

CDS COMPOSITE DESIGN SOFTWARE

MASTER INFRASTRUCTURE AND BACKEND OPERATIONS MANUAL

Website • WB • Licensing • Engineering Data • Web API • Compiled MATLAB • Git/GitLab • Release
Operations

PROCESS BASELINE 2.0

Engineering baseline: Web r18 • Model ID 2026090501 • API CDS-WEB-1.0

5 September 2026

Document control

Document	CDS Master Infrastructure and Backend Operations Manual
Release	CDS Web r18 / web-r18
Model / API	2026090501 / CDS-WEB-1.0
Process baseline	2.0
Authority	Governed Git repository and release manifests
Publication targets	GitLab, CDSComposites.com, WB, backend deployment, archive
Release date	5 September 2026

This controlled manual updates the infrastructure, licensing, backend, user, release, publication, and recovery procedures for the browser-first CDS Web r18 platform. Native MATLAB Compiler SDK/Runtime R2026a acceptance passed; production activation remains governed by the target-specific deployment checkpoints stated in this manual.

Contents

Contents	Contents continued
1. Purpose	10. Runtime acceptance gate
2. Governed command lifecycle	11. Observability and evidence
3. Authority and identity	12. Failure and recovery
4. Platform actors	13. Release and GitLab synchronization
5. Web and cloud boundary	14. Publication and rollback
6. Licensing and authorization	15. Operational runbooks
7. Engineering records and Recipes	16. Infrastructure flowchart standard
8. Backend execution architecture	17. Release inventory
9. MATLAB package boundary	18. Handoff checklist
19. Workbench execution and live synchronization	20. Process histories and response interpretation
21. Failure envelopes and progressive damage	22. Carpet plots and laminate exploration
23. Optimization setup and saved cases	24. Release acceptance and connected guidance

Four-command quick reference

Command	Controlled outcome
Model	Implement and validate a governed engineering capability.
Recipe	Freeze a dependency-complete reproducible configuration.
Release	Validate, package, commit, tag, and record an immutable identity.
Publish	Deploy that exact identity to approved user-facing targets.

CDS COMPLETE PLATFORM INFRASTRUCTURE

Web r18 • Model 2026090501 • API CDS-WEB-1.0 • Process baseline 2.0

CONTROLLED BASELINE

05 SEP 2026

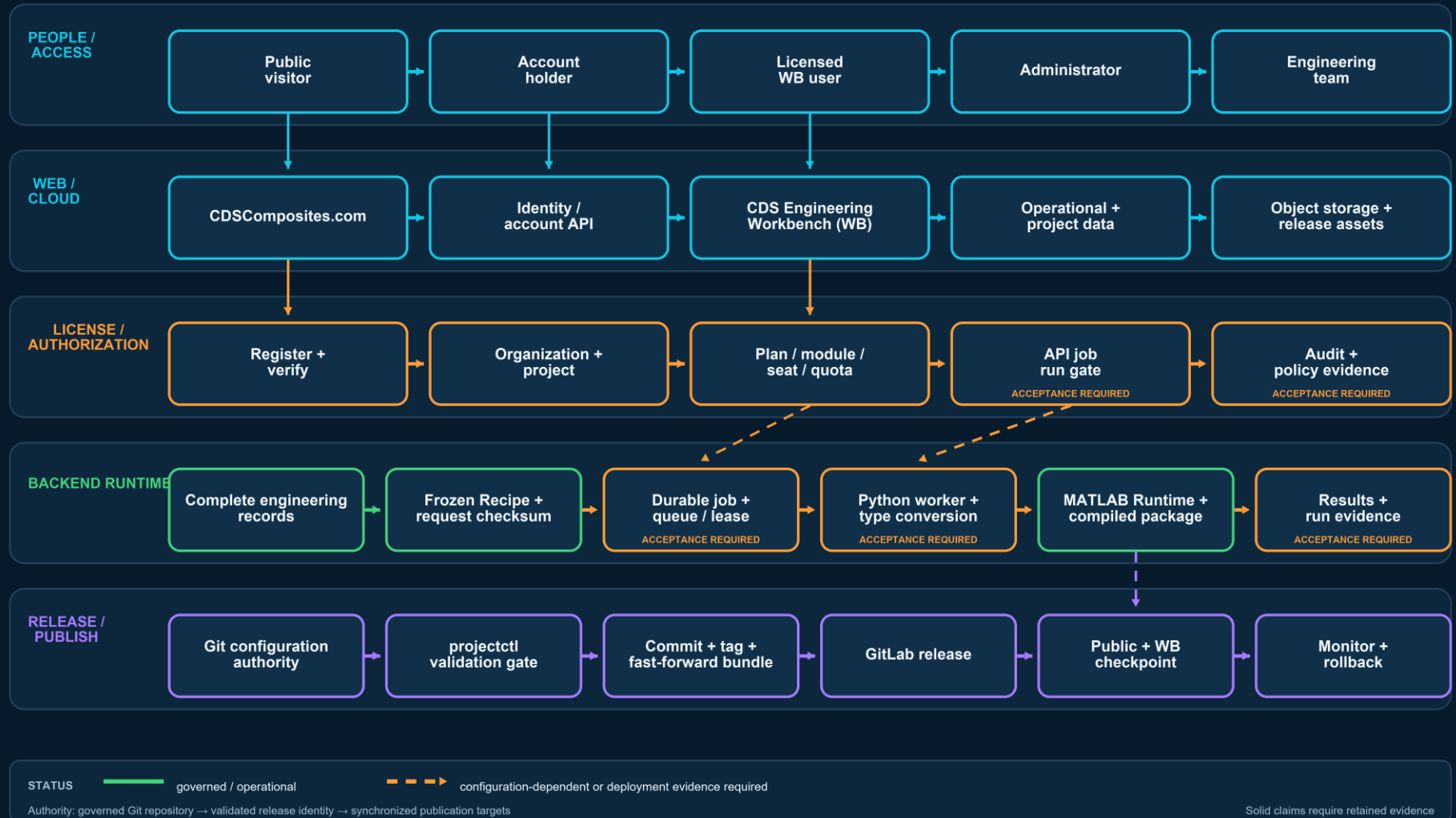


Figure 1. Complete Web r18 infrastructure baseline. Solid paths are governed or operational; dashed orange paths require configuration or target-specific deployment evidence.

01 Purpose

This manual governs the browser-first CDS platform from authenticated user access through engineering records, Recipes, queued solver execution, retained results, release control, publication, and recovery. The Git-controlled repository and its manifests remain the configuration authority. CDSComposites.com, CDS Engineering Workbench (WB), GitLab releases, download packages, and archives are synchronized publication targets.

The Web r18 release source is complete and statically validated. Native MATLAB R2026a, MATLAB Compiler SDK packaging, matching MATLAB Runtime execution, and controlled direct-versus-compiled comparison passed for the loaded r18 acceptance request recorded in the release evidence. Production activation still requires the target API, queue, worker, authorization, storage, monitoring, and rollback checkpoints described in this manual.

02 Governed command lifecycle

- Model implements or revises a governed engineering capability.
- Recipe freezes a dependency-complete, reproducible engineering configuration.
- Release validates, packages, commits, tags, and records an immutable identity.
- Publish deploys that exact identity to approved user-facing destinations.

03 Authority and identity

The release identity must agree across the MATLAB entry point, API guide, request example, database metadata, WB presentation, infrastructure map, manual, release notes, checksum evidence, Git tag, GitLab release, and public downloads.

Controlled item	Web r18 value
MATLAB entry point	CDS_USolve_Web_v2026_09_05_r18(web_request)
Request/response schema	CDS-WEB-1.0
Model ID	2026090501
Group lengths	[17,16,2,10,4,16,22,23]
Traceability results	O1-O68
Process-cycle steps	2-100
Database schema	1.1.0
Git tag	web-r18

04 Platform actors

- Public visitors browse capabilities, theory, examples, downloads, and contact routes.
- Account holders verify identity, maintain profiles, request access, and join organizations or projects.
- Licensed WB users create and edit complete engineering records, construct Recipes, submit authorized jobs, and review retained evidence.
- Administrators manage accounts, plans, module entitlements, organization access, quotas, audit events, and publication checkpoints.
- Engineering authors maintain the governed model, schema, validators, manuals, examples, and release evidence.

No browser control alone is trusted to enforce licensing. Authorization is checked by the API before a job reaches the queue and again by the worker before the compiled package runs.

05 Web and cloud boundary

CDSComposites.com owns public documentation, registration, account, support, licensing, and download experiences. WB owns engineering record editing, dependency navigation, Recipe construction, process-cycle entry, job submission, and results presentation.

Operational stores are separated by responsibility:

- Identity and operational data: accounts, sessions, organization membership, entitlements, license events, jobs, quotas, and audit events.
- Engineering project data: complete material, lamina, laminate, process, geometry, load, analysis, Recipe, and run records.
- Queue and leases: job order, worker ownership, progress, cancellation, and retry state.
- Object storage: immutable input snapshots, large result arrays, plots, reports, logs, and release artifacts.
- Static publication: public pages, theory, help, route maps, manifests, and downloads.

06 Licensing and authorization

The browser-first sequence is:

1. The user registers and verifies the CDS account.
2. Organization/project membership is established.
3. An administrator or approved commercial rule assigns plan, module, seat, quota, and validity constraints.
4. The identity service issues a short-lived authenticated session.
5. WB submits a Recipe job request.
6. The API checks account status, organization membership, module entitlement, compatible model ID/API schema, Recipe ownership, and quota.
7. The worker repeats the authorization check before invoking MATLAB Runtime.
8. The run record retains the authorization decision, immutable request checksum, model ID, Recipe ID, status, results, and audit linkage.

Signing keys, session secrets, database credentials, queue credentials, object-storage credentials, email/payment secrets, and Git credentials never enter public assets, WB client code, solver source, examples, release ZIPs, or Recipes.

07 Engineering records and Recipes

WB stores complete property records, including values that are not active in the current Recipe. Properties consumed by the active Recipe are highlighted. References use compatible selection controls and direct navigation; they are never entered as free-text identifiers. Deletion requires a dependency warning whenever downstream records would be disconnected.

Materials, laminae, laminates, process cases, structures, loads, analyses, and Recipes use stable IDs, canonical SI values, explicit units, property-level sources, foreign keys, transactions, and schema versions. NULL means unknown and is never silently converted to zero.

Process cycles accept 2-100 ordered steps containing time, temperature, pressure, and time-varying upper and lower boundary conditions. Pressure is stored and transported by Web r18 but is not yet coupled into the governing mechanical equations; responses must expose that limitation.

A runnable Recipe contains:

- Recipe identity, owner, status, compatible release, and schema.
- Dependency closure for every referenced engineering record.
- Canonical semantic inputs plus the exact resolved web request.
- Procedure and expected result summaries.
- Validation evidence and known limitations.
- File/record inventory with SHA-256 values.

SQLite .cdsdb remains the portable engineering interchange and recovery format. Multi-user operational storage may use a server database while preserving the same semantic entities and dependency rules.

08 Backend execution architecture

The production target path is:

```
WB -> authenticated API -> authorization/validation -> durable job -> queue -> Python worker ->
MATLAB Runtime -> compiled CDS package -> result/evidence storage -> WB
```

API responsibilities

- Authenticate user and organization context.
- Verify entitlement, module, ownership, compatibility, and quota.
- Freeze the Recipe and create an immutable request snapshot.
- Query project data and resolve dependency closure.
- Validate the CDS-WEB-1.0 request before queueing.
- Expose health, contract, submit, status, results, cancellation, and artifact endpoints.
- Serialize results and return stable job/status identifiers.

Worker responsibilities

- Claim one leased job and repeat the run authorization check.
- Initialize MATLAB Runtime and the compiled package once per worker process.
- Convert JSON-safe values to MATLAB numeric arrays and structures.
- Run one solver job at a time per package handle.
- Convert web_response to JSON-safe named families and O1-O68.
- Persist immutable inputs, results, diagnostics, timing, checksums, and artifacts.
- Release or fail the lease cleanly and recycle unhealthy workers.

Job states

Queued -> Running -> Completed | Failed | Cancelled

Retries create explicit attempt records. A retry never overwrites a completed or failed attempt. Cancellation is cooperative: the API records intent, the queue/worker observes it, and the final state explains whether execution stopped before or after solver invocation.

09 MATLAB package boundary

CDS_USolve_Web_v2026_09_05_r18 accepts one web_request and returns one web_response. The compiled package owns the governed equations and numerical response. The Python service owns authentication, database queries, Recipe resolution, queuing, type conversion, serialization, evidence persistence, and HTTP behavior.

web_response.outputs contains O1-O68. Named result families include primary, laminate, micromechanics, transport, Double-Double, structural, discontinuous-fiber, and diagnostics results.

Headless execution keeps plotting, animation, file, report, and verbosity controls disabled by default. Large histories and artifacts are persisted by the service, not emitted through console output.

10 Runtime acceptance gate

Before production numerical execution:

1. Compile the Web r18 entry point with the approved MATLAB release and MATLAB Compiler SDK.
2. Install the exactly matching MATLAB Runtime release/update on the worker host.
3. Initialize the generated Python package once and retain the returned handle.
4. Execute controlled plate, cylinder, beam, process, and aligned-discontinuous Recipes.
5. Compare direct MATLAB, compiled package, worker, API, and WB results using named outputs and O1-O68.
6. Record tolerances, environment, package/runtime versions, timing, checksums, and reviewer approval.
7. Enable production workers only after every required comparison passes.

Static checks are necessary but are not numerical runtime acceptance.

11 Observability and evidence

Every job records:

- User, organization, project, Recipe, model ID, API schema, worker, and attempt.
- Authorization decision and applicable module/quota policy.
- Immutable request and dependency checksums.
- Queue, start, completion, and cancellation timestamps.
- Worker/package/runtime versions and health.
- Named/O1-O68 response checksums.
- Warnings, limitations, structured errors, logs, and artifact links.

Health is separated into API, database, queue, object storage, worker, MATLAB Runtime, compiled package, and end-to-end reference-Recipe checks. A healthy API does not prove a healthy numerical backend.

12 Failure and recovery

- Validation failures stop before queueing and return field-level diagnostics.
- Authorization failures create an audit event but never create a solver job.
- Queue/lease failures preserve the immutable job and may be retried under policy.
- Worker or runtime failures retain stdout/stderr, structured diagnostics, attempt state, and host/package identity.
- Partial numerical results are never presented as completed engineering results.
- Object-storage failures prevent completion until result evidence is durably persisted.
- Recovery begins from the Git/release identity, then databases and object storage, then API/queue/workers, then WB and public publication targets.

13 Release and GitLab synchronization

The r18 release retains the official r17 lineage and reconciled local corrections without rewriting history. Validate the candidate, create a new annotated web-r18 tag, push main and the tag to the official GitLab repository, and verify both remote references. Never replace an existing release tag.

The formal GitLab release must point to the exact web-r18 tag and use the same package/checksums exposed by the public CDS download page.

14 Publication and rollback

Publication targets are GitLab, CDSComposites.com, WB, backend deployment, and archive/backup. Each target retains a rollback reference. Failed verification stops distribution and restores the prior healthy checkpoint without rewriting Git history or release evidence.

15 Operational runbooks

New licensed WB user

- Verify the account and organization membership.
- Assign plan, modules, validity, seat, and quota.
- Confirm WB access and API authorization.
- Submit a non-production validation Recipe.
- Verify the run/audit record and results access.

Backend deployment

- Verify the web-r18 tag, release manifest, package checksum, and deployment environment.
- Install matching MATLAB Runtime and generated Python package.
- Apply database migrations and verify queue/object-storage connectivity.
- Start one worker, run health and reference-Recipe acceptance, then scale.
- Confirm authorization, cancellation, failure, retry, result, and audit paths.
- Retain prior deployment and database rollback checkpoints.

Periodic checks

- Daily: public site, authentication, API, queue depth, worker health, failures, and backups.
- Weekly: download hashes, reference-Recipe execution, failed notifications, audit review, and restore sample.
- Per release: full validation, runtime parity, manifest/hash, deployment, and archive checks.
- Quarterly: key/secret inventory, access review, provider/dependency review, and disaster-recovery exercise.

16 Infrastructure flowchart standard

The controlled baseline is a 36 x 24-inch landscape editable SVG with print-ready one-page PDF and 7200 x 4800 PNG exports. The lanes are People/Access, Web/Cloud, License/Authorization, Backend Runtime, and Release/Publish. Cyan represents access/web, orange authorization, green engineering runtime, purple release/publication, and dark

navy the system boundary. Solid paths are governed/operational. Dashed orange paths are configuration-dependent, not accepted, or awaiting target-specific deployment evidence.

17 Release inventory

- Web r18 MATLAB master and validator.
- Web API/compilation guide and structured request example.
- Backend infrastructure and user manual source, DOCX, and PDF.
- Editable infrastructure SVG, print PDF, and 7200 x 4800 PNG.
- Twenty-four schema-1.1.0 sample databases and manifest.
- 160-record WB baseline material library.
- Project/release manifests, checksums, release notes, validation evidence, Git bundle, and local merge/push script.

18 Handoff checklist

- ☐ Repository and release identity agree.
- ☐ Web-only request/response contract and database schema are current.
- ☐ Manual and infrastructure diagram match the actual backend status.
- ☐ Static/database/archive gates pass.
- ☐ Native MATLAB Compiler SDK runtime acceptance is explicitly recorded.
- ☐ Git bundle base and head are verified.
- ☐ Local merge/push script requires a clean worktree and fast-forward.
- ☐ GitLab tag/release and public downloads reference the same bytes.
- ☐ Rollback and recovery references are retained.

19 Workbench execution and live synchronization

WB connects material and model records to solved laminae, laminate plies, geometry, process cycles and load cases. A simulation selects the records to use together. Before interpreting a result, check its simulation, laminate, load case, units and execution source. Shared references connect records; duplicating a record creates another design record and does not automatically duplicate every dependency.

The input and output selectors are independent. Live synchronization links their selection where supported. It is not a cloud backup, a saved snapshot or an automatic production solve. Save a changed record explicitly. Use Run or Solve when a retained numerical result must be recomputed.

The live mechanical CLT preview uses the currently visible ply materials, thicknesses, angles and loads. Saved thermal, cure and moisture contributions can still come from the previous run. Re-run the relevant analysis after changes to those inputs. A combined response is only as current as each contribution it contains. Never interpret a visually refreshed plot as proof that all saved process histories have been recomputed.

The browser/service TypeScript calculations and the compiled MATLAB CDS-WEB-1.0 package are separate execution paths. The compiled r18 acceptance establishes direct-versus-Runtime parity for its controlled request; it does not establish deployment behind every WB plot, nor end-to-end equivalence of all interactive WB features. The service response contract must retain its own truthful identity until an adapter and end-to-end checks establish a shared contract.

Materials and classes

Built-in classes use fixed recognizable icons: fiber, polymer, hexagonal core, metal ingot, elastomer dogbone and green composite microstructure. A new custom class can select its color and icon. A new material must also declare its material type because that determines the applicable input record fields. Class styling identifies a category; it is not an engineering property or approval status.

Material searches include classes as well as material names. The All Materials filter belongs above the laminate material column. Compatible solved laminae and direct material records remain distinct input choices. Color and the right-hand class icon help identify long dropdown entries; confirm the selected record name and property values before applying it to multiple plies.

Duplicate actions accept a count for design studies. Each duplicate must have a distinct identity. Shared referenced records remain shared unless deliberately duplicated and relinked. Property locks control editing and range behavior; they do not certify material data. Unknown values must be supplied before analyses that require them. Starter records are representative rather than certified allowables.

Laminate editing and geometry

Choose a common laminate configuration or edit individual plies. Fill-down controls use the selected row as the source for the selected column. Check the source row before filling material, angle or thickness values. Hidden plies are excluded from the current visible-ply preview; verify the displayed ply count and total thickness before interpreting it.

The live stack can color plies by material or angle. Balance and symmetry are independent properties: equal positive and negative off-axis families establish balance, while mirrored ordering about the mid-plane establishes symmetry. A Double-Double label or preset is not a guarantee that later edits preserve either property. The geometry type selector changes the relevant dimensions and retains each type's dimension records when switching back.

Laminate summary presents effective properties beside ABD stiffness and inverse-ABD compliance. Units differ between extensional, coupling and bending blocks. The compliance matrix is the inverse of the complete ABD matrix, not an elementwise reciprocal. Nonzero coupling can be physically valid for an unsymmetric layup.

20 Process histories and response interpretation

Upper and lower boundary tables define time-dependent temperature and boundary conditions. Fill-down buttons copy the selected value down the applicable column. Pressure is displayed with the boundary data for convenient editing; this arrangement must not be interpreted as creating independent pressure physics if the underlying record is shared.

Keep process times ordered and use 2 to 100 steps. Verify the temperature units, film coefficients, reference temperature, initial state and material kinetic/transport parameters. In the compiled package, pressure history is accepted and retained but is not coupled into the governing mechanical equations. The process chart must not imply that pressure automatically produces a structural membrane load.

Thermal and moisture output panels show the 2D field beside time-history and through-thickness sections. Solver-level resolution is the default; a common case is 51 time samples by 33 thickness nodes. The dimensions depend on the actual solved data and are not a universal fixed mesh. The 2D display is time versus through-thickness position, not a two-spatial-dimensional heat-transfer solution. The section plots follow the selected time and thickness position. A finer display cannot recover physics or resolution absent from the solved field.

Stress and strain plots can select response source, coordinate system and components. Maintain the control positions when changing source so that the selected contribution remains clear. Process time is a physical time for a transient process result. A progressive-failure pseudo-time is a load multiplier and is not elapsed time.

Settings can increase displayed samples per ply for smoother response plots. The compiled structural formulation retains three recovery nodes per ply. Additional display interpolation does not introduce through-thickness constitutive physics, interlaminar stresses, three-dimensional stress recovery or new failure modes.

21 Failure envelopes and progressive damage

Set the failure criterion and loading controls in Input Load. Output Response contains the resulting plots, tables and summaries. Check ply tensile, compressive and shear allowables, local/global coordinates, load signs and units before calculation. Membrane resultants expressed in N/mm and N/m differ by a factor of one thousand. Moment resultants have their own units and are not interchangeable with membrane loads.

WB provides maximum stress, maximum strain, Tsai-Hill, Tsai-Wu and Hashin checks. The criterion must match the material model and available allowables. Tsai-Wu interaction assumptions affect the envelope. A criterion label does not by itself provide calibrated interaction parameters or imply that Puck or LaRC theories are implemented. The compiled backend may expose a different supported criterion set; consult its API guide instead of inferring support from WB controls.

An envelope evaluates proportional load rays in the selected load plane. It is a first-failure boundary for that setup, not a general ultimate-strength surface or a certified allowable. A right-click marker option helps reveal the evaluated points. Increasing ray density refines the sampled boundary but does not change the underlying failure criterion or fill missing material data.

Progressive analysis activates damage modes and reduces the retained ply stiffness according to the selected degradation assumptions. Results depend on ramp components, step size, stopping rule and search limit. First-ply and last-ply load factors answer different questions. A search-limit stop is not proof of unlimited strength. Near-zero retained stiffness can make the numerical response ill-conditioned and should be reviewed rather than interpreted as a reliable post-failure design capacity.

Saved process/thermal, cure-shrinkage and moisture contributions can be included when selected and available. Confirm their source record and time. Fixed residual offsets are not generally equivalent to scaling every stress contribution by a single reserve factor. Compare cases only when their load vector, offsets, criterion, strength data and stopping assumptions are compatible.

22 Carpet plots and laminate exploration

Carpet plots sweep the magnitudes of two angle families and show selected effective laminate properties, such as E_x and E_y . Family lines distinguish which angle changes along each series. Hovering a sampled point can show a color-coded representative laminate to the right. This preview is a design explanation, not an automatically saved or applied laminate.

Check the family definitions, sweep limits and sample count. The supported angle-magnitude range is 0 to 90 degrees. Positive and negative off-axis assignments preserve the intended balanced families; zero-degree and ninety-degree cases must not be counted as distinct positive/negative orientations. Material, ply thickness and family proportions remain part of the construction assumptions. The carpet plot is not an exhaustive search over arbitrary stacking sequences, mixed materials, manufacturing constraints or failure theories.

The same point can have attractive elastic properties and unacceptable strength or process response. Apply or create a candidate deliberately, then run the applicable structural and process checks before treating it as a design result.

23 Optimization setup and saved cases

Select the structural load case at the top of optimization inputs. Choose the objective, failure criterion, required load factor, allowed angles and ply limits. Balance, symmetry and minimum angle fractions restrict the generated designs. Load-component buttons select the components to ramp; their callout lines identify directions in the laminate graphic. Confirm both the selected components and the values inherited from the load case.

Minimum-mass searches seek a feasible laminate within the defined space. Strength and strain objectives use their stated fixed-ply construction and evaluation definitions. Strain at first failure is not an independent maximum-service-strain constraint. Search settings, evaluation budget and seed affect a heuristic search. Best Found means the best evaluated candidate, not a proof of a global optimum. Only a completed exhaustive evaluation establishes coverage of its specifically enumerated space.

The current WB search is restricted to its supported material, thickness and angle construction. Do not assume it searches arbitrary mixed-material laminates, variable individual ply thicknesses, delamination, buckling, fatigue or manufacturing defects. Its mechanical objective does not automatically establish process compatibility or long-term durability.

Best Found appears beside the progress and candidate plots. Apply Candidate is disabled until a candidate is available. Updating an existing laminate changes that record and can affect downstream references. Create New preserves the original laminate and creates a separate candidate from the simulation context. Inspect the resulting simulation links before another solve.

Save Snapshot is located in optimization inputs beside the structural-case selector. It freezes the optimization setup and results for later review. The Optimizations tree entry remains below Simulations and uses the dark-green target icon. Selecting a saved case reopens its snapshot; selecting several enables a comparison table and Ashby, bar or line displays. Saved results must remain distinct from new live calculations and from the general Summary tab.

Compare load factors only in the context of each case's load vector and selected ramp components. Compare strains only for compatible component, coordinate and failure definitions. A saved run with zero feasible candidates does not establish that the design meets its constraints. Keep the objective, units, feasibility status, source simulation and search limits visible when sharing comparisons.

24 Release acceptance and connected guidance

The r18 loaded acceptance fixture explicitly enables load_definition element 16 and applies a detailed load matrix. It matches the R2026a compiled package over 61,494 numeric values with zero observed difference. A separate high-load fixture terminates after two of twenty requested process steps and retains an O57 history of two rows by 288 columns for eight plies. These are bounded acceptance cases, not certification across every material or geometry.

The inherited r17 acceptance fixture suppressed the detailed load matrix. Its unchanged historical results demonstrate parity for that request, not loaded structural response. Preserve this distinction when comparing release evidence.

Detailed linked guidance is published under Getting Started for connected WB workflows and failure/optimization, and under laminate theory for interactive WB methods. The website's What's New and homepage carpet/stress screenshots are entry points to those guides. Screenshots illustrate representative demo data and are not independent validation evidence.

Approval record

Role	Name	Decision / date	Evidence reference
Engineering			
Backend operations			
Release management			