DARWIN® 10.0 Release Notes

September 2022
Southwest Research Institute®

Summary of New Capabilities

New features and enhancements for FAA certification assessments:
- Axial Blade Slots
- Updated Titanium Hard Alpha Anomaly Distributions
- Security Features for Anomaly Distribution and POD Data

New general features:
- 3D GUI Visualization Enhancements
- SIF Solution for Semi-Elliptical Surface Crack on Hollow Cylinder
- Constraint-Loss Surface Fatigue Crack Correction Model
- Python-Based Scripting Tool

Changes in GUI menus and FE models that are supported in DARWIN:
- New GUI Menus to Support Objectives-Based Data Structure
- FE2NEU Finite Element Model Quality Checks (MQCs)

Speed and robustness improvements:
- Autoplate Node-Skipping Enhancement
- Autoplate Speed Enhancement for Large 3D FE Models
- Automodeling Speed Enhancements
- Speed Enhancements for User-Defined Cracks
- DARWIN Results File Import Speed Enhancements
- GUI Speed Enhancement for Large 3D FE Models
- Formation Module Programming Guidelines for Improved Execution Speed

General enhancements:
- Updated Limits for Stress Intensity Factor Solutions
- Deterministic Analysis Enhancement
- User-Defined Project Rules
- Setting GUI Preferences via an External File
- User's Guide Keyword Search Enhancement

Axial Blade Slots

DARWIN 10.0 includes a new capability for risk assessment of axial blade slots in accordance with upcoming FAA Advisory Circular AC33.70-5. A new analysis mode for axial blade slots has been included as a separate anomaly type as shown in Figure 1. It enables users to define an axial blade slot region using 1D stress profiles or 3D finite element (FE) model geometries. Users can place cracks directly on blade slot edges and surfaces in 3D FE models. Figure 2 shows an example blade

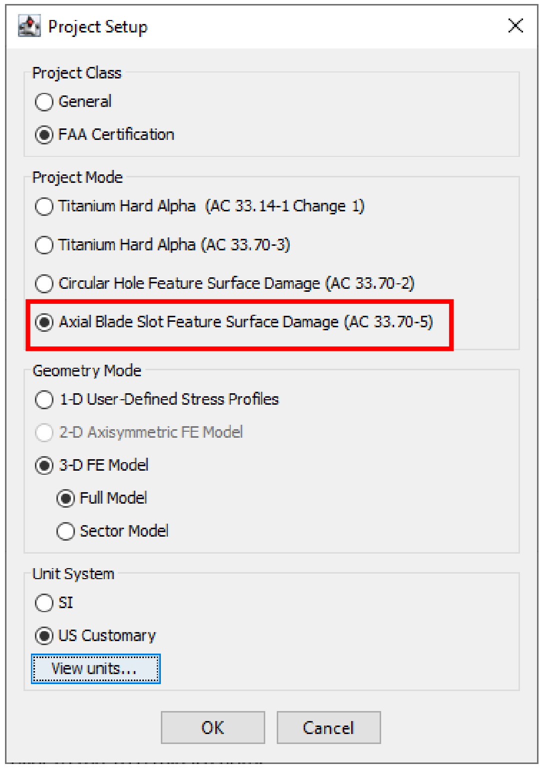


Figure 1. Enabling the new axial blade slot analysis mode in DARWIN 10.0.

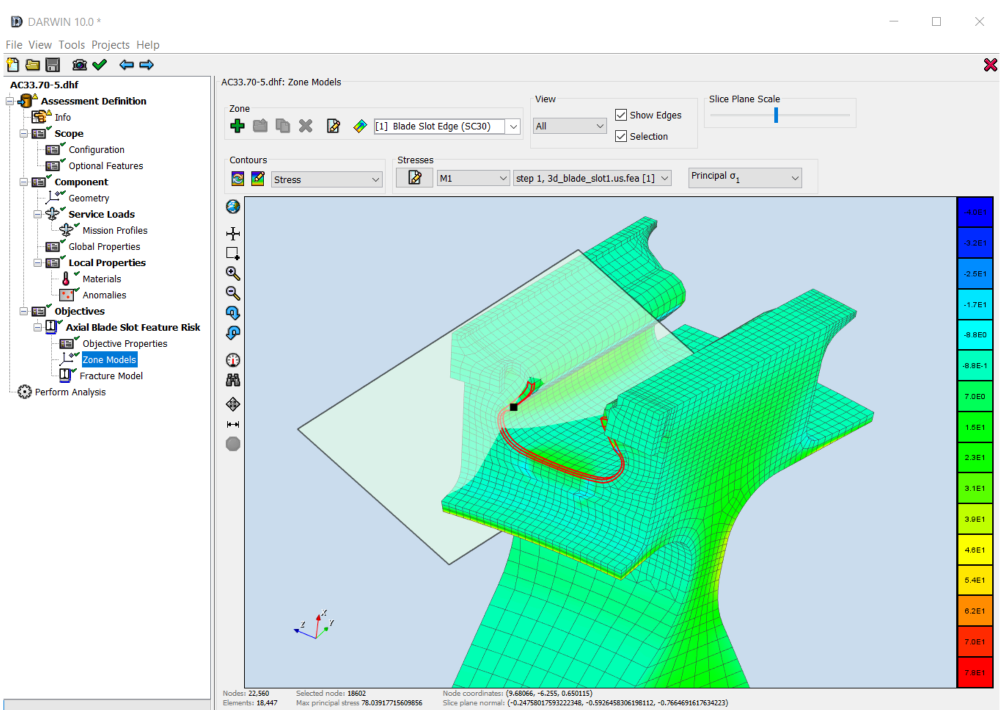


Figure 2. Defining a blade slot edge zone using the new axial blade slot capability.

slot edge region created in the GUI. This feature includes the capability to assign manufacturing process credits to axial blade slots, and provides a standardized report summarizing the results of the axial blade slot damage tolerance assessment.

Updated Titanium Hard Alpha Anomaly Distributions

DARWIN was originally developed to assess the risk of titanium hard-alpha anomalies in engine components in accordance with AC 33.14-1 requirements. In 2017, the FAA issued a revision to AC 33.14-1 (AC 33.14-1 Change 1). A new advisory circular with the designation AC 33.70-3 will be released soon. DARWIN 10.0 has been enhanced to support the requirements of both AC 33.14-1 Change 1 and the upcoming AC 33.70-3. DARWIN 10.0 uses the same analytical methods to support new/revised ACs that were used previously. However, AC 33.14-1 Change 1 and AC 33.70-3 contain new anomaly distributions that are now supported in DARWIN 10.0.

To support AC 33.14-1 Change 1, the anomaly type for certification of titanium hard-alpha anomalies under AC 33.14-1 Change 1 was renamed (as shown in Figure 3) and the revised AC 33.14-1 anomaly distributions (4 new files) were added to the DARWIN certification directory.

To support AC 33.70-3, a new anomaly type for certification of titanium hard-alpha anomalies under AC 33.70-3 certification was added to the Project Setup menu (Figure 3) and the new AC 33.70-3 anomaly distributions (6 new files) were added to the DARWIN certification directory.

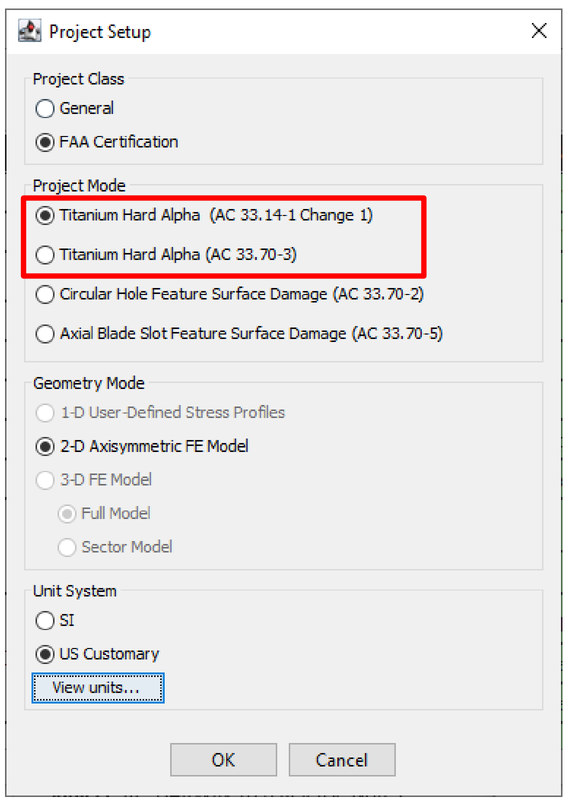


Figure 3. DARWIN 10.0 provides new/revised anomaly types for certification assessments associated with AC 33.14-1 Change 1 and AC 33.70-3.

Security Features for Anomaly Distribution and POD Data

DARWIN contains anomaly size distribution and POD curve data files that are used in FAA certification assessments. These files are located in a designated AC certification directory in DARWIN. The files cannot be edited within DARWIN, but they may be edited outside of DARWIN. In previous versions of DARWIN, the data files contained in the DARWIN AC certification directory could potentially be modified by users and then applied to FAA certification assessments without someone (the user, or a supervisor, or the FAA) knowing that the correct standard file was not used.

DARWIN has been enhanced to prevent users from modifying AC certification directory files and attempting to apply them as the correct standard files for FAA certification assessments. A special keyword (also known as a hash signature) has been added to all anomaly size distribution and POD curve data files (i.e., correct standard files) in the DARWIN AC certification directory. When the hash signature is present and the file is unmodified, DARWIN recognizes the file as a correct standard file for FAA certification assessment. If the hash signature is present but the file and/or keyword has been modified by the user, DARWIN will issue an error and will prevent the file from being imported into the GUI. If the hash signature is not present, DARWIN will recognize the file as a user-provided file and will issue a warning to the FAA certification report indicating that a user-provided file was used in the analysis.

DARWIN includes a new feature called the File Signing Tool for embedding hash signatures into data files. The capability to create hash signatures for anomaly distribution and POD curve files contained in the DARWIN AC certification directory is limited to the SwRI software development team. Access to this capability is controlled via a hidden feature keyword. However, the tool can be used to embed hash signatures in other files as well, and will be useful for protection of other features in the future.

3D GUI Visualization Enhancements

Previous versions of DARWIN include features to support 3D modeling, including the capability to visualize the interior of a 3D FE model. In DARWIN 9.2, utility regions were introduced enabling the user to view a model's interior by removing regions from the view. Users select regions with a box, by contours, or by combining multiple regions into a single region by Boolean operations. This feature enabled users to define interior property regions in 3D FE models, but was difficult to use purely for visualization.

DARWIN 10.0 includes several new 3D visualization capabilities:

- *Toggle edges:* The GUI now has the option to display or hide finite element edges. For very fine resolution models, these edges may obscure fields hidden behind them, e.g., stresses and temperatures, and users can now disable them. Figure demonstrates this capability.

- *Model clipping:* Model clipping enables users to define a plane for clipping. Regions on one side of this plane are "clipped" from the component (removed from the view) and a new interior surface appears that is defined by the clip plane. Tools within the GUI enable users to define the location of the clip plane and its angle by moving a directional arrow. This capability is available for all 3D views. Figure illustrates the interior of a 3D model by moving the clip plane through the model at a constant angle. Figure presents the interior of a 3D model at different clipping angles.

- *Iso-surfaces:* Iso-surfaces is a technique that defines surfaces based on field values. For example, an iso-surface could be defined by all stresses that are below some threshold value. DARWIN now enables users to define an iso-surface on a 3D FE model. The supported quantities are stresses, temperatures, lives, and risk values. Users define the value for the iso-surface using a slider that displays the current iso-surface value. Users may view the iso-surface above or below the defined value. Figure shows iso-surfaces at various thresholds where the threshold provides an lower bound, and Figure shows the corresponding iso-surfaces where the threshold provides an upper bound.

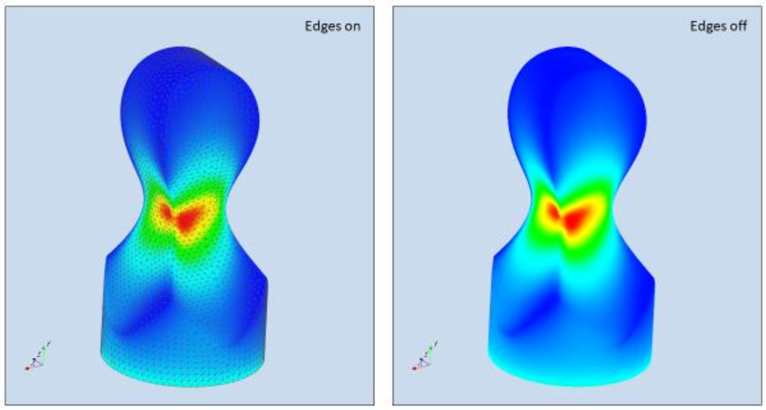


Figure 4. Illustration of Toggle Edges.

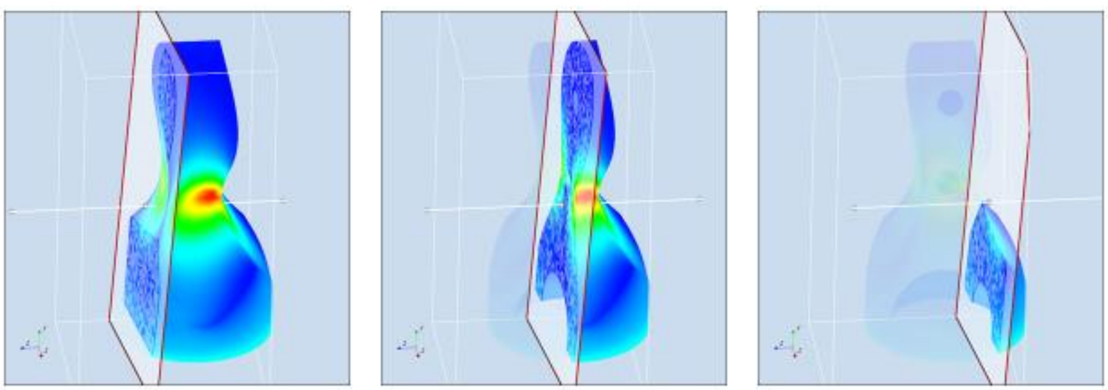


Figure 5. Illustration of Clipping at three different clip depths.

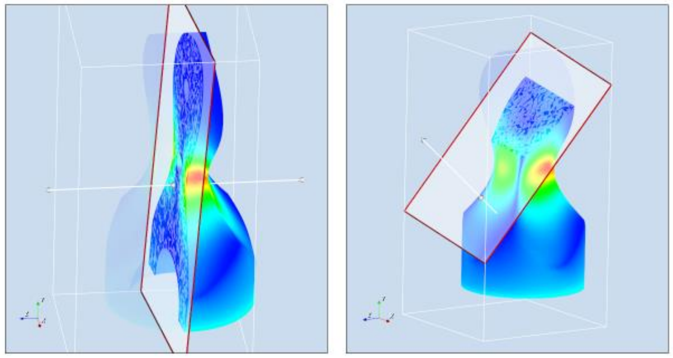


Figure 6. Illustration of Clipping at different clip angles.

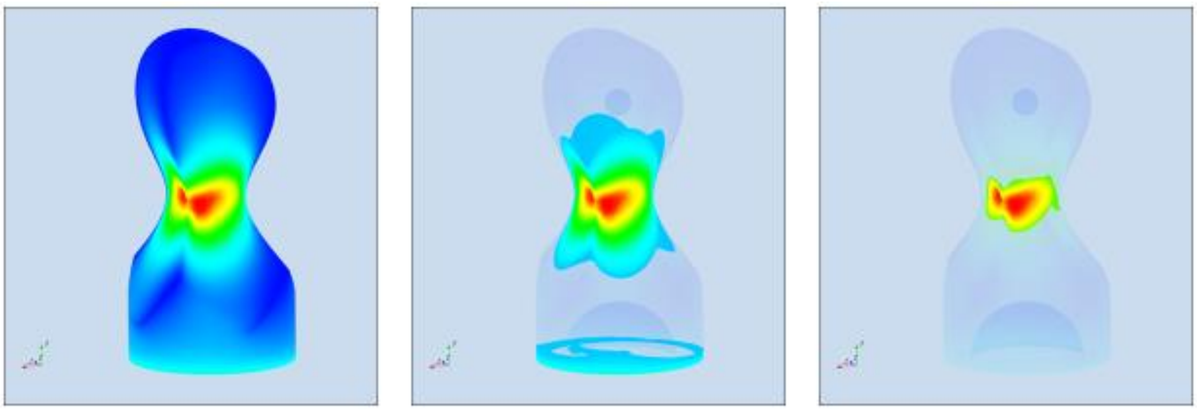


Figure 7. Illustration of Iso-surfaces removing material below a threshold.

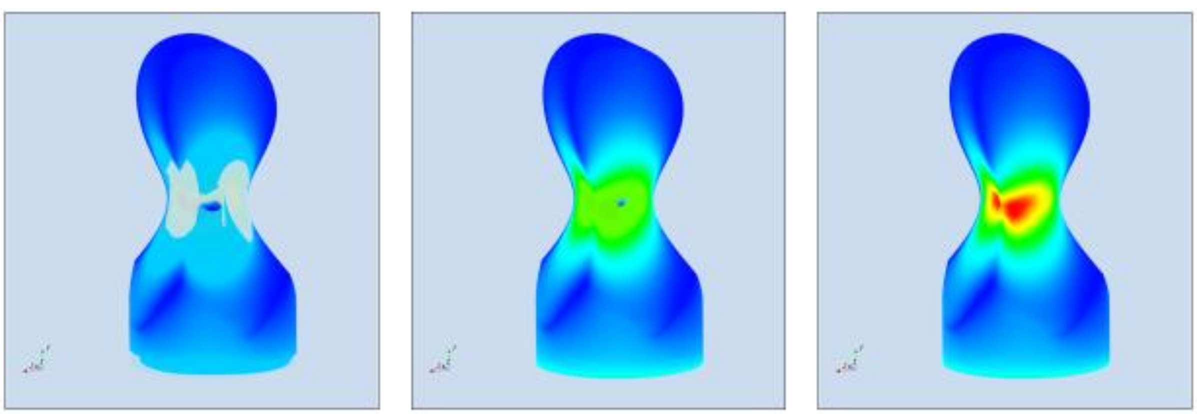


Figure 8. Illustration of Iso-surfaces removing material above a threshold.

SIF Solution for Bivariant Surface Crack on Hollow Cylinder

DARWIN 10.0 includes a new bivariant weight function (WF) stress-intensity factor (SIF) solution for an external surface crack on a hollow cylinder. The geometric parameters of this new crack model, entitled SC34, are shown in Figure 9.

The SC34 fracture plate consists of two concentric circles with the initial crack located on the outer circle. The circles are placed within the boundaries of the 3D FE model. The GUI enables the user to adjust the diameters of the two circles independently via the mouse. The SC34 crack model is available for 3D FE models (both full and sector) in both life and risk assessment analysis modes.

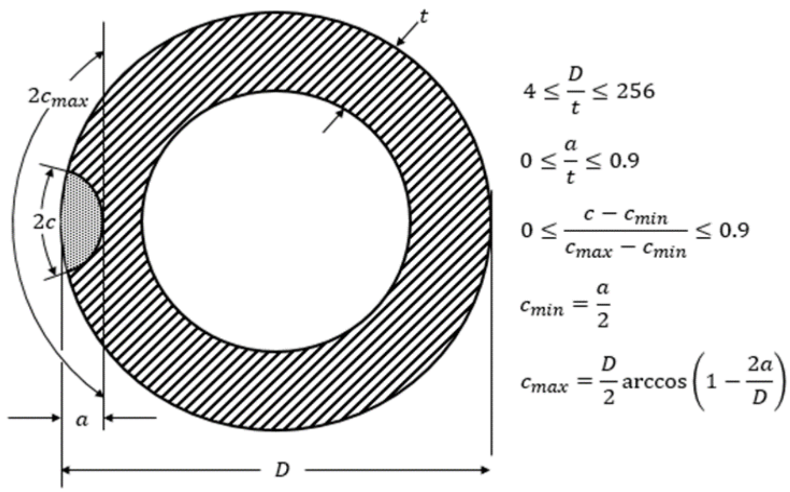


Figure 9. Geometric parameters for SC34 (bivariant surface crack on hollow cylinder).

Constraint-Loss Surface Fatigue Crack Correction Model

Changes in the stress intensity factor along a crack front can influence crack growth behavior. To account for the impact of surface effects on crack growth, correction factors are often applied to surface-breaking tips. In previous versions of DARWIN, a Beta Correction Factor (BCF) was applied to the stress intensity factor (SIF) range for the tips of part-through cracks where the crack intersected the surface (surface-breaking tips). The BCF is identical to the method used in NASGRO and was originally based on research by Newman and Raju. The BCF generally reduces the value of the SIF about 10%, although there is also a dependence on the R-ratio (at higher R values, the BCF is larger and the reduction in SIF is slightly smaller).

An alternative surface correction factor method referred to as the Constraint-Loss Model (CLM) was implemented in DARWIN 10.0. CLM accounts for the relatively large region of constraint loss along the surface-breaking crack front when an applied stress approaches the elastic limit. This method modifies the SIF value for surface-breaking tips using an equation that is calibrated to test data for specific materials. The CLM equation is given by:

$$K = \frac{\sqrt{1 + 4\alpha\beta {K_{el}}^2} - 1}{2\alpha K_{el}}$$

where α and β are positive-valued material-dependent constants and K_{el} is the stress intensity factor at the surface as calculated by the elastic solution.

The CLM method is provided as an alternative surface fatigue correction factor in the GUI Global Properties screen (Figure 10). When this option is selected, the CLM method is used to modify the SIF at surface-breaking tips of a crack during fatigue crack growth analyses. Unlike the legacy BCF method, CLM is a material-dependent model. Accordingly, it requires the addition of the material-dependent constants α and β in the applied material properties file.

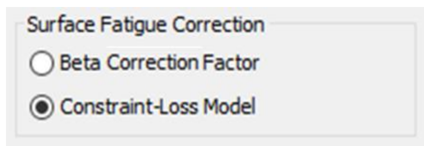


Figure 10. DARWIN 10.0 provides a new constraint-loss model as an alternative surface fatigue crack correction method.

Python-Based Scripting Tool

Process automation has become a major focus in recent years due to the potential for significant cost and time savings. Many companies utilize command line scripting languages (also called “scripts”) to manage the automation process of computer programs. Scripting enables tasks to be performed without human involvement. It is especially useful for complex and/or arduous tasks that may be either too complicated or time-consuming for a human.

DARWIN 10.0 includes a script program called the DARWIN Python Module (DPM) that enables the user to create and/or modify DARWIN input files without the use of the GUI. The script was originally developed to perform routine DARWIN software verification, but is now provided to users enabling them to modify DARWIN input files. DPM is a set of Python scripts that provide a variety of classes and methods to interact with DARWIN project files. It includes the following capabilities:

- Create new DARWIN project files
- Modify input data in existing DARWIN project files
- Query input and output data in DARWIN project files
- Execute DARWIN projects

An overview of the relationship among DARWIN and the DPM is shown in Figure 11. The DPM serves as an intermediary between user-provided Python scripts and the DARWIN project file (*.dhf) and executable program (darwin.exe). The user-provided Python script sends commands to the DPM, and DPM handles the complex tasks of querying, modifying, creating, and executing DARWIN projects.

Project creation methods in the DPM are currently limited to the following analysis modes:

- 2D/3D User-Defined Cracks
- 2D/3D General Inherent Autozoning

Support for additional analysis modes will be considered in the future based on user feedback.

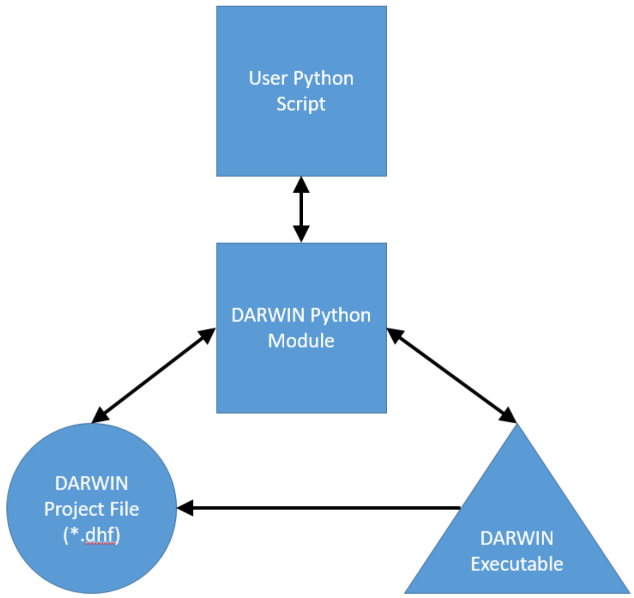


Figure 11. The DPM serves as an intermediary between user-provided python scripts and the DARWIN project file and executable program.

New GUI Menus to Support Objectives-Based Data Structure

In previous versions of DARWIN, the data for all analysis types were stored in one large data structure. This created challenges for the development of new features in DARWIN. For example, the scope of some new features was constrained by the format of the existing data structure. The introduction of new features could also potentially corrupt data associated with existing features.

In DARWIN 10.0, the data structure was reorganized into a number of independent data entities that are called analysis "objectives". An objective is an analysis goal that has a unique set of input and output data requirements. The use of independent data entities enables a unique data format for each objective, eliminating the design constraints and vulnerabilities of the previous data structure. A list and description of the available objectives in DARWIN 10.0 are provided in Table 1.

The objectives-based data structure has led to three notable changes in the GUI menus. The first change is the introduction of a new Project Startup menu that appears when a new project is created (see Figure 12). This menu enables users to declare the project class, project/geometry modes, and units for the project. These key parameters set the foundation for all of the GUI menus that follow and cannot be modified once a project has been initialized.

Table 1. Analysis objectives available in DARWIN 10.0.

Name	Description
Life @ Selected Points	Computes life to failure at user-specified locations.
Surface Life Contours	Computes life to failure at user-specified surface nodes and generates contours on component exterior surfaces (e.g., life, a, c, area, Kmax).
Axial Blade Slot Feature Risk	Computes fracture risk of an axial blade slot in accordance with the methodology defined in FAA Advisory Circular 33.70-5.
Circular Hole Feature Risk	Computes fracture risk of a circular hole feature in accordance with the methodology defined in FAA Advisory Circular 33.70-2.
Surface Feature Risk	Computes fracture risk on selected surfaces of a specified region of a component.
Surface Component Risk	Computes fracture risk on selected surfaces of a component.
Volumetric Life Contours	Computes life to failure at user-specified nodes and generates surface and volumetric contours on a component (e.g., life, a, c, area, Kmax).
Volumetric CICS Contours	Computes critical initial crack sizes at user-specified nodes and generates surface and volumetric contours on a component.
Volumetric Component Risk	Computes fracture risk for a selected volumetric region of a component.

The second change involves a restructuring of the GUI Configuration menu. In previous versions of DARWIN, analysis parameters were specified both in the Configuration menu (Figure 13a) and in downstream menus. In DARWIN 10.0, key analysis parameters are defined in the Project Setup menu, and analysis objectives are defined in the Configuration menu (Figure 13b).

The third change involves a reorganization of the GUI navigation menu (Figure 14). The new navigation menu in DARWIN 10.0 provides an improved work flow for the definition of DARWIN projects. A new menu group entitled "Component" was introduced for the definition of FE models, loadings, and material properties and anomalies. An additional menu group entitled "Objectives" was introduced for the definition of parameters associated with the specific objectives that are selected by the user in the GUI Configuration menu.

The introduction of the objectives data structure is the first and largest step toward a fully revised project framework that will ensure that DARWIN has the foundations to meet the needs of future analytical needs. Additional framework enhancements will be introduced through the development of the version 10 series DARWIN releases. Among the capabilities that users can expect to see in future versions is the ability to perform multiple objectives within a single project.

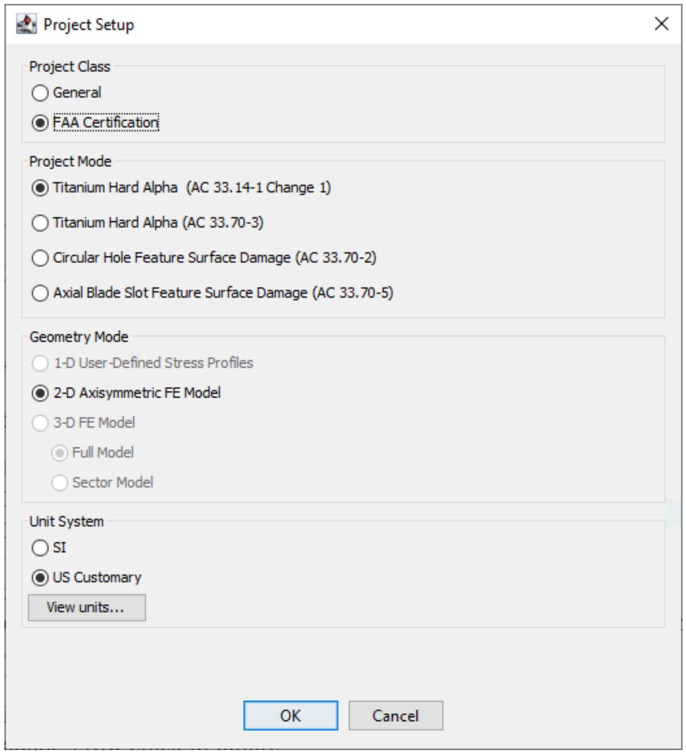


Figure 12. The DARWIN 10.0 GUI features a new Project Setup menu that enables users to declare the project class, project/geometry modes, and units for the project.

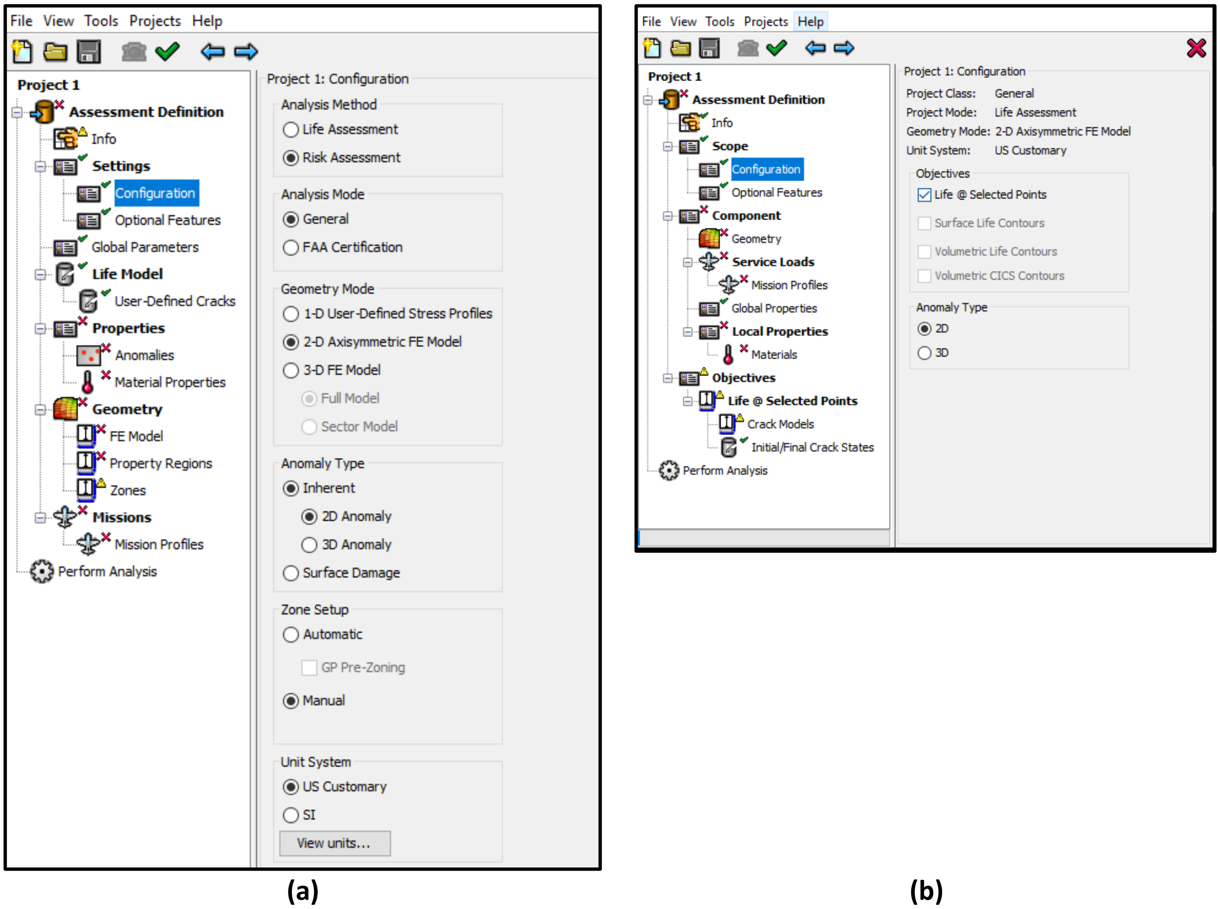
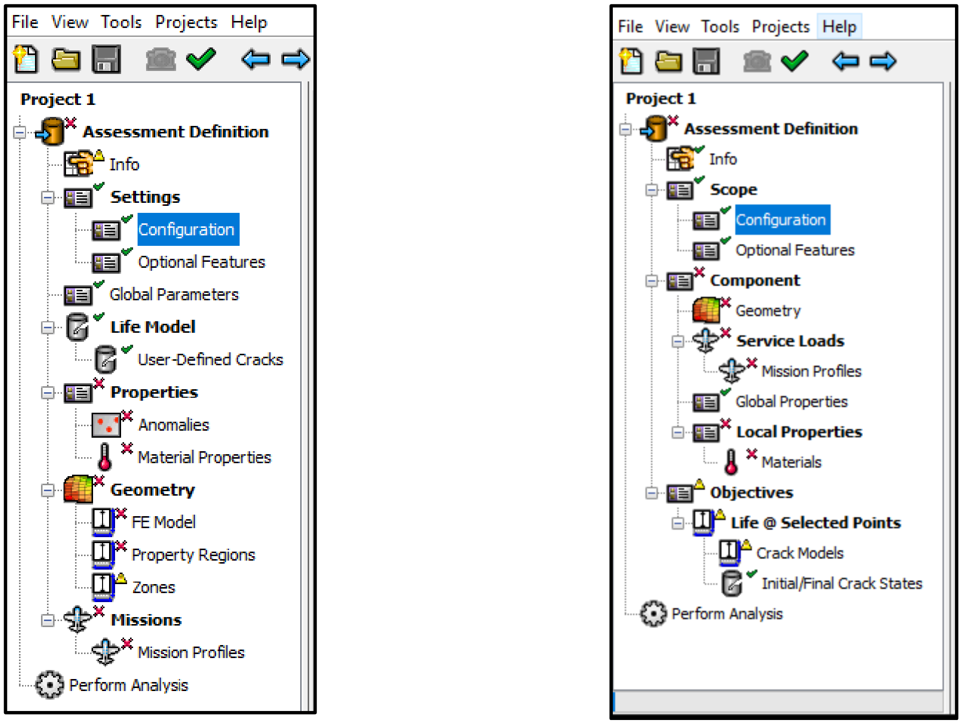


Figure 13. Comparison of DARWIN GUI Configuration menus: (a) previous versions of DARWIN, and (b) DARWIN 10.0. DARWIN 10.0 enables users to define analysis objectives directly in the GUI Configuration menu.



(a) **(b)**

Figure 14. Comparison of DARWIN GUI navigation menus: (a) previous versions of DARWIN, and (b) DARWIN 10.0. The new navigation menu in DARWIN 10.0 provides an improved work flow for the definition of DARWIN projects.

FE2NEU Finite Element Model Quality Checks (MQCs)

Accurate finite element models are required for successful execution of the DARWIN Autoplate algorithm. Fatal errors that occur during a DARWIN assessment involving Autoplate can often be traced back to errors in user-provided finite element (FE) models. To address this problem, a new capability has been introduced in FE2NEU to enable users to identify errors in FE models before they are loaded into DARWIN.

The FE2NEU 7.0 converter program, introduced and included in DARWIN 10.0, includes a new finite element model triage algorithm that assesses the accuracy of finite element models that are intended for use in DARWIN. It performs a series of model quality checks (MQCs) during the conversion of an FE file from its native format to the DARWIN HSIESTA (*.fea) format.

Once the checks have been performed, FE2NEU inserts a digital signature into the *.fea file indicating the status of the checks. If a model fails one or more of the MQCs, FE2NEU will report the checks that failed to the FE2NEU output console and will store the results in the *.fea file. Beginning in DARWIN 10.0, all FE models that are imported into DARWIN will require a digital signature indicating that the MQCs were performed.

FE2NEU performs the following MQCs:

- All nodes must be connected to an element. Orphan nodes are prohibited.
- All nodes must have stresses and temperatures assigned to them for all load cases.
- All nodes must have unique locations. Coincident nodes are not supported.
- All elements must conform to the list of elements supported by the HSIESTA file format.
- All elements must have a positive volume. Negative volumes are prohibited.
- All element faces must have a positive area. Collapsed faces are prohibited.
- Models may only contain one component.
- Duration times for load cases must be consistent, i.e., all defined or none defined.

FE2NEU reports MQC results to the FE2NEU output console and stores the results in the HSIESTA-formatted *.fea file. FE2NEU includes the capability to read existing HSIESTA files that may have been created by an earlier version of FE2NEU or that may have been created by third-party routines. Once the MQCs have been performed, FE2NEU inserts a digital signature into the *.fea file indicating that MQCs were performed. FE2NEU includes a new 'Convert and Sign' button that initiates the MQCs and stores the results and digital signature in the *.fea file (Figure 15).

The DARWIN 10.0 GUI includes a new FE MQC Results Viewer that enables users to view a summary of MQC results associated with a FE model (Figure 16). The Viewer tool is accessed from the Tools > FEA MQC Results Viewer menu in the GUI.

Beginning in DARWIN 10.0, all FE models that are imported into DARWIN will be required to include a digital signature indicating that the MQCs were performed. If a digital signature is not present, the DCE will abort the analysis and an error message will be generated. Ideally, only FE models that have passed all MQCs will be imported into DARWIN. However, some users may need to support legacy FE models that do not pass all MQCs but may still run in some of the legacy DARWIN analysis modes.

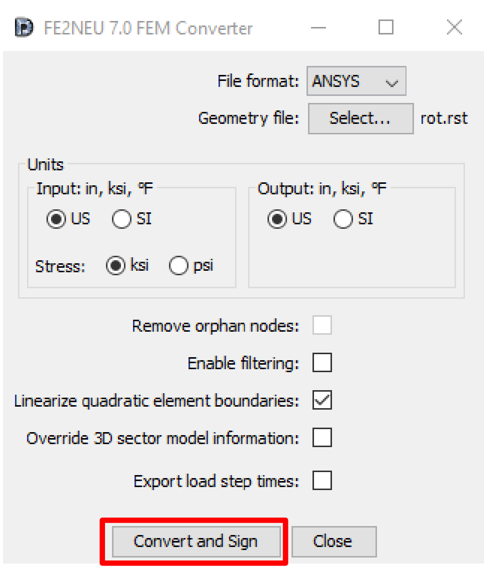


Figure 15. FE2NEU includes a new 'Convert and Sign' button that executes MQCs and stores the results and digital signature in the *.fea file.

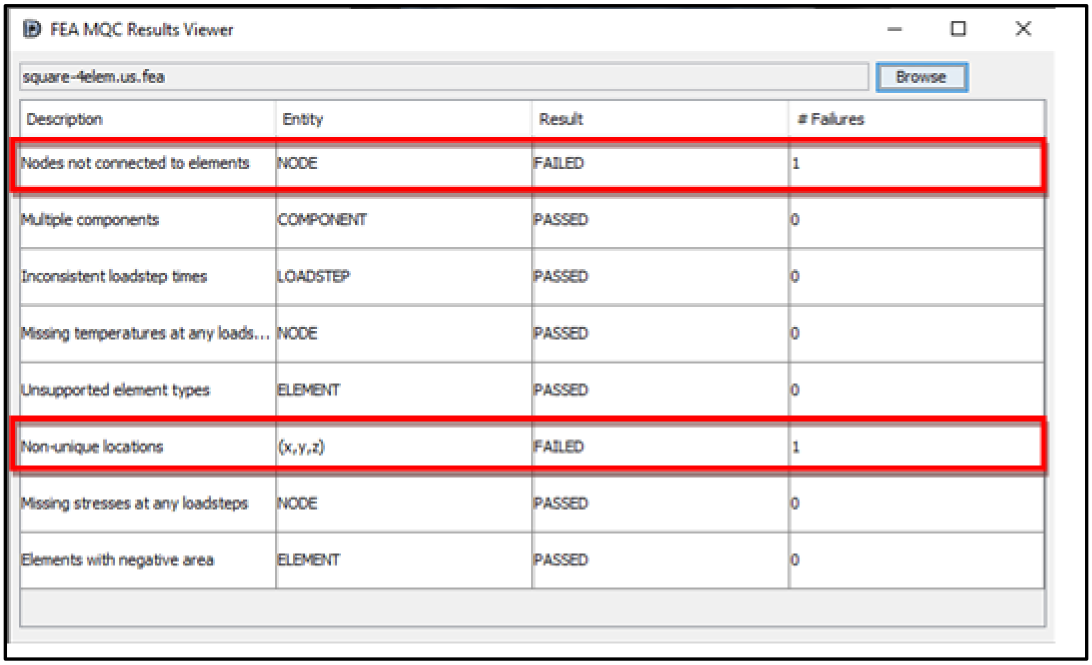


Figure 16. The DARWIN 10.0 GUI includes a new FE MQC Results Viewer that enables users to view a summary of MQC results associated with a FE model.

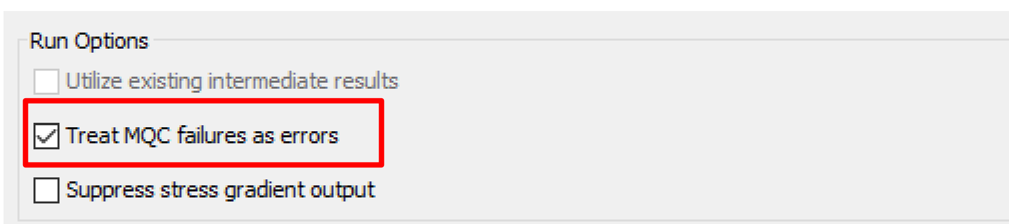


Figure 17. A new button was added to the GUI Perform Analysis menu to enable the User or Superuser to select the desired DARWIN response to FE models containing failed MQCs.

To support this need, a new option is provided in the GUI Perform Analysis menu that enables the User or Superuser to select the desired DARWIN response (either report a warning or report an error) for FE models containing failed MQCs (Figure 17). If the "Treat MQC failures as errors" option is unselected, the DARWIN Computational Engine (DCE) will issue a warning and continue the analysis. If the "Treat MQC failures as errors" option is selected, the DCE will issue an error and terminate execution. Regardless of the selected response, the GUI will always allow the import of an FE model containing MQC failures (and will issue a warning if MQC failures are detected).

Autoplate Node-Skipping Enhancement

DARWIN 10.0 includes a new capability to bypass nodes that encounter computational errors during the execution of Autoplate and the various automodeling algorithms. This new capability enables the DARWIN computational code to continue to run (via an internal wrapper program) even when the DARWIN code has terminated. Using existing restart capabilities built into the DARWIN automodeling methods, the wrapper program iteratively restarts an analysis each time an unexpected termination-inducing error occurs. During the restart of the analysis, DARWIN flags the failed node and resumes the analysis beyond the failed node. This enhancement enables DARWIN to create life/CICS/risk contours when fracture models are not available at all nodes in an FE model.

Autoplate Speed Enhancement for Large 3D FE Models

The computation time of all DARWIN automodeling features is dependent on the speed of the Autoplate algorithm. DARWIN 10.0 includes an enhancement to Autoplate that may significantly reduce the computation time for automodeling operations in large 3D FE models. In previous versions of DARWIN, the autoplate algorithm constructed a 2D FE model of the model geometry at the initial crack location in a 3D FE model. The resulting 2D FE model geometry and associated stress/temperature values were used for all autoplate computations at the location. This process was repeated at each initial crack location and could be time-consuming for computing FCG life and fracture risk contours in large 3D FE models.

In DARWIN 10.0, the FE2NEU finite element results file translator was enhanced to identify the finite elements that are located on the exterior surfaces of 3D finite element models. This information is provided to the Autoplate algorithm during run time and enables it to quickly identify the boundaries of the 3D geometry that are coincident with the crack growth plane at each initial crack location. This enables Autoplate to extract geometry and stress/temperature information directly from the 3D FE model. This new capability reduces the computation time for Autoplate and associated automodeling operations for large 3D FE models. The impact of this enhancement on automodeling speed will vary depending on FE model size and desired project objectives (e.g., FCG life at user-specified locations, FCG life contours, fracture risk).

Automodeling Speed Enhancements

DARWIN 10.0 includes enhancements to the Autoplate and OpenMP parallel processing algorithms that significantly improve computation speed for 3D FE models. Autoplate includes an algorithm that locates geometry boundaries on 2D slice planes of 3D FE models. The algorithm was restructured to locate the boundary in 30% of the time required by the previous algorithm. The OpenMP parallel processing capability includes an algorithm that manages the generation and storage of seed values associated with Monte Carlo simulation. The algorithm was restructured to manage seeds in 75% of the time required by the previous algorithm.

These enhancements have resulted in a 50-70% reduction in the computation time required to generate life and CICS contours for 3D FE models, and a 50% reduction in the computation time associated with risk assessment of 3D FE models (when parallel processing is used). Actual speed improvement is dependent on FE model size and the number of CPUs applied in parallel processing.

Speed Enhancements for User-Defined Cracks

DARWIN 10.0 includes GUI and Autoplate enhancements that significantly improve computation speed for creation of user-defined cracks in large 2D and 3D FE models with large numbers of load steps. In previous versions of DARWIN, the crack growth plane was identified by the GUI prior to executing the autoplate algorithm. This process involved the GUI loading the entire finite element stress results for all load steps into memory. It required a significant amount of computer memory (potentially exceeding available computer memory for large FE models) and could take several minutes to complete at a single location. To resolve this problem, the Autoplate algorithm was enhanced to directly identify the crack growth plane for user-defined cracks. It has the capability to extract stresses at single elements and nodes (requiring significantly less memory), and computes the crack growth plane in a fraction of the time that was required by the GUI. The GUI was also enhanced to decrease the number of times that the display screen is refreshed to further improve computational efficiency.

These enhancements have resulted in an 85-96% reduction in the computation time required to create user-defined cracks in large 2D and 3D FE models with large numbers of load steps. Actual speed improvement is dependent on FE model size and the number of load steps that are applied. For example, the time required to place a crack and create a fracture mechanics plate at a single location in a large FE model (225,000 elements, 1000 load steps) was reduced from 250 seconds to 18 seconds. The GUI random access memory (RAM) associated with this model was reduced from 12.5 GB to 1.9 GB. The enhancement has been backported to DARWIN Version 9.2.

DARWIN Results File Import Speed Enhancement

DARWIN 10.0 includes a GUI enhancement that significantly reduces the time required to import DARWIN results files. In previous versions of DARWIN, the entire results file had to be loaded into memory before any of the results could be displayed in the GUI. In DARWIN 10.0, the GUI was enhanced to enable selected portions of the results file to be loaded into memory as needed to display specific results as they are selected by the user. This enhancement provides a 90-98% reduction in the time required for the GUI to open 2D and 3D results files with large numbers of zones (i.e., 500 or more zones). It reduces the time required to open a results file and display component risk results (85-88% reduction), and also reduces the time required to open a results file and display risk contours (12-74% reduction) for 2D and 3D FE models with large numbers of zones.

GUI Speed Enhancement for Large 3D FE Models

DARWIN 10.0 includes a speed enhancement that significantly reduces the GUI response time when working with large 3D FE models. In previous versions of DARWIN, the GUI would perform a number of operations each time the user performed an action such as entering/importing data or clicking on a node in a FE model. For small FE models the response time was on the order of a few seconds, but for large 3D FE models the response time could be several minutes or more.

In DARWIN 10.0, several improvements were implemented in the GUI to reduce the response time when working with large 3D FE models. These improvements include: application of contour data to the model all at once to avoid multiple redundant display events, refresh only the contour that is currently displayed (rather than all contours), reload data only when data changes have been confirmed, initialize data only for algorithms that are currently in use, and removal of redundant display algorithms. The speed enhancement has the most impact on geometry-related preprocessing operations such as assigning elements to a geometry, selecting a crack location, and viewing stress/temperature contours for specific load cases.

This speed enhancement provides approximately 75% reduction in the GUI response time associated with large 3D FE models. This estimate is based on the response time required to perform GUI preprocessing tasks associated with a fatigue crack growth life assessment at user-specified locations in a 1 million element 3D FE model (Windows 10, dual 2.3 GHz processors, 16 GB allocated RAM). Note that GUI response times will vary depending on the FE model size, number of project objectives, computer operating system, CPU power, and available RAM (among other factors).

Formation Module Programming Guidelines for Improved Execution Speed

The User Formation Module capability in DARWIN enables users to incorporate their own crack formation life algorithms for use in fatigue crack growth life and fracture risk assessments. The user is responsible for writing and compiling a formation module library using a set of source files that are distributed with DARWIN. An application programming interface (API) is included that enables the User Formation Module to exchange data with DARWIN during run time.

Some users have reported computational efficiency problems when implementing and using their own crack formation life algorithms in the User Formation Module. A review of user-provided formation module libraries has revealed that the use of allocated arrays within the User Formation Module may significantly influence the execution time of the module. In one particular case study, User Formation Module computation time was reduced by a factor of 25 (160 sec vs 6 sec) when unnecessary allocated arrays were removed.

In DARWIN 10.0, the programmer's documentation for the User Formation Module has been enhanced to provide guidelines regarding the storage and retrieval of data within the module. Users are advised to retrieve project data from DARWIN's in-memory database (accessed via available API functions) on an as-needed basis rather than storing this information in local memory. The use of allocated arrays should be avoided whenever possible. Additional information, including specific programming examples, is provided in the User Formation Manual programmer's documentation.

Updated Limits for Stress Intensity Factor Solutions

Many of the stress intensity factor (SIF) solutions in DARWIN are also included in NASGRO, a fracture mechanics-based life prediction tool developed and maintained by SwRI and NASA. Enhancements to the NASGRO and DARWIN SIF solutions are often performed independently, so periodic updates are performed to both codes to provide consistency among their SIF solutions.

In DARWIN 10.0, SIF validity limits and transition criteria were revised for consistency with NASGRO. The revised validity limits and crack transition criteria are shown in Figures 18-22. In most cases, the new validity limits are slightly wider than the previous limits. Note that the revised SIF validity limits and crack transition criteria have been set as the default values.

The impact of the revised SIF validity limits and crack transition criteria on calculated lives is generally small or negligible. In the instances where impacts that are more significant may occur, the revised validity limits will generally cause calculated lives to increase.

In previous versions of DARWIN, the size of the monotonic plastic zone was included in the effective crack size used for transition calculations. This approach can be conservative in some cases and was inconsistent with NASGRO. In DARWIN 10.0, the monotonic plastic zone is no longer included in crack transition calculations.

Users have access to legacy SIF limits/transition criteria and plastic zone size correction via the hidden feature keyword "legacySIFLimits". When this keyword is enabled, the user has the option to select the legacy values (Figure 23).

	Old	New
Offset: y_d	Stops when y_d/h_y is not within [0.05, 0.95]	No restriction[1]
Effective crack lengths[2]: a_e^+ , c_e^+ and c_e^-	Transitions when $a_e^+/h_x \geq 0.9$ $c_e^+/(h_y - y_d) \geq 0.9$ $c_e^-/y_d \geq 0.9$	Extended limits for transition from 0.9 to 0.95

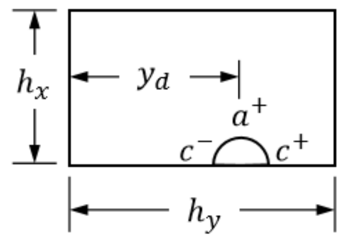


Figure 18. Revised validity limits and crack transition criteria for SC30 and SC31.

	Old	New
Offset: y_d	No restriction; extrapolated	No restriction
Effective crack lengths: c_e^+ and c_e^-	Transitions when $c_e^+/(h_y - y_d) \geq f$, $c_e^-/y_d \geq f$; f = 0.8 or 0.9 depends on offset.	Extended limits; f = 0.9 for SC18 f = 0.95 for SC29

Figure 19. Revised validity limits and crack transition criteria for SC18 and SC29.

	Old	New
Crack lengths: a^+ and a_e^+	For SIFs at $a^+/h_x > 0.96$, use ref. sols. at $a^+/h_x = 0.95$	Transitions when $a_e^+/h_x > 0.95$
Crack lengths: c^+ and c_e^+	For SIFs at $c^+/h_y > 0.96$, use ref. sols. at $c^+/h_y = 0.95$	Transitions when $c_e^+/h_y > 0.95$

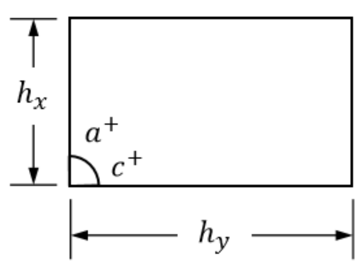


Figure 20. Revised validity limits and crack transition criteria for CC09 and CC11.

	Old	New
Effective crack lengths: c_e^+	Transitions when $c_e^+/h_y \geq f$ f = 0.8 or 0.9 depends on hole offset.	Transitions when $c_e^+/h_y \geq 0.9$

Figure 21. Revised validity limits and crack transition criteria for CC08 and CC10.

	Old	New
Offsets: x_d and y_d	No restriction	No changes
Eff. crack lengths: a_e^+ , c_e^+, a_e^- and c_e^-	Transitions when $a_e^+/(h_x - x_d) \geq 0.99$ $c_e^+/(h_y - y_d) \geq 0.99$ $a_e^-/x_d \geq 0.99$ $c_e^-/y_d \geq 0.99$	No changes

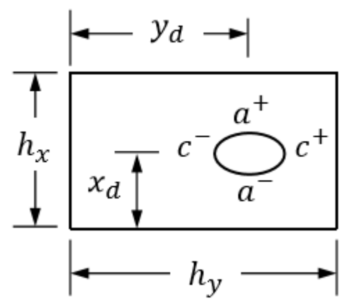


Figure 22. Revised validity limits and crack transition criteria for EC04 and EC05.

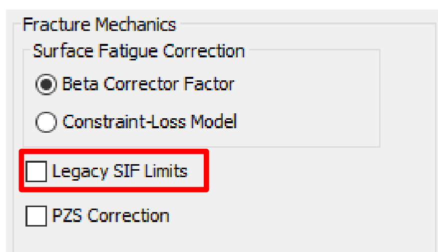


Figure 23. When the "legacySIFLimits" hidden feature keyword is enabled, the user has access to legacy SIF limits/transition criteria and plastic zone size correction.

Deterministic Assessment Enhancement

DARWIN includes a "Life @ Selected Points" objective that enables users to specify the locations of cracks for life assessment. In previous versions of DARWIN, users were responsible for defining the crack type (e.g., SC30, EC04, etc.), crack growth plane, and fracture plate dimensions and orientation. The GUI input data check tool issued an error if the user did not provide a crack type, crack growth plane, or fracture plate parameter definition. However, the DARWIN Autoplate algorithm now computes these quantities, so they are no longer required inputs.

In DARWIN 10.0, the user no longer needs to provide the crack type, crack growth plane, and fracture plate dimensions and orientation when using the user-defined cracks mode. These are optional inputs that the user can provide if desired. The GUI input data check tool was enhanced to issue a warning (rather than an error) when the user does not provide a crack type, crack growth plane, or fracture plate parameter definition in the Life @ Selected Points objective (Figure 24).

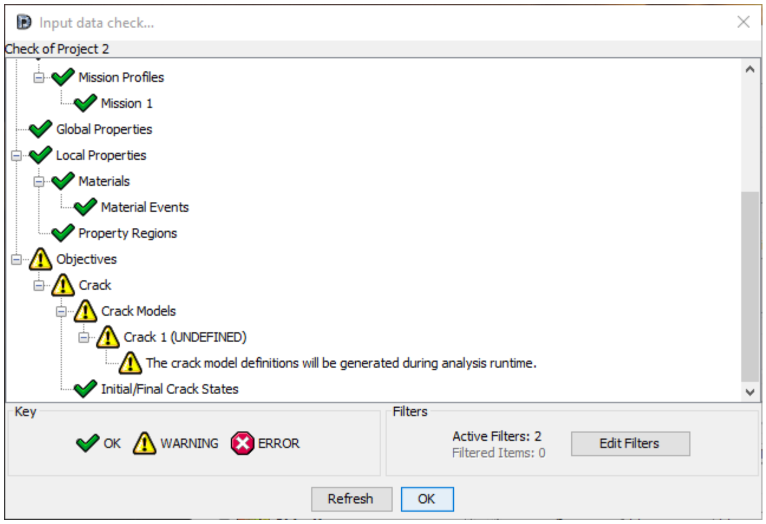


Figure 24. The GUI input data check tool was enhanced to issue a warning (rather than an error) when the user does not provide crack type, crack growth plane, or fracture plate parameter definition in the user-defined cracks mode.

User-Defined Project Rules

A number of new optional features are now available in DARWIN. Many organizations wish to provide default numerical values for their users that may differ from the default values that are provided in DARWIN. In addition, some organizations do not need all features, and wish to make a reduced set of features available to their users.

DARWIN 10.0 was enhanced to enable an individual company to specify default values for selected DARWIN inputs and to specify the DARWIN features that are available to their users. This enhancement enables the use of user-defined project rules that are defined via files called 'templates'. A template is a special DARWIN project file that contains user-defined project rules regarding field restrictions and defaults. Templates can be used to initialize new projects (called template-based projects) that are forced to adhere to the rules defined in the template.

Templates can be used for a variety of purposes such as setting project defaults, preventing users from using unauthorized features, and forcing projects to use particular field values. In addition, templates can be used to create analysis modes. An analysis mode is a targeted DARWIN project file that is used to analyze particular component types. Templates designed for analysis modes may have very restrictive rule sets.

DARWIN 10.0 provides a dedicated GUI environment for the creation of templates. This mode (template mode) is available only to super-users (also known as template masters). Template mode differs from the standard DARWIN GUI environment in two ways. First, template mode only contains a subset of the preprocessing screens provided in the standard DARWIN GUI environment (Info, Configuration, Optional Features, and Global Properties). Second, template mode contains additional fields called disablers that are used to disable or force field values depending upon the field type.

Within template mode, a template master creates a project in an identical fashion as would be done for creating any other DARWIN project. The template master then selects a value for each field and activates the value-specific disablers as necessary to disable or force field values (Figure 25). For multiple option fields (e.g., TMF Crack Growth) disablers prevent users from selecting the field value in template-based projects. For checkbox fields (e.g., In-Service Inspections) and input fields (e.g., # Samples) disablers prevent users from changing the default field value. When the template master saves the project, the GUI stores the template project file in the active template directory. Apart from allowing users to create templates, template mode also allows users to modify existing templates.

In the standard DARWIN GUI environment (non-template mode) templates can be used to initialize DARWIN projects. When a new project is initialized in the DARWIN GUI and one or more template projects are available in the active template directory, the prompt shown in Figure 26 appears to enable the user to specify if the project will be based on a template file. An organization may elect to limit users to template-based projects only if desired.

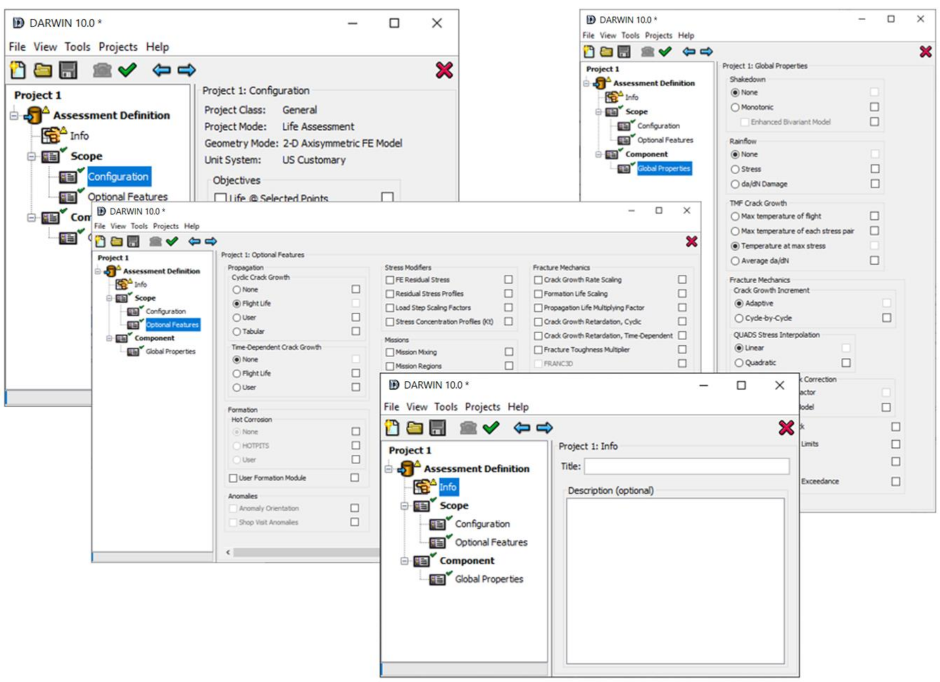


Figure 25. In template mode, organizations can set rules for fields residing in the Configuration, Optional Features, and Global Properties pre-processing screens.

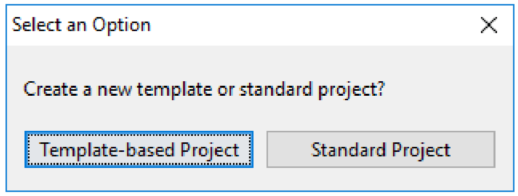


Figure 26. A new project type prompt appears when templates are available in the active template directory.

Setting GUI Preferences via an External File

The DARWIN GUI environment is defined via the GUI preferences menu and enabled hidden features. The GUI preferences menu options govern GUI pre-processing, execution, post-processing behaviors. Hidden features govern the availability of organization-specific capabilities. Each organization is likely to have a unique GUI environment, and control of that environment may be critical to ensure consistent usage of the DARWIN program among analysts.

DARWIN 10.0 includes a new capability to configure the GUI environment via an ASCII text-based file called a GUI Configuration File (GCF). The GCF is imported into DARWIN during installation and imposes specific preference options and hidden features that are available each time the GUI is used. A common DARWIN GUI environment can be imposed throughout an organization by simply assigning the same GCF file to all DARWIN installations.

A GCF follows a simple key-value file format. Each environment-related field is associated with a unique key and an expected input type. The presence of field-specific keys are optional. For unspecified fields, the DARWIN GUI applies the default values. For specified fields, the provided values are applied and cannot be modified by a user via the GUI. An example GCF is shown in Figure 27.

```
ENHANCED_BIVARIANT_PLATES=TRUE
LINK_MATERIAL_FILES=FALSE
MAX_CPUS=6
OUTPUT_INTERMEDIATE_RESULTS=TRUE
LIBRARY_DIRECTORY=../../darwin/libraries
```

Figure 27. Example GUI configuration file.

Two methods are provided for importing a GCF into a DARWIN installation. The first method involves placing the GCF in a dedicated subfolder in the DARWIN installation directory. Upon startup, the GUI checks the dedicated subfolder for a valid GCF and automatically applies the GCF rules if a file is found. The second method involves specifying the path to a valid GCF. The GCF path is defined in the GUI Preferences menu. This method is particularly useful if an organization intends to use a common GCF to control the GUI environment for DARWIN installations on multiple workstations.

User's Guide Keyword Search Enhancement

The DARWIN User's Guide is a large and comprehensive set of documentation that covers a wide range of DARWIN-related topics. In previous versions of DARWIN, the User's Guide was available as a set of individual PDF documents, each covering a specific topic in DARWIN, that were accessed via the GUI Tools menu. This format prevented users from performing a keyword search among the entire content of the DARWIN user manual.

In DARWIN 10.0, an enhanced document format called PDF portfolio was adopted for the User's Guide. This format bundles the individual PDF files into a single PDF file, enabling users to perform keyword searches within one or all of the DARWIN User's Guide documents simultaneously.