



# *Laser Processing with Raylase*



**RAYGUIDE**

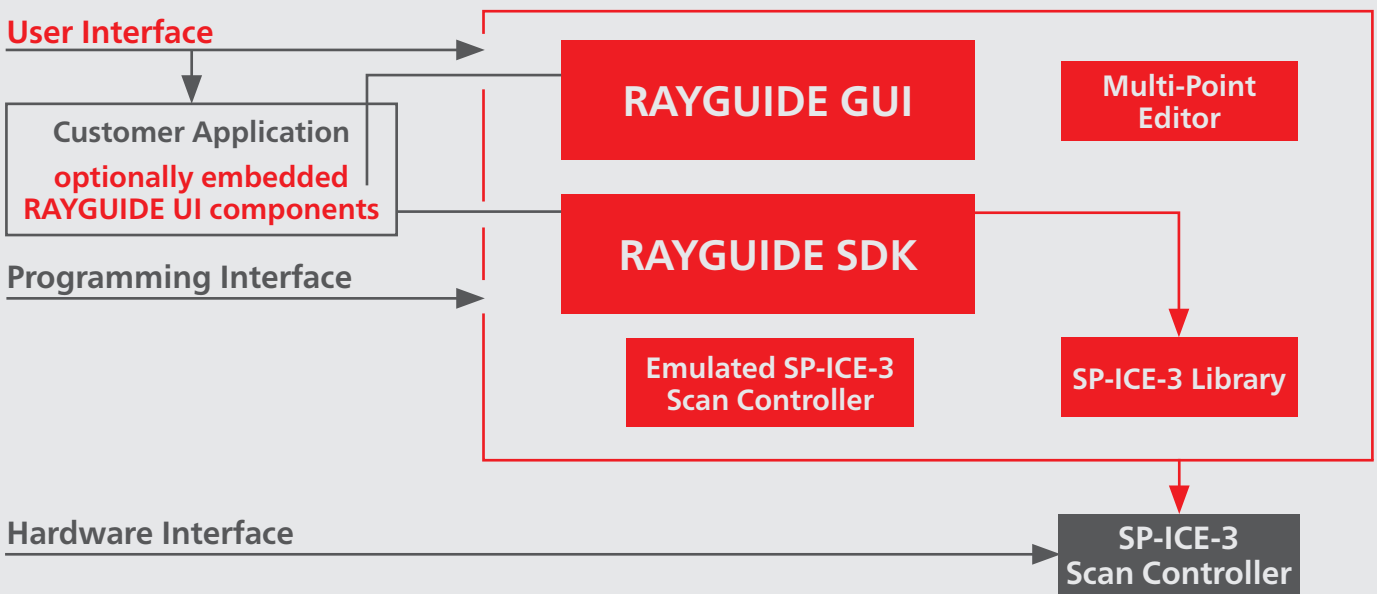
THE POWER OF WE.

RAYGUIDE is the modern laser processing software, developed by RAYLASE for end-users and programmers with the goal to increase usability and decrease programming complexity. RAYGUIDE comes with all fundamental tools to setup and calibrate the galvo-scanner-system, create laser processing jobs and execute them with all required automation including a continuously expanding range of features and functions. RAYGUIDE utilizes the full feature capacity of the SP-ICE-3 control card to complete a powerful laser processing package.

# 1. Basic Interface Structure

Vision: The API should contain all functions of the GUI and support them with easily adaptable sample code.

- RAYGUIDE user Interface for quick and easy interaction
- RAYGUIDE programming library for highest level of automation and integration



## RAYGUIDE API:

### Available as RAYLASE SDK License for programming your own application

- RAYGUIDE SDK License grants access to RAYGUIDE GUI for system configuration and calibration
- API programming in .NET environment
- Create your laser process job programmatically or just modify jobs created by RAYGUIDE GUI as required to automate your process
- RAYGUIDE API offers complete range of functions as the RAYGUIDE GUI
- Supported by sample codes and tutorials, it was never easier to benefit from the powerful predefined functions and integrate them into your system HMI

```

// 1. Create the API:
using ( MarkerAPI markerAPI = new MarkerAPI() )
{
    // 2. Create and assign the devices:
    // 2.a) Create the SP-ICE-3 device:
    IDeviceManager deviceManager = markerAPI.DeviceManager;
    deviceManager.AddDevice( "my SP-ICE-3 card", typeof( SPICE3Device ) );
    BaseScanController scanController = (BaseScanController)deviceManager.GetDevice( "my SP-ICE-3 card" );
    scanController.IPAddress = "169.254.0.98"; // <- put your SP-ICE-3 card's IP address here
    scanController.EnableLogging = true;

    // 2.b) Create and assign the scan head:
    GenericScanHead scanHead = new GenericScanHead();
    scanController.AssignScanHead( scanHead );

    // 2.c) Create and configure the laser device as needed by your hardware:
    scanController.LaserController = new CO2LaserDevice();
    scanController.LaserController.LaserProfile.HotPowerTarget = PowerTarget.LmkWidth;
    scanController.LaserController.LaserProfile.FpsPolarity = Polarity.ActiveHigh;
    scanController.LaserController.LaserProfile.GatePolarity = Polarity.ActiveHigh;
    scanController.LaserController.LaserProfile.LMPolarity = Polarity.ActiveHigh;

    // 2.d) Connect with the hardware:
    try
    {
        scanController.Initialize();
    }
    catch ( ApplicationException ex )
    {
        throw new ApplicationException( $"Initializing SP-ICE-3 card with IP address {scanController.IPAddress} failed.", ex );
    }

    // 3. Create a sample job:
    // 3.a) Create a new job and add your card to it:
    IJobManager jobManager = markerAPI.JobManager;
    JobDefinition jobDefinition = jobManager.CreateNewJob( "Hello World" );
    jobDefinition.ScanControllers.Add( scanController );

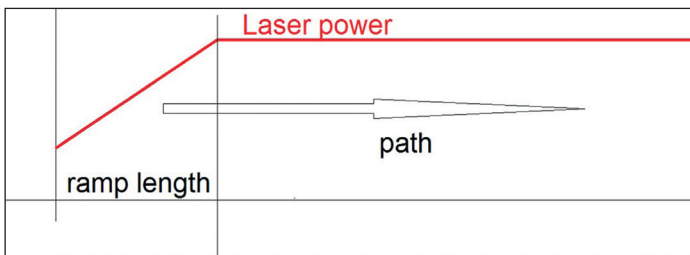
    // 3.b) Create a rectangle of size 20mm x 10mm:
    MarkableRectangle rectangle = new MarkableRectangle ( Size = new dvec2( 20000, 10000 ) );
    rectangle.MoveTo( new dvec3( 0, 0, 0 ) );
    VectorGraphicMarkerProfile profile = (VectorGraphicMarkerProfile)rectangle.MarkableConfiguration.MarkerProfile;
    profile.MarkingMode = MarkingMode.OutlineFilling;
}
    
```

## 2. Features for basic process tasks

### 2.1 LASER PROCESS PARAMETERS

To maintain the best process results, RAYGUIDE in combination with the SP-ICE 3 scan controller card supports the basic process parameters and advanced features such as:

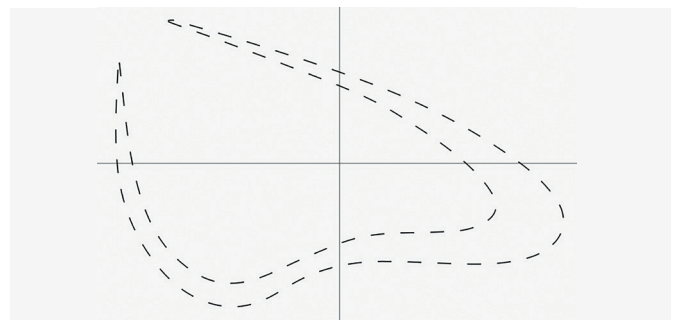
- Velocity based power correction: Enables scan controller to automatically adjust laser power depending on the actual process speed to achieve constant laser power density on the material
- Skywriting: Creates sharp corners and avoids energy burn-in at the beginning/end of the vector due to the advanced synchronization of laser and deflection unit



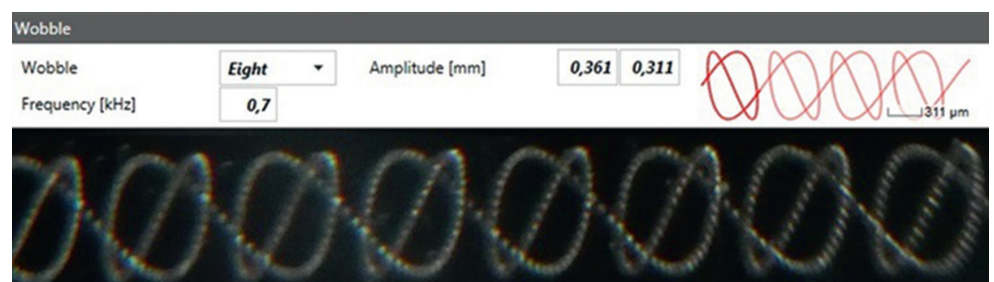
- Ramping: Advanced laser control for welding applications. Ramping the laser power at start and end of path allows to adapt the laser impact to individual process request



- Dashed Line: Enables to structure vector path with regular or non-regular gaps, independent to the style of the contour, e.g. scribing of easy-opening packaging



- Wobble: Predefined geometries such as circles, eight and lissajous figures available. Shape preview function eliminates trail and error approach to parametrize chosen wobble shape

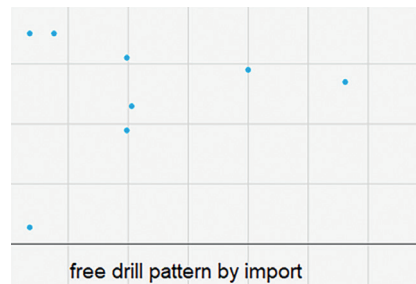


“ Vision: Automated unit tests ensure highest software quality and short development cycles. ”

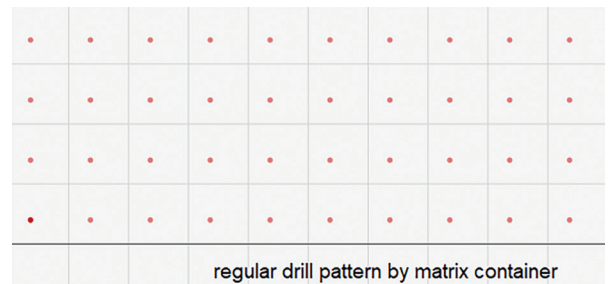
## 2. Features for basic process tasks

### 2.2 DRILL PATTERN

- Drill points can be defined by number of pulses or laser on duration
- Variable drill pattern can be imported by coordinate table while homogeneous drill pattern can be generated by matrix copy container



free drill pattern by import



regular drill pattern by matrix container

Vision: The graphical interface of the RAYGUIDE should be self-explanatory, easy to use and have a modern appearance.

### 2.3 TEXT OBJECTS

- True Type Font (TTF), Open Type Fonts (OTF) and Laser Stroke fonts
- All kinds of source possible, to realize e.g. time/date stamp, serial numbers or the combination
- Auto fit of text into a predefined placeholder, independent of its context or font
- Configure its processing direction in regular writing direction or opposite

Preview	Data
001 RAYGUIDE 16:37:45	Source: Custom
	Text: %O RAYGUIDE %t
	Add placeholder: v
	Start: 1 Increment: 1
	Batch: 1 Digits: 3
	Current iteration: 0
	Auto reset: Never

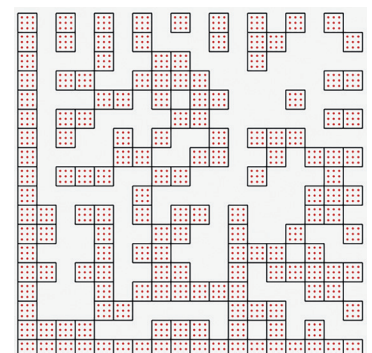
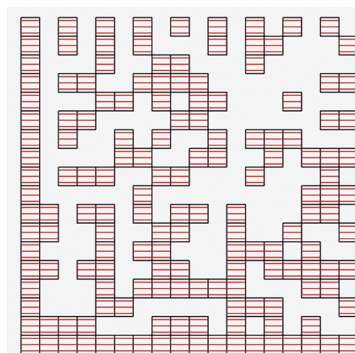
### 2.4 FILLING STRUCTURES

- Choose the optimal fill style for layouts to achieve fastest processing time. For example, *inset fill* can be faster than *hatch fill*
- Use templates to avoid redundant fill pattern generation

**Inset**

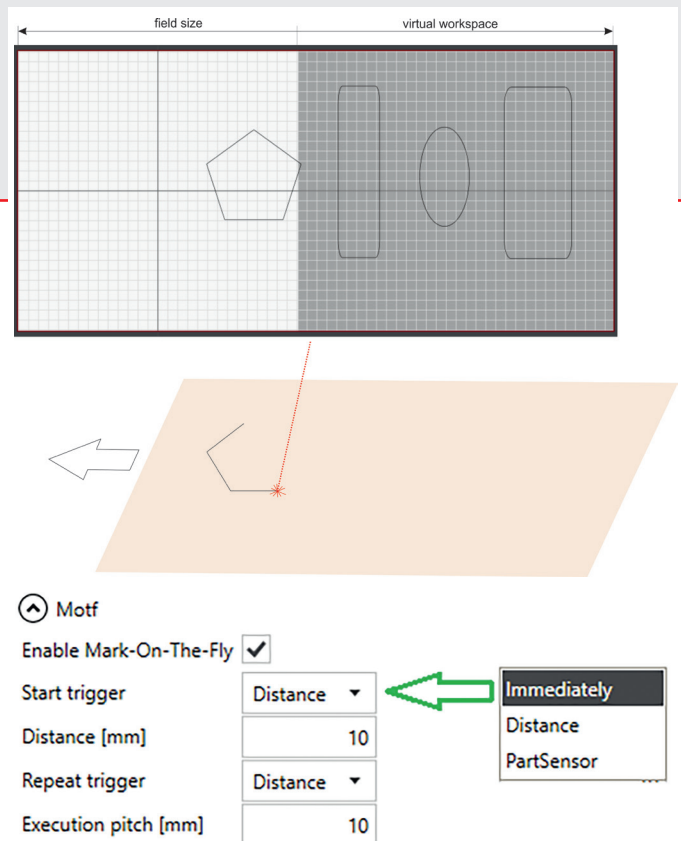
**Hatch**

- Fill codes easily with defined number of filling lines or drill dots that RAYGUIDE will distribute uniformly



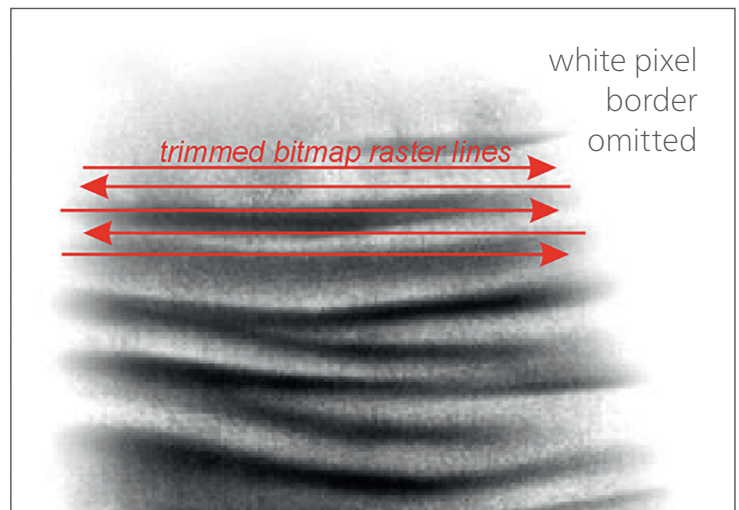
## 2.5 MARKING-&-PROCESSING-ON-THE-FLY (MOTF)

- Define virtual workspace and place objects as to be marked on workpiece
- Use simulation encoder to evaluate the maximum possible belt speed
- Review pending part buffer counter
- Possibility to suppress part sensor input to ignore secondary signal edges
- All variations of trigger options are supported



## 2.6 BITMAP MARKING

- 2 Processing Modes: Point&Shoot vs Sprint Mode
- Horizontal vs Vertical line processing
- Easy Power Scaling to achieve fast result depending on the target material
- Faster processing by trimming the raster to the minimal necessary



## 2.7 JOB ANALYTICS

- Job statistic provides detailed information and calculated process time prior to real processing

- View options provide information about mark- and jump order within an object, and can highlight special defined corners

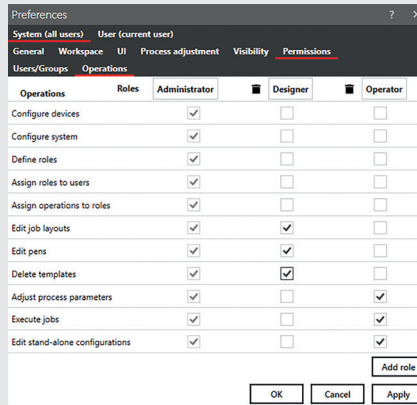


Job "Job 1" settings			
Settings Statistics			
SP-ICE-3 SN123			
Markable objects	3		
	Contour	Filling	Total
Total mark length [mm]	25515,255	0	25515,255
Total jump length [mm]	23789,101	4906,227	28695,328
Execution time [m:s:ms]	01:00.905	00:02.595	01:03.500
Number of commands	363	1422	1785
Number of points	1520	1422	2942
Number of scan lines	463	0	463

# 3. Features for demanding process requirements

## 3.1 USER PERMISSIONS

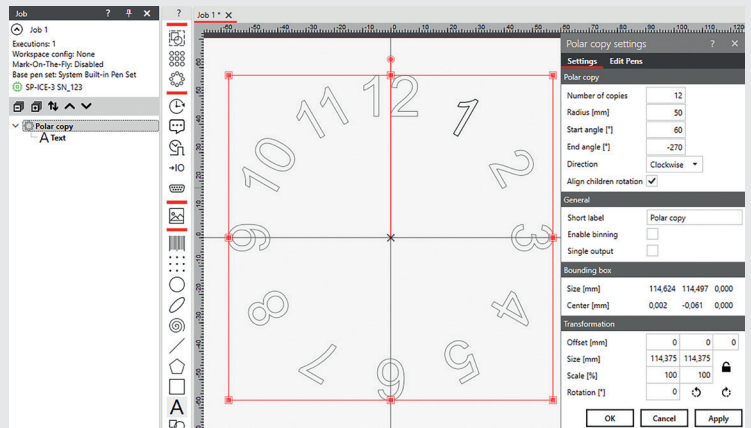
- Allow system owner to establish different user permissions
- No additional user management needed since RAYGUIDE uses Windows user/user groups
- Flexible assignment of user roles and their allowed actions
- If editing is restricted, the related dialogs become read only for reviewing



“ Vision: The intelligence of RAYGUIDE should be easily integrated into the software framework of the customer. ”

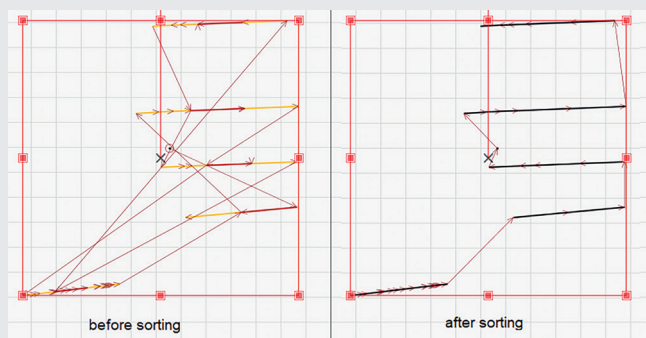
## 3.2 POLARCOPY CONTAINER WITH CENTERED ADJUSTMENT

The PolarCopy Container does align its content along a circle with the additional option to adjust its children so they are pointing to center

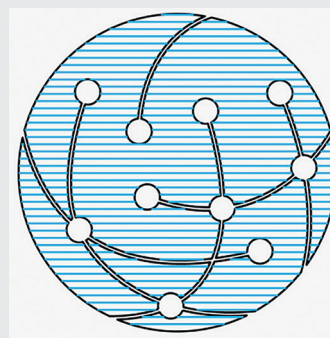


## 3.3 COMPREHENSIVE VECTOR EDITING

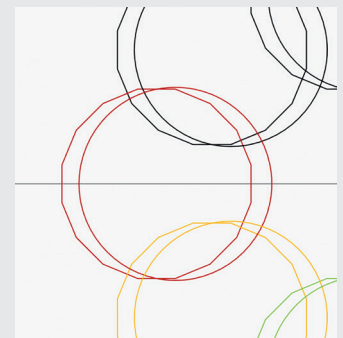
- Edit imported graphic files to consider laser-marking requirements
- Convert layout object defined by content (e.g. text) into real vector based objects
- Optimize marking order easily in job tree by drag & drop
- Combine or split vector objects
- Sort vector order automated e.g. to eliminate unnecessary jumps



Samples:  
Vector Sorting



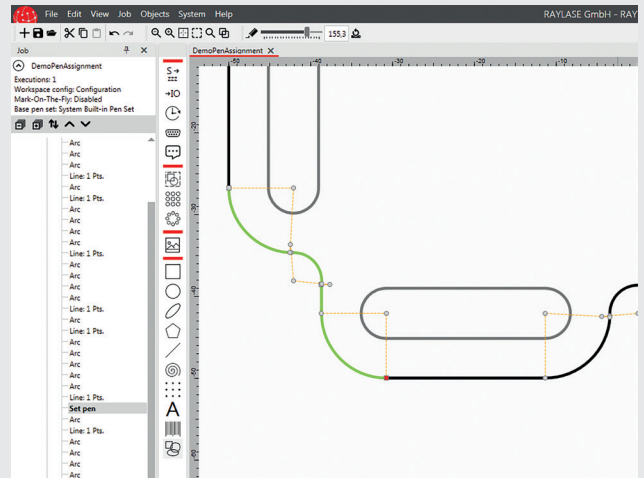
Sample: Edit layouts so they become enabled to get filled



Sample: Convert a circular polyline into an circular arc in one operation

### 3.4 PEN MANAGEMENT AND PEN ASSIGNMENT

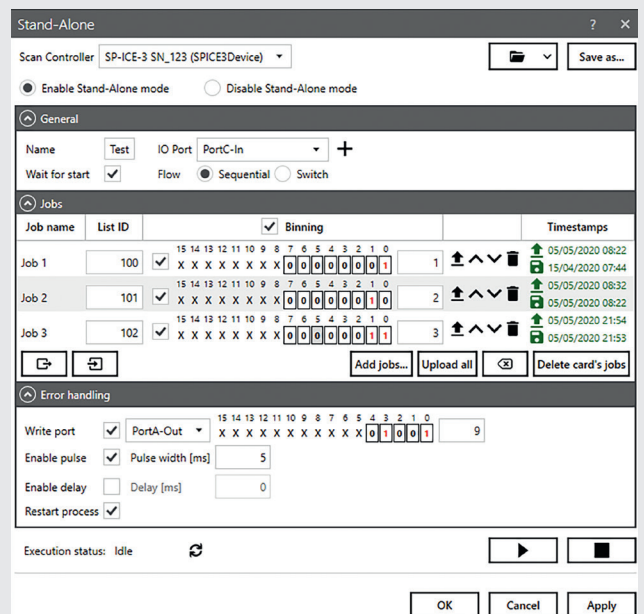
- Pen and layout objects are managed independently
- Unlimited number of pens available
- Store valuable process parameter in pen library
- Assign pens on the most granular way: per object, per layer, per path, inside a path
- Easily overview of assigned pens(s) to objects
- Address all marking quality related features by pen



### 3.5 AUTOMATED JOB EXECUTION

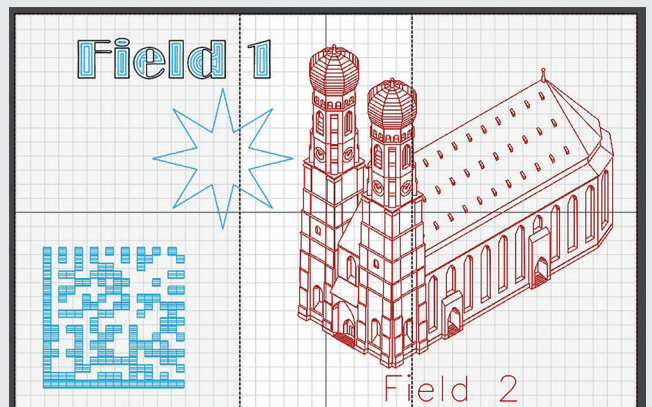
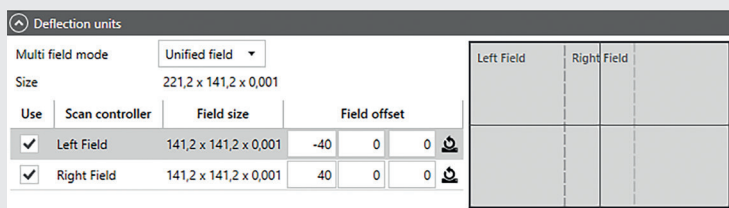
Preparing the Stand-Alone execution mode of the scan controller, job execution can be fully controlled by PLC

- The stand alone dialog allows to add and download multiple laser process jobs in one operation
- Easily assign the binning conditions and maintain the job overview by using the csv-export/import option
- The time stamp display makes any job content that has been updated since the last download noticeable
- Define the desired action in case of an error event
- Using a previous stored configuration, it is easy to clone the stand-alone setup onto another SP-ICE-3 scan controller



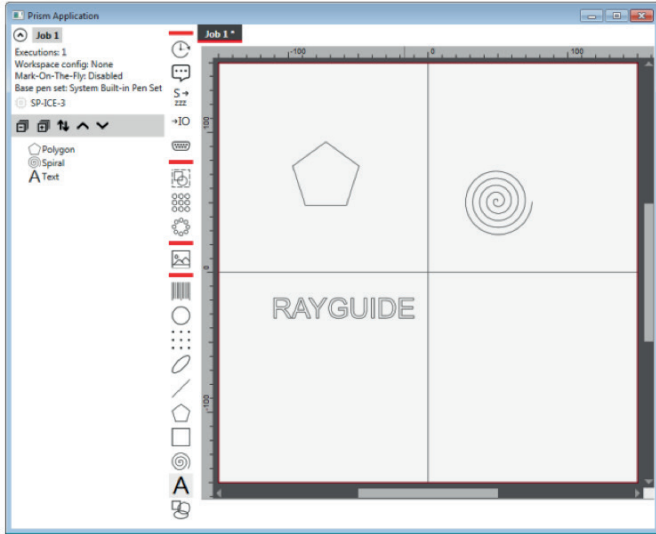
### 3.6 MULTIPLE SCAN FIELD MANAGEMENT

- 3 Options can be used to setup multiple scan fields: Unified, Intersected or Individual
- Viewport displays scan fields as arranged in machine
- Option to process one job synchronized with several scan controllers involved



## 4. Features for special process requirements

Dialog with an embedded RAYGUIDE instance



Vision: it is quite important, that users enjoy working with RAYGUIDE and discover its usability, the little improvements make the difference.

### 4.1 EMBEDDED GUI

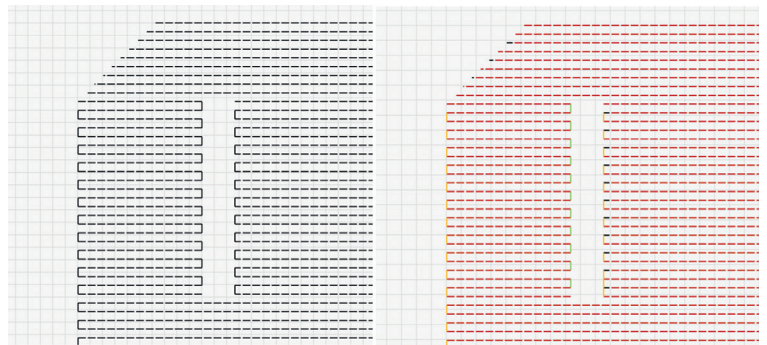
- Embed RAYGUIDE as part of your machine HMI which will be the master UI
- Use only the parts of RAYGUIDE GUI needed
- Save development time when using ready programmed controls
- Work with the widely used Microsoft WPF tool to embed RAYGUIDE

### 4.2 CUSTOMER PLUG IN'S

- Use RAYGUIDE as master UI and embed your individual machine feature controls
- Allows additional hardware controls
- Allows user specific graphic items

### 4.3 DASHED WAFER OPTIMIZER

- Feature to combine single line elements which represent a dashed line as one path
- Uses separate pen for dashed-line as for line change jumps
- Reduces the amount of delay time and graphic overhead



Regular DXF Import:  
25.654 single path`s

Dashed Wafer Import:  
965 path`s with dashed line pattern  
**Approx 60% faster processing!**

### 4.4 VIRTUAL CONTROLLER

- Design and evaluate laser process jobs using the emulated SP-ICE 3 scan controller without a physical scan controller connected

### 4.5 MULTI-POINT-EDITOR

- Ease & Improve field calibration with the powerful Multi Point Editor Tool integrated within RAYGUIDE