



Take the Right Approach to Your TMS Selection





Peer, competitor, and market expert opinions can be a distraction when determining which technology is right for your organization, particularly when it comes to the crowded Transportation Management System (TMS) market. While the vast amounts of information out there about TMS can be educational, there are many red herrings which can divert away from finding the best tool to satisfy your unique business requirements.

Prior to starting any TMS selection, it is critical to set expectations and maintain discipline throughout the process. This document will guide you and your team through a list of high-impact steps aligned with a sound process, that if done well can be completed in **less than three months**.

Define the Scope

To get started, there are several critical decisions that need to be made.

- 1 What are we solving for?
- 2 Can we solve our challenges through process changes?
- 3 Do we already have a partner in place that can meet our needs if integrated better?
 - + If not, and we look internally, do we have the right skillsets on the team?
 - + And do they have the capacity to implement, test, learn, and operate on a new system?

Depending on the answer to these questions, a TMS might end up creating more problems and making it wise to investigate managed transportation providers. Regardless of what is being evaluated whether it be software or outsourced managed services, a well- defined scope must be established before taking steps forward.



KEY INSIGHT

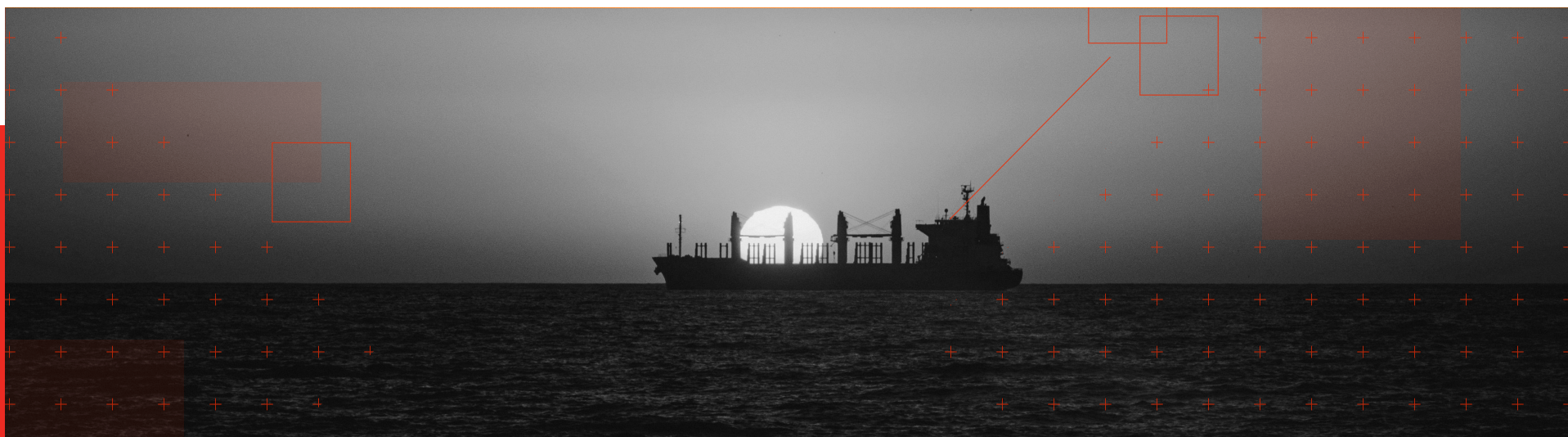
In the mind of most executives, software is a quick answer. In many of those situations they are right, but there are other solutions to explore.



Identify Your Vendor List

Once the scope is determined, identify software companies and products that exist in the market and could meet the requirements. Industry analysts like Gartner or Forrester have interesting frameworks to use as a starting point (key words = STARTING POINT). These should be referenced just like articles in trade publications, but not used as a way of down-selecting to a few candidates, nor should it be to reinforce biases.

TMS selection, specifically, has an added layer of complexity in that everyone calls everything a TMS. It is often misused as a blanket term for any software that touches transportation, when in reality there are a wide variety of sub-systems to consider. We typically consider TMS to be used for activities related to purchased transportation. The breakdown of the full TMS market into its specific niches is shown on the following page.



Transportation Management System

Tool used for purchased transportation, domestic and international, and managing planning, execution, tracking, and invoice/settlement activities. Typically, also includes optimization but not always.

- All Modes
- Contract / Rate Mgmt.
- Tactical Sourcing
- Tender/Booking
- Track & Trace
- Optimization
- Invoice Audit / Settlement
- 3 or 4 Tier Data Model

Global Trade Management

Software application used to manage global compliance requirements related to the master data (items, entities, etc.) and transactional data (shipments, filings, etc.).

- Customs Filings
- Item Classification
- Document Retention
- Denied Party Screening

Shipment Execution System

This solution is sometimes referred to as 'multi-carrier parcel manifesting' but can include other modes, but traditionally is limited to one-to-one relationship between order and shipment. This is often found embedded in ERP or WMS.

- High-Volume Rating Speed
- Robust APIs
- Manifesting
- Supports Mail
- 2 or 3 Tier Data Model

Vehicle Routing & Scheduling

Program used for carriers and private fleets to create street-level routes based on available resources (driver-pool, equipment, etc.).

- Static / Dynamic Routing
- Turn-by-Turn Navigation
- Scenario Analysis in Execution Env.

Fleet Maintenance System

System used for managing a variety of physical assets (tractors, trailers, drivers, technicians, etc.) during execution of daily workflows.

- Work Order Mgmt.
- Backhaul Mgmt.
- Driver Messaging
- Asset Tracking

Internet of Things (IOT)

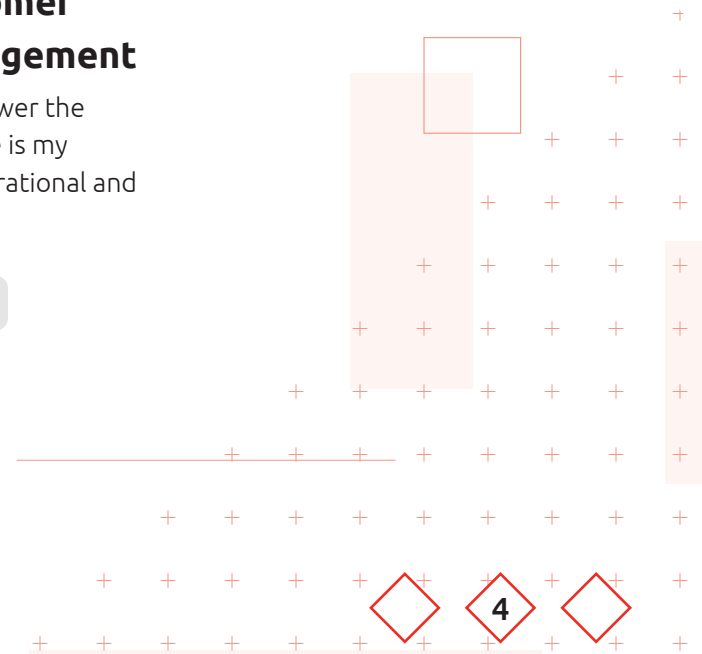
Software and hardware tools used for monitoring resources for safety, compliance, and location. These are sometimes referred to as Asset Systems.

- Electronic Logging Devices
- Temperature Recorders
- Theft Prevention

Visibility & Customer Experience Management

This tool typically will answer the common question "where is my truck?" creating both operational and customer service value.

- Real-Time Status Updates
- Customer Portal Access
- Performance Tracking





When researching potential software vendors, make sure you understand their actual business model. In transportation specifically, it is common to see companies go-to-market as software companies but also offer third party logistics (3PL) services like truckload brokerage or managed transportation. Unless well researched, long lists could very easily include quasi or direct competitors! To prevent this, it is best practice to lookup vendors on SAFER (safer.fmcsa.dot.gov) to confirm. This upfront research over the first week or two will help you develop a long list of qualified vendors to start engaging in the selection process. Over about three weeks, the long list is sure to turn itself into a short list based on initial vendor response. Some common reasons vendors exclude themselves include their inability to turn around a mutual NDA, unwillingness to participate in a disciplined RFP process, or the perception that their product is not a reasonable fit for your business. A short list should be four or five vendors, but you could do it with three or six.



Apply Functional and Technical Requirements



While conducting initial research and creating a vendor short list, business users should begin drafting requirements. There are two types of requirements that must be accounted for: functional and technical requirements. Functional requirements should serve as a list of needs that must be met for users to effectively and efficiently complete their day-to-day tasks. Technical requirements may require input from IT teams and should include needs related to security, performance, database architecture, data integrity, etc. There are a few things to keep in mind when writing functional requirements:

- + They should NOT be prescriptive, meaning they need to tell WHAT the requirement is, not HOW it is to be handled/implemented by the software provider.
- + They should NOT be based off legacy system functionality or current state processes, as future state processes should be documented, and new systems are likely to offer a wider range of capabilities.
- + They should be written in a format so that vendors can respond in a yes/no manner to each of them, starting with the phrase, “The system should allow us the ability to...”.

- + Alternatively, vendors could be permitted to break their response down into more detail by indicating whether they meet the requirement with base functionality, if it can be met with configuration, or if it can be developed.

The last bullet is important because these requirements should be distributed to short listed vendors in the RFP. The expectation is vendors will respond to each requirement indicating whether they can or cannot meet it, which will be useful when scoring vendors later in the process.

Finalize Your RFP

Since the business team should already have a head start on requirements, these can be dropped into the RFP once complete. The RFP should also include an RFP overview document, a pricing request, demo scripts, and data samples. The RFP overview document is exactly what it sounds like: educate the vendors on the scope, providing background on your company, the reason you are sending out the RFP, and expected timeline for response, demos, decision dates, etc. As mentioned, a pricing request should be distributed and should serve as a template for vendors to populate their recurring and one-time fees.

Establish Demo Scripts



If the software vendor determines the structure and content of the demonstration, you will not keep the process focused on your business case, use-cases, and specific requirements; rather you will see the wiz-bang unicorn-catching flash-in-the-pan sizzle, but not the steak. The only way to see available and relevant functionality is by prescribing which screens they show and buttons they press.

Based strictly on requirements, demo scripts consist of a step-by-step guide for vendors to follow along with when presenting and demonstrating their software. For example, in a script related to planning you may have a requirement like “demonstrate the ability to manually combine orders and/or shipments to build consolidated loads.” Separately, you may ask the vendor to demonstrate how that step could be automated. For the best demo experience, it is suggested that sample data is provided to the vendors ahead of time, so they are working with orders and shipments containing characteristics that are realistic for your business to handle.

If the decision is made to invite the vendors for demos, it will be necessary to have at least one demo prep session a week before the official demo day.

These sessions should serve as Q&A sessions for the vendor, allowing them to ask questions about the wording and direction of the scripts, and get an understanding of what your team expects to see.

Back to discipline. Demo day will be a disaster in the absence of discipline. All demo participants should be made aware of the demo goals and expectations before starting, especially as it relates to scoring. Demos are most effective when the vendors are scored against their performance on the demo scripts. This ensures each vendor is compared in an apples-to-apples manner, but even better, it helps the participants stay engaged.

From a quantitative standpoint, requirements and demo scores can be easily summarized and reviewed. Perspectives certainly need to be considered from a qualitative standpoint as well, though, to come to a vendor decision.

Time to Contract with Your Vendor



After some debate and internal negotiation, a vendor decision can be made. That's great but you're not quite done yet...it's time to negotiate. When it comes to negotiations, it is always best to base them on license and subscription fees. Proper negotiations on these fees can provide much greater long-term benefit than negotiations on implementation/one-time costs. For example, let's say the TMS recurring costs are based on freight under management (FUM) (broken out into a tier structure). It would benefit your company to try negotiating the ranges for each tier based on expected growth and timing for your business. Outside of cost negotiations, it is necessary to thoroughly review the terms and conditions provided by the vendor and negotiate where necessary. Some things to look out for include the hosting strategy that the vendor leverages, and their data retention and escrow policies.

Another area to investigate is the type of environment provided. It is not always clear or may not even come up, so be sure to request a breakdown that includes cost. There are several types of environments to consider, although not always necessary: development, testing/QA, training, staging, and production.

- + **Development:** Configuration occurs and in customization scenarios this is where the code is developed
- + **Testing/QA:** Code is tested by users to ensure there are no issues/bugs
- + **Training:** Users practice/learn how to use the system and understand workflows
- + **Staging:** Code is held in case there is a desire to wait before making it live or promoting to another environment (i.e., developers have written new code to support the requirements of new business/ an acquisition, it has already been tested, but there isn't a need to go-live with it until the new business is officially onboarded)
- + **Production:** Active code and transactions (live user environment)



If you are dealing with an on-premise solution, chances are you will have access to four or five of these above. In a true SaaS world, sometimes a production environment is all that is needed. For large-scale TMSs, though, it is suggested you have at a MINIMUM, both testing and production environments.

Quick Check-In with the Broader Strategy

Don't forget to revisit the strategic goals you set early on to make sure all boxes are checked. It should be made a priority to consider your integration strategy as well before engaging and making a decision to ensure your selection will align.

- + Will you need to invest in an integration platform (iPaaS) in order to support all integrations with third-party applications?
- + Have you properly accounted for all third-party systems that are required? A PC Miler license, for example, is almost certainly needed to support a TMS.

Each of these considerations is necessary to think of in terms of not only your strategic goals, but finances as well. A total cost of ownership (TCO) should be constructed to reflect the costs associated with this transformation over the next roughly five years.

To offset the costs, business case/benefits should also be considered to calculate the overall return on investment (ROI). Based on the outcomes of these analyses, your team can reasonably estimate the implementation timeline/roadmap with phases that directly correlate to business case capture.

Selecting the appropriate vendor for your business can be a daunting task, but when done correctly it makes all the difference. The tips and tricks outlined in this paper will set you up for success, but execution will be dependent on an engaged and disciplined team throughout.

Forerunners third-party presence and expertise in selections has proven to help our clients maintain accountability, and when coupled with our tested methodology, the right vendor is sure to be selected. Learn more about Forerunners capabilities at:

<https://forerunnersconsulting.com/>

