

TOS Implementation Best Practices

The Navis Professional Services team has been on the front line of container terminal operating system (TOS) implementations for more than 25 years. In the last few years, Navis has implemented N4 at more than 80 terminals around the world, which has included replacing in-house systems, other commercial systems, deploying at Greenfield sites or migrating customers from SPARCS/Express to the N4 platform.

Most projects go according to plan with little or no disruption to productivity; however it is not uncommon for some projects to bring their own set of unique challenges. In these instances, Navis' Professional Services team works hand-in-hand with customers to overcome potential issues and ensure that the TOS operates as designed. As a result, the collective experience gathered from these projects has provided Navis with valuable insights into the best practices for TOS implementation in all types of situations.

At Navis, we view ourselves as a technology partner for our customers, providing the most experienced professional services team available in the industry. While each new implementation project may have its own set of unique circumstances, the common thread of success among them all is the level of partnership, trust and implementation readiness required by both Navis and our customers. We understand the challenges of implementing multiple systems and completing the vision of our customers to reach a new level of efficiency and productivity for their terminal operations.

New customers implementing N4 consistently ask our Professional Services team what insights can be provided to ensure successful implementation projects. The following document reviews ten recommended best practices that have helped our customers to significantly improve the success of their TOS implementation process and greatly reduce risks to their operations.

The 10 recommended TOS Implementation Best Practices that will be discussed in this document include:

- 1. Build a Dedicated Project Team**
- 2. Use a Phased Approach to Technology Changes**
- 3. Minimize the Customization of the New System**
- 4. Ensure IT Infrastructure Readiness**
- 5. Develop a Comprehensive Testing Plan**
- 6. Plan your Data Migration Process Early**
- 7. Develop a Systems Monitoring Plan**
- 8. Develop an Operational Recovery Plan**
- 9. Create a Communications Plan**
- 10. Establish a Systems Training Plan**

Table of Contents

| | |
|---|----|
| Recommendation 1: Build a Dedicated Project Team | 3 |
| Recommendation 2: Use a Phased Approach to Technology Changes..... | 3 |
| Recommendation 3: Minimize the Customization of the New System | 4 |
| Recommendation 4: Ensure IT Infrastructure Readiness | 4 |
| Recommendation 5: Develop a Comprehensive Testing Plan | 5 |
| Recommendation 6: Plan your Data Migration Process Early | 6 |
| Recommendation 7: Develop a Systems Monitoring Plan..... | 6 |
| Recommendation 8: Develop an Operational Recovery Plan | 7 |
| Recommendation 9: Create a Communications Plan for Key Stakeholders | 8 |
| Recommendation 10: Establish a Systems Training Plan | 8 |
| TOS Implementation Best Practices Checklist | 10 |

Recommendation 1: Build a Dedicated Project Team

The structure and focus of a container terminal's implementation team is an essential element that can make or break the success of a project. The size and design of a terminal's implementation team can vary considerably, from a full-time dedicated team of 40 to a part-time team of two people. In general, teams have to be large enough to support the project, and a dedicated and empowered project team is more successful than a team that combines day-to-day and project roles. Navis can assist in identifying the team's structure so that it best matches the situation and will drive success.

Senior terminal leadership should be a critical component of the implementation team. Implementation projects are change management projects and each project needs to have a senior leader from the terminal as a visible champion and sponsor of the project to help ensure a smooth implementation. This can be the terminal's CFO, COO or even the CEO. Terminal operations management, especially front line superintendents should be involved in the day-to-day project as well. Yard, marine, gate and rail teams all need to be involved, though not necessarily all in a dedicated fashion.

It is also critical that the terminal's project team be a separate, dedicated team. The project team needs to have end-users involved in the final testing and be aligned internally to the project or business plan they are pursuing.

Finally, the team should also include a group to oversee the systems integration. Implementation of N4 is rarely done in isolation. There are typically other new systems that must be implemented as well as legacy systems, which need to be re-worked and re-integrated with N4 in order to deliver the overall solution. This process typically takes dedicated resources. Finally a team of external consultants, solely focused on the terminal roll-out, can substantially improve the speed and quality of the implementation by complementing the experience of the internal team.

After the go-live, operations and IT often return to their 'silos' and previous roles, which invariably creates potential issues in identifying and resolving issues quickly. The best organizational model is one where a terminal retains a combined IT/operations team approach with joint responsibility for the effective on going use of the systems.

Recommendation 2: Use a Phased Approach to Technology Changes

Most terminals run on multiple, interacting and integrated systems, including Position Detection Systems (PDS), Quay Crane Optical Character Recognition (QC-OCR), Gate Operating Systems (GOS) and Enterprise Resource Planning (ERP), to name a few. Going live on a new TOS and other systems at the same time is very complex and adds a level of risk that can be averted by phasing in the new systems after the initial TOS goes live.

Implementing a new GOS, PDS, and automated equipment at the same time as the TOS may be mandatory in a new facility, but existing terminals may have the option of doing this in stages and should consider doing so. There doesn't have to be a long period of time between these phases, but introducing incremental change can help mitigate change management risk and help ensure the terminal operators have time to adapt, master new processes and are better prepared to quickly handle any operational issues.

With a phased approach, the implementation project establishes the basic systems before implementing more complex and additional optimization functionality.

In addition to a phased approach, terminals implementing complex software integration projects should try to plan their go-live activities during a time of lower operational activity. Terminals that can proactively control operational volumes in the weeks and months before or around the scheduled go-live can mitigate any initial decline in productivity as users become used to new systems and processes.

Another option to mitigate risk that is often mentioned or tried is to run the live and test systems in parallel. This, however, can create additional complexity and risk and is therefore not recommended. Apart from the expense to maintain two or more systems in parallel, the behavior of the parallel systems will vary in small but operationally significant ways.

Finally, an implementation readiness checkpoint is recommended prior to the go or no-go decision. This allows time for a comprehensive go-live readiness audit before the final stage of the go-live is implemented. After the go-live decision, IT infrastructure should not be changed as they need time to be adequately tested prior to go-live

Recommendation 3: Minimize the Customization of the New System

An essential implementation best practice is to align your terminal's business processes to the capabilities of the solution as much as possible. In projects where the terminal has deep knowledge of an existing TOS, and little knowledge of N4, it is tempting to heavily customize N4 until it matches the old existing TOS. This is NOT recommended. Enhancement of TOS software is always enticing in any new system implementation; however, excessive customization also introduces a great deal of risk.

New software customizations require significant additional testing overhead for the project team. Because they must be tailored to a specific site, these customizations don't have the benefit of having been "road tested" by other customers in the past, and as such are more likely to produce potential problems. Customizations also introduce unnecessary complexity into the software. While Navis QA has tested the base functionality of the software in great depth, it is often difficult or impossible to anticipate the impact that extensive customization will have on the behavior of the base software.

It often takes extensive planning and deliberate buy-in from key stakeholders at the terminal to avoid extensive customization of the TOS. To achieve this level of internal support, it is important to ensure that the key stakeholders understand N4 software fully by engaging them in super-user trainings, ideally before business analysis commences. Many customers also find success in implementing a business case justification process to ensure that any requested software changes are assessed for business benefit, cost and risk before being approved and committed to the development stream. The goal is to get the entire project team focused on matching the business processes to existing software functionality, rather than the other way around.

Finally, getting commitment early on from all key stakeholders to the concept of changing existing business practices as part the implementation is critical to successfully avoiding risky software customization. Many times, business process changes are perceived as more difficult to achieve than software changes, and changing a business process rather than changing software can be thought of as a step backward by some in the organization. Also, business process change sometimes was not expected as part of the project.

Recommendation 4: Ensure IT Infrastructure Readiness

The IT infrastructure is the backbone to any terminal operation. It is a critical component for a successful implementation and should be tested and evaluated as to its readiness to support the new systems. As such, all aspects of IT infrastructure and how it impacts the operation of the container terminal are fundamental to implementing new systems.

From our experience with TOS implementations around the world, there are four key things all terminals should do to ensure IT infrastructure readiness:

1. **Make sure there is a comprehensive understanding of the terminal's IT architecture**, including the hardware (servers, computers, and wired and wireless networks), applications and container handling equipment (CHE). Develop and maintain an architectural design model that reflects a clear understanding of all of the components of the IT architecture and how they

connect and interact with one another. If done correctly, this will help plan for system outages and process changes.

2. **Verify the IT infrastructure is sized such that it is ‘fit for purpose’** and will support the business requirements of the terminal. This is especially important when considering integrated solutions such as the TOS and one or more sub-systems can cause problems if the messaging between these systems is not working seamlessly. Even when system(s) are working ‘as designed’ individually, operational problems can occur when systems start working together. This can create gate congestion, yard congestion and significant operational exceptions.
3. **Make certain the IT infrastructure supports the terminal’s requirements**, including the individual applications and the interaction between applications, infrastructure and the CHE. The majority of configuration challenges typically occur with databases, wireless networks and TOS configuration (nodes, etc.).
4. **Confirm adequate IT change control processes are in place well before the go-live.** These processes should be robust, agile and thoroughly tested. They should cover all aspects of the technology changes, including configuration of applications, networks, servers and databases. An audit trail on what changes have been made is often the key to understanding downstream issues that may arise as a consequence of these changes. The changes should also be added to the architectural design model.

Recommendation 5: Develop a Comprehensive Testing Plan

One of the main objectives of system testing is to help determine whether the TOS will work in combination with new processes, procedures, other systems and users, in real-world scenarios. The test strategy must carefully outline all aspects of the business processes that need ‘testing.’ Too often, testing is either de-emphasised or targeted too narrowly. Basic “kicking the tire” tests are performed by running very typical examples, or tests are limited to “happy day” scenarios. A formal test plan that describes what needs to be run is important. In addition, these tests should be run by a team of individuals that come from different areas of the operation.

Navis recommends comprehensive and advanced test plans to mitigate risks in most TOS and integration implementations. A test plan should include a suite of specific test cases including acceptance criteria for each case, a written plan for handling defects and how to handle exceptions. Developing an automated testing strategy, in combination with emulation testing, is an effective way to reduce the on-going cost of regression testing.

Investment in a structured test management process with tools, test cases and a defined a data strategy is also recommended. An appropriately designed and maintained test plan allows the operations team to sign off that the vast majority of business and operational scenarios are appropriate and have been proven to work prior to go-live.

Finally a comprehensive test plan should also include:

1. **Operational Performance Testing** to test and validate that the automated decisions the TOS is making is optimally configured for the specifics of the terminal.
2. **Systems Performance (Stress) Testing** to test representative and exceptional volumes of transactions in order to exercise transaction volumes being imposed on the infrastructure and operational transactions processed by the TOS configuration.
3. **Systems Integration Testing**, including all major sub-systems, should be tested together (end-to-end testing) to ensure that the processes, systems and people working together in a live operational environment can do so successfully.

Recommendation 6: Plan your Data Migration Process Early

The complexity and impact of data migration can cause data issues in the TOS. Incorrect data can cause major operational problems until the data is cleaned up or the affected transactions are passed through the system.

Migrated data should be thoroughly “exercised” during the testing phase. Based on our experience, it is essential that implementation teams test all combinations of container handling actions including gate transactions, yard to yard moves, vessel moves and rail. This process will help identify any data-related problems that need to be resolved prior to go-live.

When planning your data migration process, it is essential to do the following:

1. Migrate data frequently through your implementation process to test the process and refresh your data sets where it makes sense.
2. Avoid migrating historical data unless it is operationally required. Archive your old system data so that it can be accessed through alternative means.
3. Ensure your data cleanup process is extensive and that you use your testing procedures to check that the migrated data can be processed correctly.
4. Consider migrating data for containers on-site and then reprocess EDI messages (e.g. EDI350, UIIA files, Fleet Files, BAPLIEs) after the inventory has been loaded.

Recommendation 7: Develop a Systems Monitoring Plan

Having a limited or no monitoring solution in place for a terminal's IT infrastructure represents a significant risk. There are a several systems that can be implemented with the N4 deployment that will assist with systems monitoring.

Monitoring tools for N4 components that are Java processes, such as SPARCS N4, bridge, and ECN4, Java Management Extensions (JMX) are enabled by default. JMX enables you to monitor memory and thread usage. In addition, it provides other diagnostic options, such as a detailed view of all available MBeans.

N4 also provides a plug-in for Hyperic HQ, an open source, enterprise level systems monitoring tool. The implementation of Hyperic is recommended for all N4 installations (unless customer has an equivalent or better monitoring tool implemented). Hyperic enables the creation of dashboards that show the health of all of the major systems and the interfaces between these systems. Any drops in systems performance can then be identified more easily and managed proactively. Terminals can then collect basic metrics for N4 components, including XPS and the second level cache (Hazelcast). The Hyperic plug-in can also be used to develop more detailed dashboards, parse log files and to fully utilize N4 MBeans for customized alerts and reports. Hyperic can also be used to improve the ‘restart’ time of many applications, which is particularly useful in large, complex N4 installations.

The N4 Database Performance Pack provides improved visibility into the database setup and performance. As an advanced troubleshooting tool, the performance pack interfaces provide diagnostic information, identify performance problems, and verify database configuration.

Having a limited or no monitoring solution in place for a terminal's IT infrastructure represents a significant risk to system downtime and outages. There are a several tools that can be implemented with the N4 deployment that will assist with systems monitoring. Monitoring tools for the JAVA processes (N4, bridge and ECN4 JAVA Management Extensions (JMX)) are enabled by default. JMX enables you to monitor memory and thread usage. In addition, it provides other diagnostic options, such as a detailed view of all available MBeans.

N4 also provides a plug-in for Hyperic HQ, an open source, enterprise level systems monitoring tool. The implementation of Hyperic HQ is a pre-requisite for all N4 installations (unless customer has an equivalent or better monitoring tool implemented). Hyperic enables the creation of dashboards that show the health of all of the major systems and the interfaces between these systems. Any drops in systems performance can then be identified more easily and managed proactively. Terminals can then collect basic metrics for N4 components, including XPS and the second level cache (Hazelcast). The Hyperic plug-in can also be used to develop more detailed dashboards, parse log files and to fully utilize N4 MBeans for customized alerts and reports. Hyperic can also be used to improve the 'restart' time of many applications, which is particularly useful in large, complex N4 installations.

The N4 Database Performance Pack provides improved visibility into the database setup and performance. As an advanced troubleshooting tool, the performance pack interfaces provide diagnostic information, identify performance problems, and verify database configuration.

Recommendation 8: Develop an Operational Recovery Plan

As we have been discussing, the implementation of a new TOS and integrated systems is a complex and can be a challenging proposition. No matter how much your systems are tested before the go-live, there is always the risk of unforeseen and unanticipated events that can cause system outages. As such, it is critical to have an operational recovery plan in place to help restore the system as quickly as possible.

Navis recommends putting the following operational recovery processes in place prior to go-live to:

1. Ensure you have clear ownership and escalation procedures in place that will help you quickly and efficiently identify, report, and resolve system issues that can result in operational problems.
2. Ensure systems monitoring is in place to determine whether or not the system and which part of the system, is either down or is in the process of failing. It is often possible to identify system failures "in progress" allowing you to take the necessary action to avoid the failure or start the recovery process. This helps to minimize the impact on the operations. Navis recommends using Hyperic HQ or another enterprise monitoring application for ongoing system monitoring. Examples of this may be monitoring the database performance, queue sizes, memory usage of N4 or other systems.
3. Establish a clear set of recovery procedures for when a system (or parts of a system) may be down. Your ability to get your systems recovered quickly coupled with your ability to "reset" your operations is critical. This may involve setting up automated resetting or rebooting of some of your systems in order to help reduce systems downtime.
4. Determine procedures for how to bring operations back up to speed as quickly as possible. You'll need to understand the processes and sequencing for stopping, reconciling and restarting operations to ensure that you remain in synch with the information in your business applications. Recovery procedures can vary dramatically depending upon which systems and operations are running.
5. Have a communication plan in place for internal and external stakeholders when dealing with system outages.
6. Ensure an adequate Disaster Recovery Strategy is ready, tested and in place prior to go-live.

Recommendation 9: Create a Communications Plan for Key Stakeholders

A TOS has a substantial impact on a large number of people who work at the terminal or rely on the terminal operation for their businesses. It is therefore important to have a solid communications strategy and plan in place that addresses the current and potential concerns of all stakeholders. For example,

- Labour may have constraints that need to be considered in the TOS rules
- Truckers need to be aware of changes to gate procedures
- Superintendents and planners may need to have revised approaches to scheduling and planning
- CHE operators may need to have specific training on the use of the new system

Having a set of plans focused on addressing how the process and TOS changes are identified, communicated and implemented in the terminal can be a useful supplement to the overall project plan. A helpful framework to organize this work stream, and to help ensure that a complete plan has been developed, is the “ADKAR” framework, from Prosci, a change management institute. This framework includes five parts:

1. **Aware:** Are the constituents within the terminal community aware that a change is planned?
2. **Desire:** Do the constituents have the desire to support the change?
3. **Knowledge:** Do the constituents have the knowledge of how to support the change?
4. **Ability:** Does the constituency have the ability to change?
5. **Reinforcement:** Are the learning and financial incentives in place to reinforce the change once it is live?

This framework can be used to develop a communications plan to notify both internal and external constituents of the technical upgrades that are happening at the terminal. Terminals that invest in a communications plan in advance of the TOS go-live will significantly increase the likelihood of a successful project implementation.

Recommendation 10: Establish a Systems Training Plan

The implementation of a new TOS involves a significant amount of change adoption throughout the operation. It is therefore critical to have all the users of the system up-to-speed on how to use the system prior to go-live to ensure a smooth transition. In order to make sure things go smoothly requires focus in advance on change management, communications, and training.

Training, both on the TOS application and on support systems, is one of the most important aspects of managing the change that comes with the implementation. There are several factors that must be considered when establishing a comprehensive program.

1. Labour and operational constraints need to be fully thought through and understood prior to go-live.
2. End-user training on the new systems needs to be planned thoughtfully and should include exception handling and trouble management processes (not just “happy day” scenarios).
3. The timing of end user-training should be considered carefully. It needs to be close enough to the go-live to ensure that users don't forget how the system works before needing to use it, but far enough in advance to allow time for new concepts and ideas to sink in and be understood. One

good strategy is to stage or phase your training such that users receive initial core training quite a bit in advance of the go-live, and then to follow up with a refresher training closer to go live day.

4. Provide a training environment (including desks, computers, etc) where trained users can re-run training exercises or play with the system in order to “bake in” lessons learned.
5. You likely will be training your users not just on the ‘system,’ but on new operational business processes that will be changing as well. Make sure these new processes and any operational implications are all fully explained and understood by the end-users.

Presenting the material in a context of cause and effect will also help end-users to understand and the rationale behind why N4 does things ‘differently’ from what they are used to and the impact of their actions on the new system. This can present a particular challenge when a terminal is implementing N4 to improve system planning processes (eg, vessel, yard, rail) or when N4 is being implemented to ‘automate’ and improve a manual planning process.

Stakeholder training beyond the terminal staff is also important to keep in mind as a terminal moves through the implementation process. Insufficient education and communication with the terminal community—such as truckers and shippers, for example—is critical for smooth implementation. Process and operational changes should always be documented, explained, maintained and readily available to all key terminal stakeholders.

User Acceptance Testing (UAT) can be an effective means of introducing a broader group of users to the new systems and processes. This can assist in validating the effectiveness of training that may have been given and may help determine whether your users (and maybe your new processes) are ready for the transition.

TOS Implementation Best Practices Checklist

In summary, here is a checklist of Best Practices that will help to keep an implementation on track for success.

- Build a Dedicated Project Team
- Use a Phased Approach to Technology Changes
- Minimize the Customization of the New System
- Ensure IT Infrastructure Readiness
- Develop a Comprehensive Testing Plan
- Plan your Data Migration Early
- Develop an Operational Recovery Plan
- Develop a Systems Monitoring Plan
- Have a Communications Plan
- Have a Systems Training Plan

Note: Navis Professional Services partners with customers implementing N4 to provide detailed checklists to assess N4 Go-Live readiness and to identify critical issues that should be addressed in order to proceed with a scheduled go-live. This process helps to ensure a successful go-live with minimal operational disruption. See your Navis Professional Services representative for more details.