

Developing a product and supporting it after it's built is a complex endeavor involving many professional disciplines. The cost and risk associated with complex product development can be minimized—if and only if—all of the disciplines in a product's life cycle are able to work together effectively.

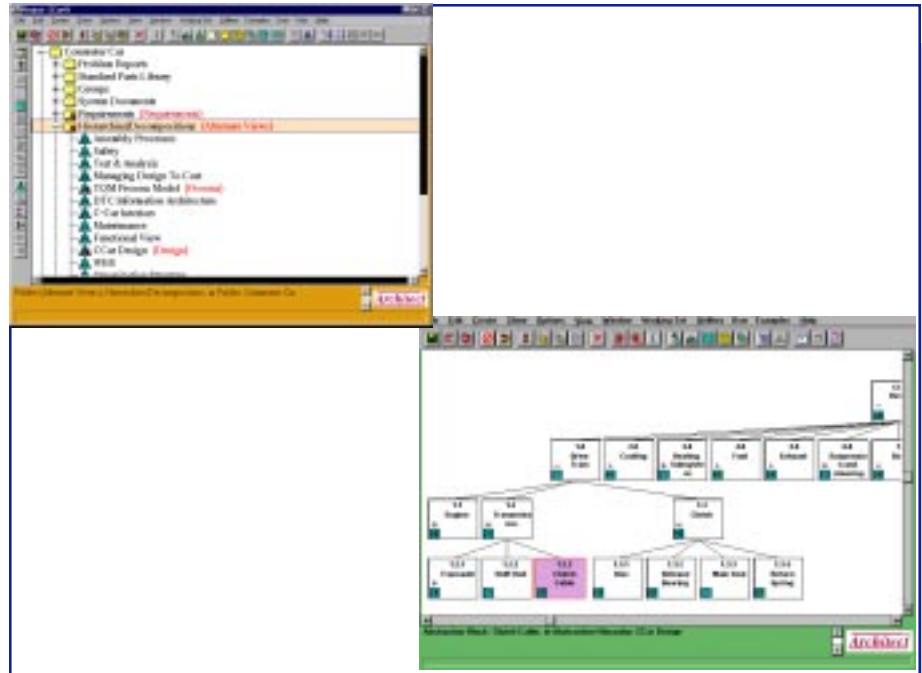
Systems engineering provides today's enterprises with a holistic approach they can adopt to master the intricacies of product development.

To apply systems engineering in product-related setting, SDRC offers a web-centric solution called the System Level Automation Tool for Engineers (SLATE™).

## The SLATE Approach

SLATE is a groupware solution that provides a first-of-its-kind systems engineering approach to product development:

- A series of graphical building blocks allow product design teams to quickly describe and build complex products on a systems basis. By taking advantage of SLATE's optional Metaphase integration, companies can capture and store these building blocks and their relationships for future use in a PDM environment.
- Methodology-independent groupware supports a team approach to product design. Widely dispersed members of project teams are able to collaborate—even when these teams include external suppliers, contractors, subcontractors, and customers.
- Automated document-generation allows companies to capture, manage, and reuse documents that are a living by-product of their design process.
- The ability to draw multiple perspectives into take-to-market life cycles enables companies to minimize product cost. By managing various forms of



SLATE's web interface lets companies organize the life cycle for a given product into a "system" of interrelated documents, market requirements, and viewing perspectives.

their intellectual property, companies can control the processes associated with designing and testing their products' individual parts and components—as well as delivering products in their fullest form.

The end result is a systems engineering environment that not only captures information about a company's products, but manages customer requirements and links these requirements to design alternatives.

## SLATE Product Line

The SLATE product line is broken into the following features sets, with each product accessing the same database with different capabilities.

**SLATE Architect** - is the full featured SLATE system and contains all system modeling, requirements engineering and document management functions in a client/server, single or multiple user environment. Architect is typically used by system architects who build conceptual models, work trade studies, analyze performance parameters and develop early system specifications. In addition, the Activator Authoring, process customization, flow and data diagramming, process modeling, Technical Performance Management and Report Generation are available in SLATE Architect.

**SLATE REquire** - is a Requirements Engineering and Management system. Although the SLATE Architect was built to integrate all requirements into your architectural design process, REquire enables you to make architectural trade-offs in the context of customer requirements by including Document Management, Requirements Engineering, Requirements Management, Traceability, Report Generation, Activator Execution (not Authoring) and much more. Since REquire is built on top of the same database as Architect, it can grow with you as your culture supports tying requirements into your product architectures.

**SLATE reVIEWer** - enables local or remote viewing & annotation of a project database (using Notes), and Report Generation. This product is used by personnel that perform quality checks, red line reviews, specification reviews, design reviews and other technical or managerial reviews of the product design. The reVIEWers are able to attach action items, suggestions, questions, decisions, answers and so forth as Note objects to any design object in the project database. Modification of any design element is not permitted from this product. reVIEWer is often used by prime contractors and subcontractors to review common design criteria or designs without allowing access to the data for modification.

**SLATE VIEWer** - enables local or remote viewing of SLATE databases across a network. No editing capability is enabled, only the ability to browse and generate reports. This product is used primarily by prime and sub-contractors to enable remote sites to see design data under configuration control, or that is not desired to be changed by the viewers. Quite often this product is used to give high-level organization managers access to generate daily/weekly reports or metric analysis reports for a program.

## Working With SLATE

SLATE provides a graphical user interface that enables system architects to design products around building blocks they already understand and combine these elements into an overall product.

Users employ graphical icons to represent components, products, customer requirements, budgets, project notes, documents, hierarchical relationships, and other product-related entities. In turn, users can build higher level “systems” by attaching or linking these icons to each other.

Users can classify SLATE building blocks by assigning “attributes” to given icons or sub-typing these graphical elements. Users can describe the behavior of a building block by attaching it to a simulation model and appropriate activator. Users can capture and enforce life cycle processes (such as the requirements management process).

## SLATE in Action

SLATE’s systems approach to product development can be applied in many areas.

**Requirements Definition.** Certain market segments—like the defense and aerospace industry—have long product life cycles that demand product requirements be aligned with continually evolving product definitions. Companies in these industries use SLATE’s groupware capabilities to capture customer requirements—and relate these requirements to system functions, work tasks (wbs elements), and other system breakdowns including logistics, training, materials, and base support systems.

**Product Design.** Companies that provide discretely manufactured products—such as automotive companies—can use SLATE’s abstract building blocks to generate product structures (e.g., for cars or trucks) comprised of assembled parts and components. The vehicle’s parts are hooked to each other, and customer requirements are linked to various parts and assemblies. Budgets can be attached to specific components

to track costs, performance metrics, physical attributes and other critical design parameters. Designers can link project notes to documented decisions, supplier concerns, and other issues.

**Test Facilities.** Companies can use SLATE’s activator feature to link requirements and test cases, and verify that validation tests do in fact meet the product’s customer requirements. SLATE monitors and reports on test results, providing complete traceability and audit trails on each requirement.

SLATE automatically generates test documents and specifications to describe conducted tests and their results. SLATE enables companies to enforce a test methodology (e.g., if customer requirements change, SLATE ensures that a test is rerun to verify that the product still is compliant).

**Proposal Generation.** SLATE enables companies to capture customer needs (by extracting requirements from RFPs) and relate these needs to a series of building blocks that can be turned into a response document. Proposal response teams can attach project notes to SLATE icons to describe the team’s approach for meeting customer requirements.

**Project Management.** SLATE enables companies to ensure that a finished product meets customer expectations (as defined by the product’s requirements). SLATE flags design items that are at risk. It enables system architects to attach budgets to a functional product structure. In turn, companies can measure and track cost, time schedules, and project resources. SLATE captures industry standards and relates them to functions performed by new products. Product functions can be mapped to physical components.

SLATE also captures project organizations (reflecting given individuals and their skills) and relate these units to specific responsibilities. This allows managers to determine who is designing specific components and how these components are driven by specific requirements.

## Business Advantages

### *Determines Design Impacts*

- Enables companies to understand the impact of design decisions before products reach the market.
- Catching product problems during the design process costs 10 to 100 times less than catching them on the factory floor.

### *Facilitates Team-based Product Design*

- Enables project teams to communicate effectively.
- Ensures that everyone in the take-to-market process works from the same assumptions.

### *Builds Lessons Learned*

- Captures the design lessons learned by a company's project teams.
- Can be leveraged to transform this knowledge into reusable best practices.

### *Reduces Risk in Product Development*

- Enables participants in the early stages of product design to coordinate their work with participants in the later stages.
- Prevents runaway projects that otherwise result from numerous start-overs, constant redesign, and substantive rework.
- Enables companies to catch bad design decisions up front.
- Avoids endless workarounds and/or expensive redesign.

## State-of-the-Market Features

### *Systems Engineering for the Entire Enterprise*

- SLATE reflects and “understands” the systems design process.
- Companies can deploy SLATE with little or no customization, thereby allowing everyone in the enterprise to work from the same design database.
- SLATE uses building blocks specifically designed for systems engineering by systems engineers.
- SLATE's groupware supports widely dispersed teams working at the front-end of the development cycle—where it's possible to identify and avoid major problems.

### *Integrated Requirements Management*

- SLATE delivers requirements to all project participants in real-time, enabling better decisions to be made faster.
- SLATE manages requirements by directly relating them to product design; this enables design decisions to be matched to customer needs.
- SLATE establishes interrelationships that account for market requirements, this ties design components to market requirements through the features they provide.
- SLATE can associate integrated quantifiable requirements (such as weight, power, or cost) directly with the product design. As designs change, SLATE recalculates cost or weight roll-ups.
- Simulation models can be integrated with the design to support system-level tradeoffs.

### *Documentation as By-product of Design Capture*

- SLATE generates touch-free documentation as a by-product of the design capture process.
- Documents automatically reflect design content; modifying the design changes the document and ensures up-to-date accuracy.

### *Downstream Feeds*

- SLATE can include downstream integrations to discipline-specific environments (e.g., software development, simulation).
- SLATE provides data management integrations.
- Bi-directional links support changes that ripple upstream as well as downstream; companies can make cross-discipline trade-offs.

## Architecture

SLATE is built on an open, standards-based architecture to ensure that your solutions can expand as your organization evolves and grows.

### **Open Interface Support**

X/Motif, Windows, Java, CGI

### **Open Platform Support**

Sun Solaris  
HP-UX  
Intel Windows NT  
Intel Windows 95/Windows 98

### **Open Network Support**

TCP/IP  
NetBIOS  
HTTP

### **Foreign Language Support**

Single-byte enabled

### **For More Information**

For more information, contact your local SDRC representative or call 1-800-848-7372.