces probably resulted in very small reductions in performance or durability of affected components and systems. But given the 25- to 30-year life expectancy of a product like the PW4098, even small performance deviations could add up and cause significant warranty or service expense over the life of the product. For example, if a critical component like a turbine airfoil misses its
life expectancy by only a few percent, it could mean one or more unplanned engine removals for maintenance over the life of the engine. For such a product, a single unplanned engine removal could add up to a $150,000 incremental cost to the customer.

Although most unattended interfaces are not critical, some are. Those critical unattended interfaces mostly occur when the teams involved come from different parts of the organization. The costs can be substantial. In the PW4098 project, two unattended interfaces turned out to be critical, and their costs varied with the time it took for problems associated with the interfaces to be uncovered and resolved. One related to a change in the structural loads transferred between rotating hardware assemblies in the high-pressure core and resulted in excessive loads on coupling hardware during tests of early development engines. Consequently, Pratt & Whitney had to disassemble, redesign, and rebuild the test engines. We estimate that this one problem added 1% to 2% to the cost of the program and a three- to four-month delay to certain elements of the program. The other unattended interface was uncovered later, after early production assembly had begun but before any engines were shipped. This one, also related to loads transferred between engine modules, reduced the life expectancy of one of the main engine bearings. The number of affected parts and engines was significantly greater than those associated with the first problem — and so was the impact on the program in terms of cost and time.

We have found unidentified interfaces to be less common than unattended interfaces. However, unlike unattended interfaces, their occurrence is almost always positive for the project because they reveal potentially critical but unanticipated interdependencies among a product's parts or systems. Many of the unidentified interfaces we uncovered at Pratt & Whitney related to investigations into possible engine-
level design problems that could have resulted in excessive strain, overheating, or insufficient pressure in the test engine. Because the teams involved talked to each other as they began to see or anticipate the unexpected problems, they were able to mitigate them before the product testing phase of the project, resulting in considerable time and cost savings potential. When unidentified interfaces like these are uncovered, the main question is whether to formally incorporate them into a project's schedule and routines or leave them be. This decision depends largely on how critical the communication is. In the case of the interfaces described above, Pratt & Whitney formalized some of the relevant team interactions in planning the development of its next-generation engine.

The conditions that generate unidentified interfaces are different from those that cause people to overlook interfaces. Several of Pratt & Whitney's unidentified interfaces occurred between teams working on engine-level design scenarios that created adverse structural or thermal loads. This, in turn, generated the need for technical solutions across distinct components. In one case, three teams from both the high-pressure turbine and the low-pressure turbine had to interact with teams working on the combustion chamber to optimize the thermal environment and resulting durability of their respective components. These were deemed critical interfaces that had not been identified by the system architects. Fortunately, the people on these teams had worked together in the same roles in previous projects, making it more likely that they would have unplanned exchanges of information.

How to Manage Mismatches

Once the root causes of mismatches are understood, an organization can then decide how to deal with them. Potential solutions can be varied, including redrawing organizational lines, reassigning or creating new interface management responsibilities and facilitation tools, or even redesigning the system architecture (some of these are discussed below). To find the solution appropriate for your project, consider the following three steps:

1. Review organizational and system boundaries. For projects in which a significant number of unattended interfaces span organizational boundaries,
be missed, and the workload associated with fewer direct and indirect interfaces was more predictable. Too much modularity, though, can lead to myopia, particularly at the subcomponent level. At Pratt & Whitney we found that the design teams of highly modularized subsystems were less likely to talk to teams working on other modularized subsystems than were teams working on more integrated subsystems. In going modular, therefore, product designers still need to pay careful attention to critical interfaces across those subsystems.

2. Form teams to handle mismanaged interfaces. Managers also have the option to manage critical interfaces— to ensure that unidentified ones occur or that unattended ones are formalized—by assigning such work to the teams already tasked with the interaction or by making people on the involved teams formally and actively accountable for the interfaces. We would recommend this approach for managing identified interfaces across boundaries—the interfaces design teams are most likely to ignore.

Another way to handle missed interfaces—one that also avoids the need to significantly change the organizational structure—is to extend the responsibility of existing integration teams. Most large projects have such teams: At Pratt & Whitney there were six teams managing system issues like air and fuel efficiency, which affected the design of practically all engine components. Even though the management of team communications is not usually the primary function of integration teams, by the nature of their work, these teams communicate with almost all other teams in the organization. Accordingly, they are in a position to learn in real time about the status of critical interfaces during the design process and to bring unconnected teams together to handle critical interfaces that need special attention. These integration teams could be made responsible for flagging those critical interfaces that are not being properly attended to. In the PW4098 project, the secondary airflow team (one of the six integration teams) was responsible for managing the engine's multiple internal thermal and pressure management systems to optimize engine aerodynamic performance and component durability. It regularly set up meetings and other communications between otherwise unconnected teams to address critical interfaces.

Unfortunately, most integration teams focus on milestone planning and resource allocation, paying only marginal attention to the quality of communication between component teams. That needs to change. Consider the pain that could have been avoided in the last phases of the development of the Airbus A380 if one of the integration teams had realized that the electrical harnesses team in Germany and its counterpart team in France, which were responsible for different sections of the fuselage, were not properly communicating about their design interface specifications.

3. Select appropriate communication support tools. Many design teams miss interfaces because project planners haven’t thought through their use of communication tools and shared platforms.

Many design teams miss interfaces because project planners haven’t thought through their use of communication tools and shared platforms. To complete work, the various tools and platforms must be carefully integrated. In planning the development of the engine project that followed the PW4098, for example, Pratt & Whitney linked full engine aerodynamics and secondary airflow analytical models with component models to help the design teams manage their interfaces with the support of the integration teams. Specific people on each team were then tasked with tracking the impact of design changes across the component and system models and communicating those findings to their counterparts on other teams.

Our approach provides a systematic method for an organization to learn how and where it is exposed to the risk of communication failures between design teams working together to develop complex products. Moreover, an organization can use our tools to determine how changes in system architecture, or the emergence and removal of interfaces between system components, will affect its ability to avoid communication failures in the future. By using DSMs to document the architecture of the product for every generation of a product family, managers can identify key differences between old and new architectures. With the alignment matrix, managers create a compact and visual representation of interfaces and interactions that allows them to diagnose how their organization addresses design interfaces. Most important, the alignment matrix can help managers properly direct their efforts to align team interactions with design interfaces to prevent costly problems from occurring later in the product life cycle.

Reprint R0711J
To order, see page 155.

Harvard Business Review and Harvard Business Publishing Newsletter content on EBSCOhost is licensed for the private individual use of authorized EBSCOhost users. It is not intended for use as assigned course material in academic institutions nor as corporate learning or training materials in businesses. Academic licensees may not use this content in electronic reserves, electronic course packs, persistent linking from syllabi or by any other means of incorporating the content into course resources. Business licensees may not host this content on learning management systems or use persistent linking or other means to incorporate the content into learning management systems. Harvard Business Publishing will be pleased to grant permission to make this content available through such means. For rates and permission, contact permissions@harvardbusiness.org.