



## Radley steers their automotive EDI application into the future with new web technologies and a new UI

[Radley LLC](#) develops software products for manufacturing, warehouse management, and electronic data interchange (EDI). One of their strategic EDI products, iCARaS, is a Synergy-based release management and automated shipping solution for the automotive industry. iCARaS has a web UI with business logic and data access layers implemented in traditional Synergy. At the heart of iCARaS are thousands of Synergy routines for release accounting, demand reporting, shipping labels, and more.

### Challenge

iCARaS was introduced in the early 2000s and was developed using standard web technologies of the time (classic ASP and JavaScript, among others), along with access to Synergy logic via Synergex's xfServerPlus and xfNetLink COM products. However, as web development evolved and UI standards changed over two decades, Radley felt it was time to give iCARaS a visual refresh. Radley's investment in iCARaS's business logic made the application a leader in its industry, but with this long-term investment came some technical debt that they would need to overcome to position iCARaS for the future.

### Goals

To preserve their investment in iCARaS, Radley had several important goals for its revitalization:

1. Implement a cleaner, modern UI
2. Replace aging web technologies (particularly classic ASP and xfNetLink COM) with modern web technologies that would be viable for the long term and give Radley developers the flexibility to use their choice of modern front-end tools
3. Minimize changes to existing Synergy business logic routines.

“We really wanted a cleaner UI—that was the first goal—and more modern technology tools. And we wanted an achievable solution for this, one that we could use long term. Resource-wise, we don't have the capacity to completely rewrite the product frequently, so we needed a solution that uses the newest technology so that it can last a long time.

**Tim Bunker,**  
**Development Manager**

**RADLEY**

A CAI SOFTWARE COMPANY

## Solution

Bringing iCARaS forward into today's web development landscape presented a number of challenges, including a particularly problematic dependency. The UI had been built on classic Active Server Pages (ASP), Microsoft's original server-side scripting framework for creating dynamic web pages. Classic ASP was approaching end of life (EOL) and had the potential of becoming a security concern due to its age. However, updating iCARaS to a current version of ASP (i.e., [ASP.NET](#) or [ASP.NET Core](#)) didn't appear to be an option because xfNetLink COM, which was used to access the core Synergy logic, works with classic ASP, not ASP.NET. Due to this dependency, it appeared that a UI redesign would require either further investment in a technology reaching EOL (i.e., rewriting the UI with classic ASP) or the creation of an entirely new version of the application, one that would be based on modern web development technologies.

Given the challenges Radley faced with iCARaS, they carefully considered their options and were on the alert for new technologies and ideas. So, at the 2018 Synergy DevPartner conference in New Orleans, Radley developers in attendance listened with interest as Synergex introduced a new open source product built on industry-standard technologies and designed to bring Synergy data and logic fully into the modern web development landscape. The purpose of this product was the creation of flexible, scalable web services that make Synergy data and logic available through standard web service endpoints that are accessible to any modern front-end technology and are familiar to modern web developers.

This new product was [Harmony Core](#), and Radley immediately saw its potential, recognizing that it could be the key to extracting iCARaS from old technologies, bringing it into the modern web stack, and freeing it to be refitted with a new UI. Radley was particularly interested to learn about Harmony Core's Traditional Bridge component, which can be used to access the Synergy logic that iCARaS was currently accessing via xfNetLink COM.

With this realization, Radley began working with [Synergex Professional Services \(PSG\)](#) to lay the foundation for a revitalization of iCARaS. The project centered around a one-week on-site visit by a PSG consultant. Prior to the consultant's arrival, Radley provided PSG with the repository schema for their database and the XML export file for their Synergy method catalog. PSG then used these to set up some of the basic elements for a proof of concept (POC) that would be the starting point for PSG's onsite work with Radley's developers. When the onsite visit commenced, PSG demonstrated the proposed solution and began working with Radley developers to fill out the POC to demonstrate the specifics of the design and show how the design would meet Radley's requirements. The POC soon became the actual solution as Radley approved the design, and the solution progressed as PSG and Radley's developers worked out the details, many of which were routine for Harmony Core solutions, but some of which were unique to this solution.

Traditional Bridge's ability to tie our current application to the new technology was really the point that convinced management to say, 'OK, we can do this.... We can move forward with this solution.'

**Tim Bunker,**  
**Development Manager**

# RADLEY

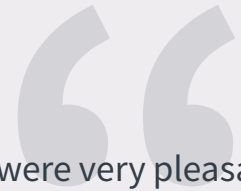
A CAI SOFTWARE COMPANY



One challenge that made this project unique was the requirement to support optional parameters for Synergy routines. Optional parameters are supported by xfNetLink COM, but not by other xfNetLink clients, so this functionality hadn't been required for other Harmony Core solutions. It was an entirely new requirement. Fortunately Harmony Core was designed for flexibility and extensibility, with the understanding that every Synergy-based application is different. In this case, Synergex's Harmony Core developers extended Traditional Bridge with a mechanism to pass and process complex parameters (including optional parameters) via JSON messages sent in HTTP POST requests, a mechanism based on the industry-standard [data transfer object pattern](#).

In many ways, however, the iCARaS project was similar to other Harmony Core solutions:

- **Harnessing the power of code generation** - The code generation functionality for Harmony Core is easily configured for most Traditional Bridge applications. iCARaS had a Synergy method catalog (SMC) for its use of xfNetLink COM, so PSG configured code generation to use the information in the SMC to generate web service endpoints and intermediate code layers. Harmony Core's code generation made it possible to quickly expose large numbers of existing traditional Synergy routines and set up client access to the new endpoints with minimal effort.
- **Extending Harmony Core for file access** - Another aspect of iCARaS that is common to many Synergy applications is the need to access different data files (e.g., different customer files) depending on which user is logged in. Harmony Core has an extension point (a class called FileSpecResolver) designed specifically for this functionality, which PSG used to set up custom rules for file access.
- **Using OData for queryable data endpoints** - For direct data access (access that doesn't involve a middle layer of Synergy logic), PSG set up code generation to create [OData](#) endpoints that expose data via URLs that support data queries. With well-designed OData endpoints, there's no need to code API endpoints for every possible variation of data access; web service consumers can simply query exposed data.

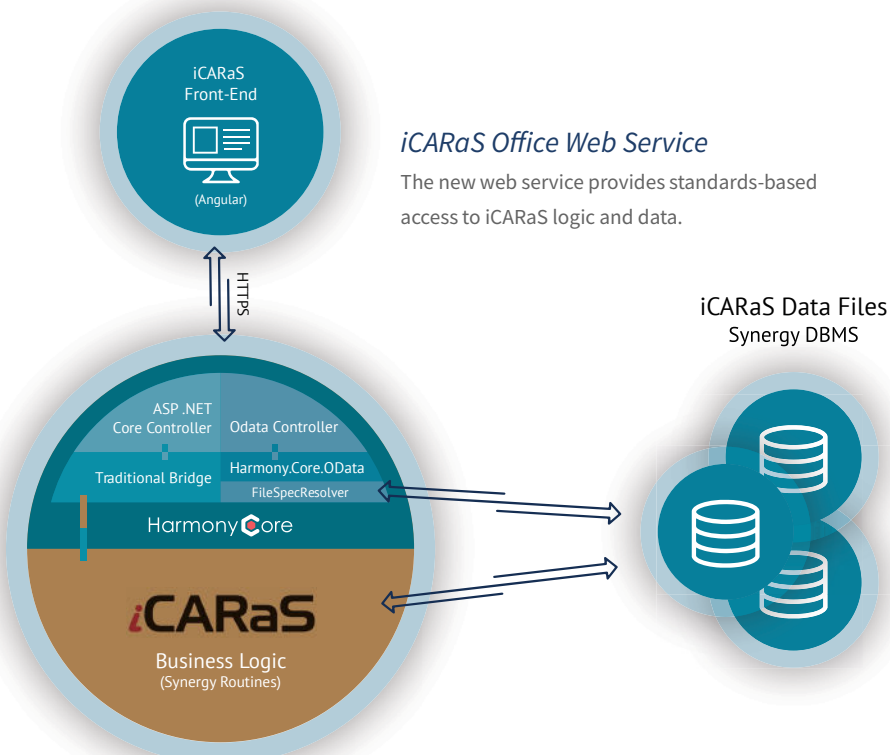


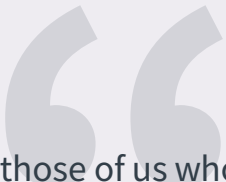
We were very pleasantly surprised with how well Traditional Bridge fit our needs to call our existing traditional Synergy methods since we have a very large code base of more than 2500 modules. We also really like the ability to embed this new implementation into the current solution for a seamless transition for our customers.

**Alex Nowak,**  
**Software Developer**

**RADLEY**

A CAI SOFTWARE COMPANY





For those of us who've been around for 20 years...it's really given us some hope for moving forward. For the last eight to ten years, we knew we had this looming technological debt that we didn't see a way out of, didn't see a way to eliminate. And now, with this project moving forward and the success we're having, we all...see that we have a future with this product without tearing it apart and starting over.

**Tim Bunker,**  
**Development Manager**

**RADLEY**

A CAI SOFTWARE COMPANY

With these and many other technical details resolved, the PSG engagement concluded. At this point, Radley developers had collaborated with PSG to establish all the essential elements of the solution. Together they had:

- Finalized the project design to handle the specific challenges posed by iCARaS, demonstrating that the design and the underlying technologies would meet Radley's requirements.
- Set up a web service that exposes iCARaS data and a good portion of its Synergy logic as OData endpoints.
- Configured code generation to simplify future development.

Additionally, Radley developers were able to see first-hand how the new system worked and how they could work with Harmony Core and the new web service to continue development on their own. And with the new web service in place, they were able begin work on a new UI for iCARaS.

## Outcome

The resulting Harmony Core solution placed iCARaS firmly in the modern web technology stack, enabling Radley's developers to create a new UI for iCARaS using their front-end tools of choice. For this, Radley chose the popular [Angular web application framework](#), along with components from [DevExtreme](#), DevExpress's JavaScript component suite. Additionally, the nature of Harmony Core and modern web development has facilitated a phased approach to development. Radley continues updating the UI one area at a time, introducing changes to iCARaS in a controlled way that customers see as UI improvements without the disruption of a large rollout.

Furthermore, Radley's collaboration with Synergex didn't end after the initial PSG engagement. Radley continued bi-weekly calls with PSG, enabling Radley developers to get general advice and assistance with troubleshooting. With these regular check-ins, Synergex was able to validate the approach to ongoing development, help Radley move the project forward, and address any new challenges that came up.

Thanks to Harmony Core and their partnership with Synergex, Radley has been able to leverage two-decade's worth of investment in iCARaS's logic by giving it a new UI that's appealing to customers and by replacing aging web technologies with industry-standard technologies that position iCARaS for future technological changes.

What's next? Once they finish with iCARaS, Radley plans to use similar solutions for their other two Synergy-based applications: iR\*EDi, an EDI product for automating business transactions, and then iSC, a supplier management solution.

Contact us to start your journey.

