The Seven Sources of Problems: Part 1

How To Eliminate Problems Before They Impact Your Business

Robin Yearsley
The **S7VEN** Sources of Problems

How to Eliminate Problems *Before* they Impact Your Business

This Edition is Presented By:

[ITIL Training Zone](https://www.itil-training-zone.com)

Your Place. Your Pace. Your Progress.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without prior permission from the author except where explicitly authorized to do so within this publication and within the strict terms defined. This publication must not be changed, adapted or otherwise modified when complying with these terms.

Limit of Liability / Disclaimer of Warranty: The Author has used his best endeavors in preparing this publication. The Author makes no representations or warranties with respect to the accurateness, completeness and suitability of the contents of this publication and specifically disclaims any implied warranties or merchantability or fitness for purpose. There are no warranties which extend beyond the text of this paragraph. The accuracy and completeness of the information contained within this publication and the opinions herein are not guaranteed or warranted in any way. The Author will not be liable for any loss of profit or any other consequential, special or incidental damages.

*ITIL is a Registered Trade Mark, and a Registered Community Trade Mark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.*
# The S7VEN Sources of Problems

## CONTENTS

**Introduction** ........................................................................................................ 5

**Controversial Statement** ........................................................................................ 10

**Introducing The S7VEN Sources of Problems Model** .............................. 11

Blame Statements........................................................................................................... 12
Why Blame Prevents/Hinders Problem Elimination..................................................... 13
Eradicating A Blame Culture......................................................................................... 14

**Source #1 - Acceptance into Production** ................................................................. 15

The Model....................................................................................................................... 15
Four Essential Ways to Manage ‘Known’ Errors............................................................ 18
Single Points Of Failure Analysis.................................................................................. 19
Steps to Eliminate SPOFs............................................................................................ 20
Supportability – Key Actions....................................................................................... 21
Testing Evidence............................................................................................................ 22
User Acceptance............................................................................................................ 23
“Does Not Meet (Business) Requirements”.................................................................... 23
Education / User Knowledge....................................................................................... 24
Overcoming Spreadsheet Challenges......................................................................... 24
Two Guiding Principles............................................................................................... 25

**Source #2 - New Changes** ..................................................................................... 26

The Model....................................................................................................................... 26
The “Failed Change” Chain......................................................................................... 27
Lack of Appropriate Approval...................................................................................... 28
Factors that Hinder Effective CABs............................................................................ 29
Poor Testing.................................................................................................................. 29
Poor Implementation Planning.................................................................................... 31
Poor Execution of the Implementation Plan................................................................. 32
How to Create a Great Implementation Plan............................................................... 32
The Impact of another (concurrent) Change............................................................... 33
Your Acceptance into Production Criteria.................................................................. 34
Playing it Tough............................................................................................................ 35
The “Yes, However…” Technique.............................................................................. 35
The S7VEN Sources of Problems

Source #3 – Upgrades and Patches

The Model
Being Aware of a Patch or Upgrades Actual Existence
Finding the Right Balance between ‘leading edge’ and ‘Bleeding edge’
Maintaining Your Organizations Overall Adherence
Having the Physical Capability to Actually Test
Managing Change Through Your Change Management Process
What to do if you are Forced to Accept a new Upgrade/Release
The Resources and Skills You Absolutely Require

Source #4 – Vendors

The Model
Know your “Vendorscape”
Vendors are Human too
The Quality and Relevancy of Contracts
Three Stages of Vendor Enlightenment
SLA’s and Vendor Relationships are the Major Differentiating Factor
Implement a Vendor Scorecard Programme
How to Create a Simple, Low Over-head Scorecard That Works
Integrate Deeper with Vendor’s Processes
Involve Vendors More

Source #5 – User Error

The Model
User Education – or Rather the Lack of it
Purposeful Documentation
The Delivered System “Fails to Meet Specified Requirements”
Validation, Robustness and ‘Self-Healing’
Five Things That Delivered Great Results

Source #6 – How Production is Executed

The Model
Stop/Start Recovery Procedures
Automated Tools and the Ability to Handle ‘Out of Line’ Situations
Impact of Changes on Production Schedules
Human Error
The S7VEN Sources of Problems

Source #7 – Failures........................................................................58

How To Execute Your Problem Elimination Programme........62

Additional Resources.....................................................................66
Introduction by the Author

This eBook was written for all the hard working, conscientious Service and Support professionals who aspire to deliver quality services to their customers every day.

Within the context of ITIL, the primary audiences who stand to gain the most value from this eBook are: Service Desk Team Leaders, Incident Managers, Problem Analysts, Problem Managers, Change Managers, Availability Managers, Service Level Managers… in fact anyone who’s involved or needs to improve the {Incident ➔ Problem ➔ Known Error ➔ Change} chain.

If you want to learn new and effective ways to identify and eliminate the Problems you experience in your Production environment – then this eBook is definitely for you.

This eBook will show you the often invisible sources of problems within your production environment and offers you time proven, practical advice and guidance on how to effectively and systematically eliminate the sources of those Problems. ‘Prevention’ is not only better than ‘cure’ – but is cheaper, easier and far better for your business lines and customers.

Problems cost businesses millions of dollars each year in terms of people’s time, wasted effort and lost productivity. Problems also bring to mind a very unwelcome association of ‘negativity’ and gives customers something real to ‘negotiate’ with. If you then go on to consider repeat problems that happen again and again or even very serious problems that happen just twice – then you can basically multiply this ‘negativity’ effect tenfold. So prevention coupled with elimination forms an essential aspect of managing Production environments today.

I understand you are a busy professional - so why should you read this particular eBook? Why should you also forward this book to your colleagues in other areas of your organization?

Well, because this eBook is a compilation of my personal experience of over five years working in a rapidly changing, dynamic and fast growing global corporation. It doesn’t matter which one or where it was located. I can vouch for the fact that this global corporation was massive with millions of customers worldwide.

I really believe that if I can share my personal experiences with you in terms of how I, combined with my excellent team of dedicated service professionals, eliminated thousands of Problems quickly and successfully over those five years – then there should be something in here that strikes a chord with you.

If this eBook gives you just one new idea that you can take away and use successfully to eliminate time wasting and costly Problems in your business – then I have done my job and creating this eBook will have been well worth it.
Thinking about it this eBook actually represents the sum total of literally thousands of hour’s worth of investigation, analysis and elimination efforts.

This eBook is full of ideas but factual. It’s not theoretical, academic or full of consultancy speak. It is deliberately meant to be straight to the point, pragmatic and present with you lots of ideas, approaches and activities that you can replicate within your organization.

Over the next 60 pages I want to share with you many of what I call ‘golden nuggets’ of know-how that I actually spent about 50 hours a week for 5 years experiencing first hand.

If you consider these numbers for a moment – it actually cost my former employer a LOT to enable me to gather, learn and implement these Problem elimination activities.

I can even roughly calculate the cost for you: -

1. Calculate the number of man-hours I worked over the five years

   50 hours a week × 48 weeks a year × 5 years

   = 12,000 man-hours

2. Calculate my cost per hour (including pension, bonus, taxes etc): -

   = $80 per hour

3. Calculate total man-hours multiplied by hourly cost: -

   12,000 man-hours × $80 per hour

   = $960,000

Wow - that was just the cost of my involvement!

I ran a whole team of dedicated service professionals across multiple service areas who all worked extremely hard to eliminate Problems.
So, let’s do one last piece of math here to sum it up:

4. Calculate the cost of 20 people who, say were 50% dedicated to Problem elimination activities over this 5 years period:

\[
\text{20 people} \times \$960,000 \times 50\% \text{ utilized on Problem elimination} \\
= \$9,600,000
\]

So for me and 19 others it cost my previous employer just short of $10M over five years just to tackle and eliminate the Problems we encountered.

Also, this cost does not include lost productivity time for the business, the cost of failing to serve customers or the loss of new potential revenue if we ever lost prospective customers.

So, you could say that it cost a minimum of $9.6M to produce the ‘experiences’ that led to this report.

\[(I \text{ fully realize that this is a very basic model with some big assumptions but it does provide you with an indication of the sums of money involved.)}\]

Obviously there are many other costs and more advanced models on the true impact of problems – but I just wanted to give you an idea of the value behind this eBook.

All of my five years worth of experience with Problem elimination has been crystallized down for fast and easy consumption over the next 60 pages.

You don’t need to hear all the ‘stories’ that accompanied over 1,570 distinct problems over the five years. Neither do you need all the gory details about the human factors that contributed to the problems or assisted their elimination. I have spared you that and just extracted the relevant pieces.

I am assuming that as a key Service Provider yourself, looking to eliminate problems, you’re just too busy fire fighting and getting on with the day to day job of providing service to worry about such things.

No, the goal of this eBook is to tell it like it is, share all the key practical insights and show you a powerful way how to adapt these techniques to work in your own organization. Simple as that.

I imagine some of the ideas and techniques you will read – you may already be doing something about - so you should take some pleasure in seeing them replicated here! It is always gratifying to read elsewhere that others are following a similar approach.
But no-one can ever have a monopoly on good ideas (last I heard) so there should be another tranche of ideas on these pages that you can grab hold of and pilot or try either next time a problem occurs or preferably put them in place BEFORE they occur.

At the very least – my advice is certainly ‘food for thought’ and you can engineer your own ingenious ways about how you can best adapt them to fit your local operating environment.

If you enjoy reading the eBook then I would kindly request two things: -

---

**TWO REQUESTS FROM THE AUTHOR**

1. **You are free to forward a copy of this eBook to anyone who you feel would benefit from it!**

   For example your staff on the Service Desk, the Incident Coordinator, the Problem Manager, the change team, the Availability folks – you know who’s best.

   As the author – you have my express permission to freely share this eBook – so long as it stays intact and is not changed in any way. This means the content, layout and copyright.

2. **If you have any comments, feedback, observations or additional ideas that WORK in real life – please take just a couple of minutes to e-mail us and tell us about them.**

   What worked for you? What did you really relate too? What did you not like? What else is missing? What are your successes?

   Be good to hear from you! Who knows – if ever this eBook gets revised – your success stories could be featured in the next version (obviously I will ask for your permission first).

   **Thank You.**

---

I encourage you to share this eBook with colleagues and friends who will find the contents beneficial and to e-mail me at the address below to let me know what worked for you!
Controversial Statement

Let me begin by controversially stating there are only really 7 sources of problems. Just seven.

Every IT system, service, application or Infrastructure “problem” can be traced back to at least one of these seven specific sources.

This eBook presents you with a series of activities, approaches and techniques that you can practically use within your own organization all based around a model I developed some years ago. The model describes seven sources of Problems and highlights practical actions and steps you can take to eliminate each of the seven sources.

Attacking the root cause of your problems and successfully eliminating them once and for all is obviously the right thing to do.

However, you also need to get really tough on the sources of your problems in the real-world operating environment of your business. But first…

Do you also investigate and eliminate your “near misses”?

Take another look at the picture on the left. It was taken from inside another plane!

In the Airline Industry, where human safety is paramount, any ‘planes found within a certain distance of each other always have to take evasive action. Following this each pilot always have to follow through with a detailed report on how they got to that position in the first place and what they need to do to prevent a recurrence of such a near miss.

Imagine if your Problem Management team did the same? I bet your team would be considerably larger than it is today.

Nevertheless, the seven sources model helps you to consider not only actual “near misses”, but potential “near misses” too. Real proactive Problem elimination.

We will begin by taking a very detailed tour around each of the seven sources and I will present back to you my recommended approaches, techniques and activities that I found to really make a difference and deliver bottom line results. I recommend that you read all the following sections in the order presented in this eBook because they are often inter-related and I believe that each section builds nicely on the previous one.

Let’s start at the beginning of our journey…

© ITIL Training Zone
Introducing the Seven Sources Model

The seven sources model is simply a reference model to help structure your thinking and improvement plans for eliminating Problems within your organization.


This is the basic model at the highest level and shows you only the first set of labels for each of the seven sources.

The arrows do depict whether the source of the problem originates from ‘outside’ your production environment or in the case of Source #6 from ‘inside’. I will cover this off in much more detail as I explain each source in turn.

As a service professional you will be all too aware of the stigma and negativity that unfortunately accompanies most problems.

Whose Fault is it?

Before I cover each of the seven sources in detail there is a key point that I would like to highlight to you first. This is critical in ensuring that all of your efforts to eliminate Problems from your production environment are successful. It relates to the subject of blame.

Unfortunately blame cultures exist in many organizations because as human beings working within a defined corporate hierarchy the people higher up sometimes feel the easiest way to protect themselves from the negative consequences of the impact of Problems is to apportion blame.

If your culture revolves around someone senior always shouting the question, “Whose fault is this?”, then you have my sympathies.
The **S7VEN** Sources of **Problems**

The first thing you should try to instill in your organization is that Problems DO happen and sometimes they could not be foreseen. After all that's why we take out insurance on our cars and homes – to cover unforeseen risk.

Pointing the finger at individuals and blaming others is always counter productive in the long term. If a blame culture is re-enforced for a sustained period then those humans at the end of the pointed finger will usually react by stepping back, hiding behind paperwork and processes and eventually will become less productive than you require.

As one of the many people in your organization who is responsible, in some way, for contributing to the successful elimination of Problems you need to be aware of HOW you can be blamed. Then, you need to formulate your immediate response typically with an action plan to prevent the same thing from ever happening again. You can use the ideas and approaches in this eBook to assist you.

I have provided you with this helpful mapping of each of the seven sources of Problems to the typical 'blame' statements that you will hear from time to time.

<table>
<thead>
<tr>
<th>Problem Source</th>
<th>Why YOU'RE to blame!!</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Acceptance into Production</td>
<td>- It got into your production world, you let it in, you caused the Problem! You’re to blame!</td>
</tr>
<tr>
<td>2. New Changes</td>
<td>- YOU let it in through CAB, they implemented it; YOUR change caused the Problem!</td>
</tr>
<tr>
<td>3. Upgrades / Patches</td>
<td>- You let them upgrade it (with or without change control), YOU caused THE problem!</td>
</tr>
<tr>
<td>4. Vendors / Suppliers</td>
<td>- This entire table applies! But considering Vendors instead of your local environment, they caused YOU the Problem!</td>
</tr>
<tr>
<td>5. User Error</td>
<td>- You let your customer's use a system that they could break (bit unfair - some users will try hard to break their systems – albeit unintentionally). YOU caused the Problem!</td>
</tr>
<tr>
<td>6. How Production is Executed</td>
<td>- The way you run your systems (re-scheduling batch processes, clock changes, deleting a job and it's dependencies fall over. YOU caused the Problem!</td>
</tr>
<tr>
<td>7. &quot;Failures&quot;</td>
<td>- Electricity failure, mechanical breakdowns, elastic bands snapping! YOU did not see this coming? YOU caused the Problem!</td>
</tr>
</tbody>
</table>
I sincerely hope that you do not work within such a culture but unfortunately it is all too common these days.

The good news is you can use the approaches within this eBook to enable you to take immediate positive action to prevent any kind of similar recurrence without ever promising that “it won’t happen again!”

How can you possibly EVER promise this?

There are always a million different reasons why something similar could happen again. As a human being and more to the point, as a service professional, you can only ever take rapid action, execute quickly to close any gaps and then move onto the next priority.

Why am I stressing the whole ‘blame culture thing’ so early on? Because it plays a significant factor in how successful your long term Problem elimination activities will be.

Consider these four real life people implications of people who have been ‘blamed’ for a recent Problem occurrence.

<table>
<thead>
<tr>
<th>Why Blame Prevents/Hinders Problem Elimination?</th>
</tr>
</thead>
<tbody>
<tr>
<td>➣ <strong>Blamed people are ‘scared’ to make a difference next time</strong></td>
</tr>
<tr>
<td>- loss of innovation and new ideas</td>
</tr>
<tr>
<td>➣ <strong>Blamed people will get someone else to do it next time</strong></td>
</tr>
<tr>
<td>- loss of productivity</td>
</tr>
<tr>
<td>- loss of motivation</td>
</tr>
<tr>
<td>➣ <strong>Blamed people feel embarrassed and will re-trench</strong></td>
</tr>
<tr>
<td>- loss of creativity and energy</td>
</tr>
<tr>
<td>- loss of respect for senior management</td>
</tr>
<tr>
<td>➣ <strong>Blamed people will feel less likely to get involved</strong></td>
</tr>
<tr>
<td>- loss of valuable source of talent</td>
</tr>
<tr>
<td>- loss of contribution going forward</td>
</tr>
</tbody>
</table>

The impacts are all very negative and will directly reduce the number of qualified and skilled people that you can engage with, over time, to support your Problem elimination efforts simply because their hearts are no longer in it due to the fear of what happens the next time they are blamed!

**So what can you do to help eliminate any blame culture?**
I led an unadvertised and unannounced programme of cultural change over the first two of the five years (yes – it took that long to work!) to help drive out and eradicate the blame culture.

Here are five approaches that I found successful: -

<table>
<thead>
<tr>
<th>Eradicating A Blame Culture</th>
</tr>
</thead>
<tbody>
<tr>
<td>➜ Re-education</td>
</tr>
<tr>
<td>- People who make mistakes are re-educated</td>
</tr>
<tr>
<td>- This is documented on their personnel file</td>
</tr>
<tr>
<td>- People learn from their mistakes, take things on board, improve and do not repeat them</td>
</tr>
<tr>
<td>- Management who previously pointed the finger and blamed others, started asking for people to be &quot;re-educated&quot; instead. Much more positive!</td>
</tr>
</tbody>
</table>

| ➜ Over communicating the next time |
| - Next time someone who previously got blamed for something did something great I would over communicate this fact to the person who had previously blamed them for that something. This helps to turn-around the perception of that person and builds trust and confidence for the future. |

| ➜ Taking joint ownership |
| - I regularly stood up and simply said, “I will own this and ensure that it is eliminated”, but I also worked with the person who was being blamed to eliminate the problem. This team work helped the person’s confidence and eliminated the Problem at the same time. |

| ➜ Having a ‘quiet’ word |
| - Sometimes, it’s necessary to state clearly and directly what needs to change, one to one, face to face. People really appreciate the honesty. |

| ➜ Firing the unforgivable and negligent to protect everybody else! |
| - Very occasionally due to negligence or even gross carelessness – I had to fire someone. |
| - This protects the work that you are doing and demonstrates that you are serious about your elimination efforts. |

Now we have addressed the key cultural factor associated with Problem elimination – it’s time to take our tour around the seven sources.
**Source #1 – Acceptance into Production**

We will begin to work our way through each of the seven sources starting with source #1 - Acceptance into production.

This is one of the most complex and all encompassing sources of Problems and actually can contain any or all or the other six sources of Problems. That’s why I’ve deliberately used the rainbow color effect for this arrow.

It’s important for you to read all of the other sections to fully appreciate the full extent of this source. The difference is – they all come from upstream, from new projects, from new systems, from major release, from QA environments – this source originates not from production but further up the ‘food chain’.

For now, the important point to understand and appreciate is there is a point in your development into production process that actually allows a multitude of Problems to enter your production environment. That is why I am covering this source first.

If you can begin to tackle this source then the techniques, approach and actions you take can quickly prevent many new and previously unseen Problems entering production.

The diagram below highlights the ‘Acceptance into Production’ arrow.

I have used the multi-colored arrow deliberately to represent the fact that any of the other six sources of problems could actually enter your production environment via this route and cause you a new problem.

Notice that the arrow I have used is ‘broken’ (to the left hand side) to represent the fact that typically most advanced IT Service Organizations have already installed several “Quality Gates” or “Checkpoints” (for example at the end of testing) to help ensure that new Problems DO NOT enter production.

We all know that despite these best endeavors they invariably do!
Such "Quality Gates" are meant to prevent poor quality systems and badly produced support documentation in the first place.

You must put yourself in a position where you know when a new system or release reaches the final gate - prior to entering production – that new system is:

- Known, understood and capable of being supported
- Supportable by people who understand what they need to do
- These people are ready to begin supporting it
- The services to maximize the value of the system for the business are in place and functioning correctly
- There are no nasty surprises!

That’s quite a list. To ensure this is the case and all those things are done – there are lots of activities that must happen.

I fully appreciate that such activities mean expending time, management guidance and effort. Therefore a cost is to be incurred by your business. Many organizations will be unwilling at first to support such an ‘investment’.

This is mainly because the final implementation date or perhaps ‘time to market’ has already been agreed and publicly stated – long before the system was ever built or tested!

Sometimes internal business politics will dictate that this new system or release WILL go live on a certain date and time – regardless of the state it is in.

I’m sure you have experienced this by now. It can be disheartening for all involved to be over ruled by senior executives who are pretty detached from the grass roots - however as guardian of your Production environment it is your job to ensure that this does not happen – or if it HAS to happen – it does so in a pre-agreed fashion, as far as time and resources will allow. Call it a damage limitation exercise if you want.

To help you prevent new Problems entering production from such damage limitation exercises you must have: the right people involved, key activities planned and the correct information captured and documented.

Furthermore this information must be ready for communicating to the right executives at the right time. With damage limitation - timing is everything.
Sometimes **hard facts** about the likely risks to the ongoing success of the business are just about the only thing that executives will listen to.

Remember it is one scenario to stick to published timescales and allow a Problem riddled release to enter production – but quite another to actually impact the thousands (or millions) of customers who already do business with you! Even Senior Executives will take this on board.

Anyway, let me step back from the damage limitation scenario and get back to the things you can do to prevent new Problems from entering production from source #1.

As we know all too well, the types of activities that work best to eliminate potential problems (such as testing and education) are sacrificed if the project’s running late or over budget – OR – such activities are allowed to be performed so late and with little focus that the new system might have well been "thrown over the fence" into production. Sad but true.

This scenario always places massive pressure on Service Organizations and significantly contributes to create (or sustain) the “Reactive” support organization within production.

This, in turn, actually drives up the likely volume and frequency of problems due to inherent instabilities and inherent weaknesses in support capability. Breaches of Service Level Agreements and incurring penalties are also imminent.

**So, what is the bottom line here?**

*The business desire to get a new system in place riddled with potential Problems into a production environment that has not geared up for supporting and managing the new system WAS BIGGER THAN* their desire to not impact current customers and control the cost of managing the system in the future.

Well, more problems occur when the support services for a new system or release are not: in place, rehearsed and the people present are physically trained, experienced and capable of delivering that service.

**So, what can we do about this situation?**

Straight from the 5 year ‘data bank’ of best practices – here are my recommendations on some of the key proactive actions that can be executed during the forthcoming system/service introduction - to prevent the above scenario from taking place.
The \textbf{S7VEN Sources of Problems}

\textit{‘Known’ Errors}

Some time before a new system, project or major release (whatever your local terminology is) actually enters production (and therefore has the potential to negatively impact your business) you need to formally take stock of the volume, status, ownership and ‘risks’ associated with items called ‘known errors’. \textit{(My apologies to ITIL purists at this point because I deliberately use this term to mean something different to the usual definition)}.

These are called defects, challenges, bugs – heck, I have even heard them called “special features” – that will ruin your service levels, raise the volume of incidents and ultimately cause you new Problems.

Here are four essential ways to manage these kind of ‘known errors’:

\begin{itemize}
  \item \textbf{FOUR ESSENTIAL WAYS TO MANAGE KNOWN ERRORS}
  \item Ensure that the Problem Management team has FULL visibility of all known errors \textit{way before} the system enters production.
  \item Allocate resource to \textbf{proactively} work with system and service testers to enable a better picture of the types, volumes and possible impacts that known errors \textit{might} cause.
  \item Ensure people work diligently with the testing teams to drive out as many known errors as possible – \textit{before} they enter production.
  \item Use a risk/impact/exposure matrix to ensure that all known errors are being tackled in strict priority order.
\end{itemize}

I’m realistic enough to know that inevitably there will be \textbf{some} known errors that HAVE to be accepted into production. You should be too. It’s absolutely impossible, in these times of highly complex and massively functional systems, to ever be 100\% confident that ALL known errors have been caught, investigated and eliminated before entering production.

The key message though is that they should be \textbf{KNOWN} errors. Known, managed, tracked and carefully monitored until they have been eliminated.

By \textbf{KNOWN} I mean: documented, communicated to all the right teams and all the appropriate service restoration workarounds already documented, tested and in place – ready to execute.

In the real world this means the support teams are educated to actually execute certain service restoration procedures quickly and effectively.
It also means that you will require capable resources to be actively engaged
during the mid to later stages of new systems developments to prevent Known
Errors getting into production. This is an investment in time and effort. Some
people are also better at doing this than others. It’s not a job for the faint
hearted – that’s for sure.

The good thing is that the more problems you prevent up front – the less you
have to deal with in production. In turn, this means the less impact you have
on your customers, your business and the less time you spent running around
like a headless chicken, stressed out because the current service is not
available.

Finally all known errors should have a follow-up plan stating when their root
cause will be eliminated and what resources/timescales are involved in doing
so. Ownership of this plan should lie with a senior manager who has direct
responsibility or the availability and well being of production – NOT –
development or the release. That person will probably be on vacation by then.

**Single Point Of Failure (SPOF) Analysis**

This is a really powerful and proactive technique that you can use to identify
all the “weaknesses” in a system or service to help you rectifying that
weakness of eliminate it altogether.

What do I mean by the term ‘weakness’? Well, you can read these two ways.
Either the system has already failed and is causing a production problem –
OR – it is highly likely to fail soon – so you had better take action.

**Why do single points of failure exist in the first place?**

There is obviously a cost involved in installing a highly resilient and redundant
Infrastructures. Sometimes the original business case for such high levels of
resilience really does not ‘fly’ with lots of expensive duplication options - so
single units are opted for instead. It keeps the initial budget down and it also
makes the ROI for the project look that much better.

So exciting new systems that promise to “deliver the world” to your business
lines or customers can sometimes be built, or connected, or operated with
SPOF’s already built in and delivered.

Oftentimes single points of failure are very subtle.

SPOF’s can, for example, take the form of just a single file system that drives
a new application, as opposed to a complicated fully redundant device located
at a back-up data center.

SPOF’s also apply to human beings. How many individuals do you know in
your organization who is the ONLY person who can do something?

Do you recognize the person in this scenario?

*When a service is down or running extremely slowly – everyone rushes to this one person who amazingly can restore service and seemingly eliminate the problem 'on the spot'.*

It is critical to proactively identify who these people are and then attach diligent resources to them for a number of hours (or days) to document what they know and how they do it.

I encourage you to seriously think through where you have single points of people failure, knowledge failure or capability failure.

Is there just one person who knows how to restore service for a given system? Find them. Document what they know and how they do it. Educate at least two other people. Get them to rehearse executing this restoration activity. Remove all known SPOF’s.

In a complex environment with multiple actual or potential SPOF’s you would follow these steps: -

**Steps To Eliminate SPOF’s**

- First identify your SPOF’s
- Risk assess and then prioritize your SPOF’s
- Work to eliminate or reduce them to an acceptable level
- Document the execution of rehearsals
- Take action to improve restoration procedures

Whilst SPOF elimination is in motion, you may wish to make sure that any existing and forthcoming SLA’s are reviewed to reflect the level of risk - in particular if it could impact service availability. You should be proactively informing SLA owners and Service Managers of these risks. They will very quickly support you with your SPOF elimination efforts.
Unsupportable Components

It is often the case that many months after a system has gone live there is a component failure that no-one can fix anymore.

So, for example, a certain brand of hard disk fails and impacts service availability. The Vendor has ceased to exist, was bought out by a larger vendor or that specific product is not within their portfolio anymore.

I particularly found this to be relevant in the Telecoms and Data Communications Industry.

Another example of ‘unsupportedability’ can be because the support resource has left the company or moved on. Maybe no-one even knew that this person was the only resource in the entire company that knew how to support this piece of infrastructure? (See SPOF elimination).

Even more subtle than this example is how Versions and Releases of components “fall off” a Vendor’s support portfolio.

Key actions to be taken in this area include:

<table>
<thead>
<tr>
<th>Supportability – Key Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>➔ Identify and list all components</td>
</tr>
<tr>
<td>➔ Verify each component’s ongoing supportability</td>
</tr>
<tr>
<td>➔ Keep a special calendar of upgrade paths</td>
</tr>
<tr>
<td>➔ Ensure ALL Vendors support portfolios are linked to Underpinning Contracts</td>
</tr>
</tbody>
</table>

This may seem like a lot of work however any overhead is made much easier when this approach is integrated into your Configuration Management Database structure. Why not check this out with your Configuration Manager?

So each infrastructure and service component or configuration item should have their supportability status recorded and tracked. Validating unsupportable components then becomes as easy as producing a report.

Where unsupported components ARE found - you need a supportability plan
to execute against it.

Remember the golden rule of supportability: -

**Supportability**

*If it's in production it must be capable of being supported.*

---

**Testing Evidence**

Another potential major source of Problems to explore is during the “testing” phase of any new system or major release.

Why not ensure that your Service teams get a good chance to review the results of any tests before testing cycles complete?

For example why not free up some quality time for your Service desk, Incident Management and Problem Management folks to go and visit the test area where user acceptance testing is taking place? Get them to talk through the types of defects raised and what the current focus areas are. At the very least they will be more aware of the forthcoming new system and therefore be more geared up and knowledgeable when implementation time comes.

I found this approach to be really useful and effective. Sometimes some of the team came back with some great ideas for how to best support the new system. Ideas that would not have surfaced until AFTER an incident had been raised or there were Problems to eliminate.

For my money, there really is no better approach than to allow Service teams to get a good feel for the types of failures found in testing and the remedial work necessary to overcome these bugs. In most cases, this can provide a general indication about what types of problems can be expected later on.

Keep in mind though - there will always be surprises!
**User Acceptance**

Did you know that spending quality time ahead of implementation directly with your user community can also pay dividends?

One positive thing that I used to ensure happened was that the future Service Managers of a particular new system (plus the Service Desk Team Leader and the Incident and Problem Managers) spent quality time with their future customers, in this case the internal business lines.

These people got an earlier view, a better handle and also first hand experience about what kinds of bugs and issues the users experienced and what there attitude was to how these were being fixed.

A pleasant side effect of this ‘caring’ approach is that the business lines will appreciate your time and consideration and you have already begun to start building up a positive relationship even before the new system enters production and the monthly service reviews begin.

**“Does not meet (business) requirements”**

It is an unfortunate fact of real life that sometimes end users receive systems that they ‘deserve’ rather than the one’s they asked for! This may be the fault of the new systems Project Manager – who has failed to fully engage with the appropriate business people – or maybe the fault of the business for not making the time to fully engage at the appropriate stage. Either way if the end system does not do what it’s supposed to do then this will lead to new Problems later on.

Taking a really detailed example here – consider the scenario where a certain menu sub-option is missing and this ‘defect’ has crept its way through testing and UAT. It is going to hit the Service folks some stage soon after go-live. This is a Problem waiting to happen! It could have been ‘caught’ earlier on and something constructive done about it.

Overall you should ensure, via testing and user acceptance, that the forthcoming system has been signed off and approved in as many aspects as physically possible by the business. This will prevent such Problems from occurring later in production.

I would also recommend here that the Service Level Manager, the Service Desk and the incident and problem teams hold a copy of this sign-off for at least the first few weeks after a new system enter production. This makes it easier and faster for them to refer back too. Also bear in mind that where a new system does ‘not meet requirements’ this will lead to a higher level of early requested changes for the first few weeks of the systems life in production. Higher volumes of changes (emergency or otherwise) will feed the next source of Problems - ‘Change’ - which I will cover for you now.
The **S7VEN** Sources of **Problems**

**Education / User Knowledge**

Any ‘gaps’ in end user education or knowledge often means that end users will quickly find local ways around overcoming the shortfalls in their new systems capabilities. This can also lead to new Problems further down the line.

The classic example I found over the five years was what I always referred to as the ‘local breeding colony of Excel spreadsheets’.

To overcome deficiencies in new systems, end-users would regularly create multitudes of spreadsheets designed to extract live data, manipulate it and produce desirable output. In many cases there is nothing wrong with this.

Why not? It is cheap to produce, very relevant and often well executed. But most times such Spreadsheets lie within the depths of end-user’s local hard drives. Sometimes this data became lost or corrupted and an important part of the overall ‘system’ became inoperable. The Service Desk would regularly receive such calls and these would always be requested by the end-user to be logged as ‘critical’. This immediately creates pressure for the support teams to resolve the problem as quickly as possible, for something that they never even knew existed.

Such ‘gaps’ can be overcome:

---

**Overcoming Spreadsheet Challenges**

- Register such Spreadsheets as ‘critical’ assets in the CMDB
- Introduce a specialist support team to tackle the ‘top 10’ most frequently occurring types of Spreadsheet challenge
- Introduce local end-user working practices that influence end-user behavior around the creation, use and management of all such ‘critical’ Spreadsheets
- Encourage end-users to request system extensions for any large/complex Spreadsheets to transform them into an integral part of the system

---

© ITIL Training Zone
Such gaps are important to investigate and they should be closed as far as possible in advance of any system acceptance into production.

Before I wrap up this section there is one more key process that I need to highlight to you. Traditionally many organizations have already recognized the types of ‘incoming’ challenges that new systems or major release represent and have tackled this by creating a Service Introduction team.

In early 2007, the next ITIL refresh will be published, which includes a brand new process called Service Transition. From the materials and information available at the time this eBook was written this new process looks to be synonymous with what I describe as ‘Service Introduction’.

The aim of Service Transition is for a team of people to work proactively with the project and release teams to ensure that all of the above challenges are met head on and tackled before there is any negative impact on production.

The introduction of such a Transition process will involve an investment in people’s time and energy to: design, create, educate, implement and run on a day to day basis.

Service Transition’s goal is to ensure the smooth and impact free transition of a new release, system, project or service into production without impacting service levels or increasing the support costs of the organization.

For now, I will leave this section by highlighting two guiding principles: -

**TWO GUIDING PRINCIPLES**

1. Implement a strong and well adhered to "Service Transition" Process - to prevent poor quality and unsupportable systems entering production.

2. Ensure Leaders have the strength of character to be able to say, "No", to new systems implementation - when there is enough evidence that an unacceptable level of risk to the current operation will be introduced IF the new system is implemented.
Source #2 – New Changes

We continue our tour around the Seven Sources model with a closer look at the next source of potential new Problems and that is "New Changes".

For this source we have moved away from looking at the challenges of the development or project environment – and are firmly focused on the actual live, day to day, production environment. Specifically, I am going to share with you my insights on how new Changes that are entering production can cause new Problems and what you can do to help eliminate this source.

I need to point out that I have deliberately separated "Upgrades and Patches" into the next source of problems which we will cover in section #3. Strictly speaking “Upgrades and Patches” can be classed as a sub-set of "New Changes" but for reasons we’ll go into in section three - I have created a separate source just for them.

For now though, please bear in mind that these criteria for "New Changes" also apply to source #3 - "Upgrades and Patches".

So how do Changes cause Problems?

Well, strictly speaking they shouldn’t!

But we are dealing with real life here and in the real world things do not always go according to plan.

Using pure ITIL terminology we can link together the relevant stages of a failed change and create what I call the “Failed Change” chain. All the individual pieces must work effectively as well as the linkages in between. To ensure linkages between processes work, human beings must communicate effectively and know what they have to do at each and every stage.
The “Failed Change” chain works like this:

**The Failed Change Chain**

- Failed changes cause incidents
- Incidents are logged and progressed
- Incidents are resolved and service is restored
- This becomes a **Problem**
- Problem Management will investigate the underlying root cause. Once it is determined it becomes a **Known Error**.
- A new **Change** is raised to eliminate the Known Error once the Change is successfully implemented.
- If the Change **fails** it begins the chain again…

For completeness here, we already know that **Successful** Changes don't cause problems. They *enable* improvement and the realization of opportunities often linked to new business value or the protection of service.

Now, we won’t get too bogged down with all the details of the ITIL workflow here but we are going to focus on what you can do to prevent (or minimize) the ultimate source of problems for “New Changes”.

There are several key reasons why changes fail. Let me highlight the key one’s that I identified as the most likely to cause new Problems in production.
Lack of Appropriate Approval

The change did not receive the appropriate level, duration, depth or quality of review prior to its approval.

So, the end result is that the change is (often unwittingly) misapproved. You can now think of it as a ‘time-bomb’. It’s been approved (armed) and it’s heading for your production environment (to explode).

This change should have had some important or additional aspect of it validated or raised as an objection during a Change Advisory Board (CAB) meeting or other senior level approval meeting - but this didn't happen.

So, some impact is going to happen as a result, at some point in the future. It may not be straight away. (The bomb is armed and ticking… tick… tick… )

Tackling this challenge is best done through your CAB meetings directly, but you can also create special "advisory" groups who pre-approve forthcoming change packages prior to them going to the full CAB for final approval. Think of this as added protection from those people who are suitable knowledgeable about the finer details of the change.

Such advisory groups will generally focus on specific elements of change. I created two such advisory groups: the "Technical Advisory Group" and the "High Availability Advisory Group". You should aim to create several of these groups making sure that you tag the responsibility onto a group that already exists - to avoid any unnecessary internal politics about the (alleged) resource utilization of your Change Process.

Remember - all Changes involve some degree of risk.

You will NEVER be able to consider all possibilities, permutations or eventualities within the timescales you have available.

You just need to ensure that the organizations Change process and the people who facilitate and approve changes are provided with the best information and the best opportunity to mitigate any risks prior to implementation.

This all sounds rather obvious though – doesn’t it?

I mean – OF COURSE – changes must be properly approved or else they should not be implemented into production – this is the primary role of the Change Manager.

The trouble is – in real life – many factors can hinder the operation of your Change CAB meeting and lead to its downfall.
Factors that hinder effective CAB’s include:

- Not having the **correct people** present to approve
- Not having all the **relevant facts**, information and details available in time to enable a timely approval
- Senior Executives ‘**railroading**’ a change through by applying pressure for approval
- A **weak or inept Change Manager** or Change team who allow some requestors more ‘space’ to drive mis-approved changes through
- An organization that does not really support what the goals of the Change Management team really are. They pay **lip service**.

You really need to assess the effectiveness of your CAB members and the CAB itself. You can use the above factors as a guide to help you do this. Where you find gaps or opportunities for improvement you should create an action plan and personally take ownership of ensuring the successful completion of all the tasks on it.

**Poor Testing**

It never ever ceases to amaze me how frequently un-tested changes are approved to enter production. I found this to be the case early on in my five years.

One classic reason for this is because the CEO shouts loudly and wants some new functionality in place ASAP!

Everyone beneath the CEO suddenly cuts corners, ‘moves mountains’ and drives the change through in double quick time - missing all the appropriate check-points. The CEO gets what they want – but at what price?

There’s not much science here: **poor testing = higher risk change**.

It’s that simple.
Any CEO who is shouting loudly will not alter the inherent risk factor. He or she merely changes the downstream human desire to please and implement the change as quickly as possible.

Ask yourself the question - How much LOUDER will the CEO shout if all your systems fail and your organization takes a massive 'hit'?

You need to be a very strong IT leader to stand up, be counted and present the risks to the CEO in the correct way.

I found that open, constructive and reasoned discussions led to reasoned decisions. You will need to be armed with all the right facts before you have these discussions. So long as the discussion takes into account the facts, then the correct decision making process has occurred.

Remember, it’s always far easier to return to the CEO a little while later and explain how that risk you told him about yesterday actually did occur – despite the agreed plan to mitigate it – than it is to advise him ‘cold’ of a failed change that cost you lost dollars.

You are not being selfish or self-congratulatory here. No, you are protecting the continuous operation of your business. This not only means your customers ability to do business with you but your companies’ ability to carry on generating profit.

Whilst we are on the point about ‘shouting’ CEO’s. This reminds me of a story I heard from a very straight talking and committed Change Manager who once worked within my Service group.

He was in CAB one Monday morning and there was this one particular Senior Executive who was complaining bitterly to everyone about how it was unreasonable to 'slow down’ his change request (I think the Change Manager had asked for some further information to support his requested implementation time).

Anyway, this senior manager suddenly snapped, “This damn Change Process is just red tape bureaucracy gone mad – we need to be fast and change quickly – my approval alone should be enough – Change Management is just slamming the brakes on everything!”. The room fell silent and all eyes turned to the Change Manager.

He calmly replied, “You’re absolutely right. Change Management are the brakes. You know, we slow things down just long enough to enable us to accelerate at top speed around the next bend in the road. We also protect our business from ‘crashing’ by verifying risks and impacts on live service before we do changes, and very occasionally we slam the brakes on to prevent ongoing failure by enabling emergency changes to enter production in record speed. So, you are the accelerator and I am the brakes. Let’s work in tandem to drive our business forward…”

The room fell silent. Ah, the power of words.
Anyway, back to the main thrust of this section – this is the lack of testing before a change is approved...

Another reason why this occurs actually lies within the “testing methodology” itself. Perform an audit for yourself and ask these questions: -

➤ Are the right skilled resources carrying out the testing?
➤ Are the testers allowed enough quality time to implement a fix to any given bug?
➤ Are testers influenced by the business too much?

The list goes on... but the consideration here is simple: -

The overall quality of your testing function is directly proportionate to the quality of the live system or service that you are expected to support after implementation.

**Poor Implementation Planning**

Do your changes come attached with a well thought out and properly walked-through quality implementation plan?

If the change is of any reasonable size then have the implementation team walk through it with the Change team to review the key stages and ensure that all appropriate support activities are in place.

There is no substitute for great planning. It prevents the pain of production Problems later on.

In my experience the very best Change teams ask the most challenging but obvious questions, especially the “What if this happens?” kind of questions. So if the change was badly planned, poorly thought through or the right answers to those challenging questions were not forthcoming then the Change did not get approved to move into production.

A suitable and well structured implementation plan will actually guide all the relevant change designers, builders, testers and implementers to do the right tasks, at the right time – with in built validation.

You should take an action to review the people, capabilities and experience levels of the people who make up your Change team. Don’t go off firing anybody or transferring them to some remote outpost of Service – instead put a temporary ‘coach’ in into the team to support them and provide them with ongoing feedback to improve their performance. Remember: They must be asking challenging questions to be effective.
Poor Execution of the Implementation Plan

A great plan badly executed will still create new Problems. So there is much value in helping your teams to develop the ‘art of execution’.

*I do not mean chopping anyone’s head off, although I’m sure some people that feel this radical measure would actually prevent a stack of new problems entering production!*

The ironic thing is – even the best laid plans if poorly executed, are a waste of time. The best laid plans also need to be flawlessly executed. If things go wrong, as they often do, small workarounds and fall back positions should have already been identified, documented and be ready to execute - in advance.

---

**How To Create a Great Implementation Plan**

*I pulled together a cross-functional group of operations, service and support people to create a generic change implementation plan that would be fit for purpose for ALL and ANY changes that we could possibly install into production.*

*All the key headings were at such a level as to capture what needed to be recorded and used to help the change implementation run smoothly.*

*This whole exercise took about three hours from 8 people. That’s a 24 man-hour investment that directly led to a fall of 12% of failed changes – the very next week!*  

*I estimated that this result represented an ROI of over 500%.*

---

Finally, the best advice I can give you about execution is this - have the best person for the job in charge of ensuring timely and accurate progress against the plan during the execution of it.

So, if things start to drag on or fall away, this person knows when to effectively 'pull the plug' on the change and fall-back to (or restore) normal service. Such a person will know not to carry on regardless and perhaps just ‘hope for the best’. Similarly, this person will also have the strength of character to say ‘no’ to other more senior people involved in the change implementation.

*If you want to learn more about the ‘art of execution’ then I would strongly recommend you pick up a copy of a book by Bossidy, Charan and Burck entitled: “Execution: The Discipline Of Getting Things Done”. It’s 278 pages of sheer excellence and was voted Tom Peters’ best business book of 2006.*
The **S7VEN Sources of Problems**

**The Impact of another (concurrent) Change**

Ask yourself how well do you really know what's going on?

Do you (or your team) spend quality time examining the Forward Schedule of Change (FSC)? Do you even have a Forward Schedule of Change?

Is examining the FSC an everyday activity for your team?

Do you have it written into your procedures that when a new problem is opened someone checks the FSC to cross check for a potential match?

Let me give you an example here. If the Facilities folks are testing the UPS power supply systems over the weekend – is this a good time to also be implementing a major new Release into production?

How can your Release, Change and support teams possibly back-out such a large change, over a period of time, without any power? (assuming the facilities Change goes wrong of course!)

For Change Management to be effective - you need to have: -

"**One complete version of the truth**"

I cannot stress this enough to you. How can you ever really know the full extent, scope and degree about what is happening in your production environment without a full and complete list of ALL changes that are approved and scheduled.

All change programmes should ultimately 'come together' into this one version - preferably onto one high level change plan. This activity can actually drive up the quality of Changes too - since it means IT, Technology and key stakeholders have to talk regularly with their change business counterparts - about change!

I've lost count of the number of times Change teams have identified 'risky' change going on elsewhere when the IT folks were also planning a change.

Having an up to date and complete “one version of the truth” also ensures that you can successfully manage your overall risk (by re-scheduling some changes), avoid conflicts and just as importantly you can safely request ALL relevant Changes are ‘frozen’ and not implemented should a major Incident ever occur in production.

What else can we do to prevent "New Change" being the sources of Problems?

Well, there are two final prevention techniques that can help.
Your Acceptance into Production Criteria

Why not create a list of criteria that must be satisfied before any new system enters production?

Such a list will contain items like:

<table>
<thead>
<tr>
<th>Acceptance into production criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>➔ Support Documents</td>
</tr>
<tr>
<td>➔ Infrastructure Diagrams</td>
</tr>
<tr>
<td>➔ List of new CI’s</td>
</tr>
<tr>
<td>➔ Education Plans and Materials</td>
</tr>
<tr>
<td>➔ New Work Instructions / Procedures</td>
</tr>
<tr>
<td>➔ New Support Model</td>
</tr>
<tr>
<td>➔ Early Support / Warrant Support</td>
</tr>
<tr>
<td>➔ User/Business usage information</td>
</tr>
<tr>
<td>➔ List of Known Errors</td>
</tr>
<tr>
<td>➔ List of defect/bugs/issues encountered</td>
</tr>
<tr>
<td>➔ Operating Schedules (Jobs/Back-ups)</td>
</tr>
</tbody>
</table>

You should ensure that each of the items on the list is pre-defined, known and an established part of the way that new Changes are delivered into production.

Over time, all of the teams concerned with creating and delivering these materials and documents will produce them without being asked too! In the early days, it is very hard work because you are seemingly asking them to slow everything down and you can be labeled as someone who is slowing the delivery of the new system or Release down.

No, you are helping to assure its final acceptance and ultimately its smooth running and management in production.
Playing it ‘Tough’

To protect production from Changes that are going to cause new Problems means that sometimes you have to play tough.

Call it ‘tough love’ if you like but you cannot say yes to everyone all of the time. With conflicting schedules, different priorities and varying levels of quality across multiple teams in both development and production – it’s impossible.

You and your Change manager need to develop a solid and consistent reputation for being ‘hard but fair’.

That’s my best advice in this section.

Sometimes I would overhear one of ‘those’ conversations between support folks leaving CAB. I would always smile to myself if I heard one of them say something like, “You know what? That Change Manager is a tough cookie. He knows the right questions to ask and does not let poor quality change get into production. I admire him for that.”

Anyway, one great communication technique that I always encouraged strongly was this: NEVER simply give a straight “YES” or “NO” to a request.

A simple YES is not definitive enough and is open to abuse. By abuse I mean the Change requestor sometimes feels like he is empowered now to do pretty much anything he likes with his Change!

A simple NO is viewed as obstructive and creates a negative impression for the whole Change team and the Change process. This gets people frustrated – even before they go to CAB to present a Change for approval.

The “Yes, however…” Technique

Never give a straight YES or NO. Instead say...

⇒ “Yes, however…
⇒ and then **state the required condition**
⇒ and then **state the negative impact if this condition is not met**.

You can always answer any question with, “Yes, however…” , then state the condition, and then state the negative impact.

© ITIL Training Zone 34
I will give you an example to illustrate the power of this technique.

**Example #1**

**Change Requestor:** I must have this Change approved by 0945 today otherwise the CEO will blow a fuse about this new functionality not going live!

**Change Manager:** I understand. My response is Yes, however… you must have all the appropriate signatures on the approval document otherwise the CEO will be answering calls from stockholders wondering why the new system is preventing customer deliveries!

Doesn’t that sound better than this response?

**Change Manager:** No way - you have not got all the approvals I need. Get them first and we’ll think about it!

**Summing Up This Section**

I have deliberately focused on Change as a large source of Problems because they certainly were for me. It’s an area that seldom little is written about but I believe warrants an entire eBook in its own right!

Hopefully, I have got your brain thinking now about the types of things that you can do to prevent poor quality Changes entering your production environment.

Remember the bottom line on change is this:

**If the change is in bad shape - then it enters production at your peril!**
Source #3 – Upgrades and Patches

Let me move on now to take a closer look at a specific category of Change namely, “Upgrades and Patches”.

The first question that needs answering is: Why have I deliberately separated this out from the previous source, “New Changes”?

The answer lies in the fact that, in my experience, it deserves a special category all of it’s own because “Upgrades and Patches” are:-

- An essential and key component necessary to maintain the health and supportability of all service Infrastructure
- Frequent, very technical and often complex and challenging to apply
- Driven by Vendor Technology roadmaps – change is a constant
- Requiring special consideration - when to apply? How to test? How to back-out? Install these pre-requisites first... the list goes on
- Are often mandated – you have no choice but to install them ASAP!

When you think about your entire Infrastructure that actually enables your IT Service – quickly estimate the number of individual components involved.

Have you got a number in your head yet?

Then, multiply this number by the estimated number of bug fixes, OS patches, PTF’s, Hot Fixes and any other essential upgrade or patch that they need.

*I'll wager you have now got a four or maybe five figure number in your mind.*

*Right?*

The ongoing and frequent updating of all your Infrastructure components is necessary to maintain your organizations Infrastructure maintainability and is something that is usually within your support contract with each Vendor.

Falling behind with updates and patches can often lead to you being ‘forced’ to perform an upgrade - just to remain within the terms of your support contract. On the flipside moving too close to the latest release can lead to you suffering from “first fault found” syndrome.

*With upgrades and patches there is always a very delicate balance between “leading edge” and what I call “bleeding edge”.*
The S7VEN Sources of Problems

For each IT Service you should use your configuration maps to determine what strategy to apply to upgrades and patches for your underlying Infrastructure.

Do you have such a process in place today?

If you do – great! If not, how can you really assure the ongoing availability of your Service, given that it may or may not be error prone or out of contract?

Here is the diagram for Problem source #3 – Upgrades and Patches.

Let’s consider some of the lower level details with this potential source of Problems.

There are several factors to consider: -

1. Being “aware” of a patch or upgrades actual existence in the first place

This sounds a bit simplistic but part of the challenge in maintaining the correct balance and level of upgrades/patches is actually knowing what’s out there and available to install in the first place. Different vendors provide different levels of service and your contract will also differ depending on who signed the maintenance agreement, when the system was installed and what tools you have in place to search/validate/upload/test/install your upgrade/patches.

Most Technology Vendors use the “post it (on the Vendor’s Intranet) and they will come if they need it” approach.

This does three things. It enables the Vendor to say, “Go and download this patch and try it in your test environment” (so the bucks with you!).

Secondly, it means that you have to keep up with what’s out there and perform the validation and risk assessment yourself.

Thirdly, it provides the Vendor with another revenue opportunity, in that they...
can offer to do the ongoing risk assessment and validation for you – for a fee – on top of the maintenance contract that you already have with them. (nice money if you can get it!)

You need to develop an excellent “radar” on a per component, per Vendor basis.

For each of your vendors and the components they provide map out exactly how and how often new patches are released. Ensure that you have an automated alerting system to advise you of any new critical patches.

This function will occur across all of the applicable support teams and it will do you no harm to delegate someone to keep an eye on how well all of the support teams are doing at this.

I requested my Availability Manager carry out monthly audits of all of the support teams upgrade and patch schedules, as well as how they checked daily for new patches. Knowing that someone is responsible for this activity means it is far more likely to get done!

The additional bonus I got from placing someone to oversee this activity was that it broke down the silo’ed mentality of the support teams because there was always one person who has the complete, overall picture of what was going on. One month, we discovered that a Vendor released what they deemed a ‘critical’ patch for one type of servers without issuing anything for another type with similar operating characteristics. We discovered that the Vendors own internal processes were not up to scratch and quickly insisted that the Vendor fixed their challenge!

2. Finding the right balance between ‘leading edge’ and ‘bleeding edge’

Your Service Level Agreements are a good starting place to help you understand how far to lean towards either ‘leading edge’ or ‘bleeding edge’.

For those higher rated and more critical components you will need to be more conservative and tilt towards the ‘leading edge’ avoiding ever being a ‘test’ site for a “bleeding edge” patch.

Where installing a patch represents an operational risk (above what you would normally expect, say for example, you are the first company to use this patch) then this should be performed with everybody understanding and agreeing to these risks.

Another good approach is to try to determine the inherent “stability” level of the component(s) in question. If the component is complex, new or highly dynamic then more frequent and closer inspection of its patch status should be undertaken to protect your environment.
3. Maintaining your organizations overall adherence to recommended upgrade/patch schedules

Often, Vendors will dictate to you what and how often you should update operating systems, middleware and database components. To name but a few. Vendors will also very helpfully post an alert on their company website for you. The trouble is there are often hundreds, if not thousands, of similar postings. Talk about trying to find a needle in a haystack!

But any failure to keep up with the correct level of patching can lead to difficulties in trying to obtain real-time support.

I experienced this scenario first hand and at the very worst possible time you could wish to experience it – when we were trying to log a priority one Incident for emergency assistance!

To prevent a recurrence of such a situation (we had to apply some emergency patches before the Vendor would open a new Incident ticket and investigate our dilemma!) I recommend the following type of action: -

- Create a chart (from your CMDB and Vendor Schedules) that drives upgrade and patch validation activity. Ensure a checklist is created that schedules the review – upload – test – deployment – validation of all upgrades and patches.

- Carry out spot checks from time to time. This is good from an audit perspective and often reveals interesting nuances in the Vendors release policies.

- Ensure that the Support Team Leaders work closely with the Change team to pre-define and schedule on the Change calendar all the forthcoming upgrades and releases in advance. Essentially you are reserving those slots for this essential activity.

- Build relationships with other IT organizations who have similar Infrastructure components like you. Ensure that the Support Team Leaders talk at least monthly to these organizations about their concerns, plans and thoughts behind their upgrade paths. Sometimes another pair of eyes reveals something that you did not spot.

Following these actions will help you avoid being in an ‘unsupportable’ position.
4. Having the physical capability to actually test the next upgrade/patch (install and back-out!)

As time moves on and systems are upgraded and changed you need to ensure that you maintain suitable test environments where patches and upgrades can be tested in isolation of the production environment.

The further you go to replicate production the better however this entails more money and resource. Every organization has its own upper limits.

Its one thing to be able to test one stand alone server with a new patch in a test environment – but quite another to have four fully resilient servers capable of being used for testing that match production.

Some key questions to ask yourself and the Support teams are: -

- Have you got the appropriate logical and physical test environments where you can realistically (or as far as you can) test the new patch without first impacting live production?

- Does the test environment enable you to ‘load’ the system, like production, to test for how the upgraded component works in reality?

- How easy is it to back-out the patch, once installed? “Bleeding Edge” patches are sometimes recalled or superseded quickly.

- Have you a formal interface in place between “Patch Management”, Incident Management, Problem Management and Change Management?

5. Managing Change through your Change Management Process

It goes without saying that installing new patches still falls within the control of Change Management, but time and time again, due to the complexities of patches, technical support folks often seem to have created their own ‘fast path’ process for these types of changes where certain steps in the procedure are ignored or not subject to controls.

This is dangerous and should be reviewed immediately.

6. What to do if you are ‘forced’ to accept a new upgrade/release in a critical (service down) situation

This does happen in the real world. With the massive importance of preventing security breaches and virus alerts, there are many times when your Technical staff will demand that a Vendor patch is implemented without delay. This should obviously be channeled via the emergency Change route.
Another interesting flavor of this is where you have no other choice than to accept a Vendor recommendation to receive a patch. For example, you have recently suffered from a service impacting incident and your Problem Management team is investigating further. They discover that in order to eliminate the root cause the Vendor recommends that you install their latest patch.

**The trouble is – you are the first in the world to receive it!**

Only the inner sanctum Vendor “labs” have ever tested it before, presumably on Infrastructure that does not even remotely resemble yours. This is not encouraging is it?

Worse still the Vendor is absolutely unable to guarantee (because everyone’s environment is different) that this will eliminate the problem – but it’s the best you are going to get for now.

You have just been issued with a fait à compli (sometimes referred to in the Service Industry as a ‘hospital pass’!)

In situations like this – you need urgent buy-in from your own Senior Management and you need to form an overall opinion based on the current ‘pain’ experienced in production versus the risk of something else going wrong. Sometimes it’s tough in Service.

**7. The resources and skills you absolutely require to pro-actively protect your environment**

One recommendation that I like to make – is to ensure that you have the right skills, capability and attitude actually leading the technical upgrade/patch function. It is often a monotonous chore validating the availability of new patches against the Infrastructure estate, but one that can significantly protect your environment from new problems. It needs to be correctly resourced.

This means the right number of competent people, in the right place, doing the right things at the right time.

To wrap up “Upgrades and Patches” let me just finish off by stating this: as with most things in life… this is a bit of an ‘art’ rather than a science. No two environments are ever the same – so Vendors are incapable of testing all eventualities on their test systems.

So you should plan to prioritize, automate alerts, resource up correctly, interface to other key areas, perform testing and spot checks - and above all else maintain an “expect the unexpected” philosophy!
The **S7VEN** Sources of **Problems**

**Source #4 – Vendors**

Now let me shift the focus onto one of the largest sources of Problems – our vendors. Interestingly, all seven sources of Problems can apply to your Vendors, since nowadays they are pretty much integrated into all aspects of our service delivery operations.

**For this reason, I recommend you come back and re-read this section a second time after reading all the remaining sections.**

Firstly, let me recap where we are right now on the 7 Sources Model.

```
4. Vendor (source)

- All the other six points apply here
- The quality and relevancy of contracts, SLA’s and relationships are the major differentiating factor
  - Review all contracts and SLA’s for relevancy - go out to tender to secure a better deal
  - Implement a Vendor Scorecard Programme - measure Vendor performance
  - Integrate with Vendor’s processes
  - Involve Vendors

Production Environment
```

As we already know the services your Vendors provide, combined with those underpinning contracts that your legal people signed up to, directly influence the possible capability offered by your Vendors to your operation.

**Underpinning contracts are therefore CRITICAL to the overall Service you provide to your business.**

In typical Service Delivery Operations, the texture of the support landscape is an interesting blend of internal support, outsourcing, traditional support contracts and a shared services methodology. For some organizations you can add into this mix the fact that each Service you provide is supported in a different way, by different folks and with differing levels of Vendor involvement. All these ‘flavors’ represent a lot of complexity in your Service Operation. You can start to appreciate just how complex an area this is!

The real-world fact is – Vendors are human too. Vendors are just as prone to the exact same Seven Sources of Problems as any other IT Service Provider.

We should therefore consider “Vendors” as a direct extension of our own organizations and so we should re-orientate ourselves and look at Problems from their ‘internal’ perspective.
The **S7VEN** Sources of **Problems**

This should include the challenges they face, how they are structured, how they work internally to provide their outside service to you and how well they recognize and close down each of the Seven Sources.

In my direct experience I recommend that YOU approach this Source of Problem from the perspective that YOU are “leading the way” by evangelizing and instilling the “Seven Sources” model into your Vendors psyche, organizational structure and ultimately in the way they deliver their service into your organization.

So, the focus of this section is about how YOU can work effectively with your Vendors to ensure that they are taking the appropriate internal actions to prevent the seven sources of problems from occurring or recurring.

**How can YOU do this quickly and effectively?**

Here are some recommendations for action and improvement that I have had direct experience with. These approaches and techniques work in real-life.

**Know Your “Vendorscape” – Who does what, when, how and for who?**

Often there is a prevailing attitude that will immediately challenge the premise of working more closely with your Vendors.

It usually comes from the CIO and it goes something like this, “I pay for a Service and I expect it to be delivered – therefore I do not care how they deliver it – just so long as they do. After all, that’s what I pay for”.

Now, this is a perfectly acceptable view, especially in a heavily outsourced Service Operation where the vendor has the lion’s share of the delivery responsibility and the internal IT / Service provision capability has already been considerably diminished.

**However, in my experience, there is always a lead influencing role that can be played by the internal Service Provider, Vendor Manager or YOU.**

This person in your organization, after all, is responsible for the quality of Service provided to the internal business units, and therefore has a responsibility to take positive action on every situation, issue and problem that occurs.

So I guess I am recommending a partnership ‘mindset’ right from the outset where the Vendor is asked, guided and led to take positive preventative action against the seven sources of problems.
Vendors are human too.

I recall once asking a very wise and experienced colleague, who had personally managed over 450 Vendor relationships over twenty years, the following question: -

"What works best: hitting your vendors hard and getting them to deliver against the contracts they are supposed to deliver – OR – working in partnership on a structured programme of change to deliver the desired result?"

His answer...

"You can hit Vendors hard once or twice and you will get results you need. But you need to realize that these are only ever short lived results. People don’t change just because you hit them hard. Sure, they react and will move and will deliver what you are looking for but if they feel that you will just go on hitting them, in the longer term, you will see things change for the worst.

You will get a bad reputation.

The Vendor will change personnel to put someone tougher and smarter in place.

You will see your underpinning contracts get reviewed and an excruciatingly painful level of detail added to counteract unreasonable demands.

The very reasons why you selected that Vendor in the first place will begin to erode before your very eyes.

It is far better to have open, honest and meaningful conversations that lead to properly resourced action plans, with ownership and smart execution. That way you get what you want.

The Vendor continues to enhance the relationship and add value. It’s a win-win."

Wise words indeed. I decided to leverage the benefit of this man’s 20+ years experience and adopt this partnership approach.

This did not mean I became ‘soft’ and relaxed my position on any challenges – it did mean that I became much more open, direct and positive about how Vendor partnerships can be successful.

The humanistic approach really worked – but it involved broadening the mindsets of both my colleagues and the Vendors staff over a period of time.
I have to be honest with you, my human tendency to use the Vendor ‘hammer’ did take some weeks to totally disappear but the results I achieved were nothing short of amazing.

Failed and emergency changes fell dramatically, new incidents nose dived to record low levels, new changes were incredibly well presented and availability sky rocketed.

This mindset shift and the accompanying results started to come through after two months and took around a whole year to embed as a ‘new behavior’ across the Service teams.

A pleasing side effect was the reduction in the cost to deliver key services brought around by the large reduction in reactive work. With the falls in emergency and failed changes as well as service impacting incidents and problems – we could re-deploy many resources into the Service Transition team.

This has the knock-on effect of further preventing new Problems entering production. A virtuous circle!

After about 18 months effort, we could start to talk about reducing the volume of resources within the Transition team and lowering the annual price of the contract.

Interestingly, it was the Vendor’s account manager that raised this opportunity with me! She was keen to demonstrate ongoing added value to our organization by lowering the cost of the contract, enabled by the Vendor reducing their internal costs to deliver.

Before I get too carried away, let me get back to some of the key activities that we focused on in that first, often difficult, twelve months.

The Quality and Relevancy of Contracts

My first key point here is to ensure that you have the absolute legal right and also the ability to improve things actually written down in black and white in your contracts.

All underpinning contracts and associated Service Level Agreements (SLAs) should contain enough scope (and the right words) to allow the recipient of the service the option to ask for internal processes to be improved or re-engineered in light of the challenges that you experience.

It took me some months and much legal wrangling to alter some of the wording in the SLAs and contract that my predecessors had willingly signed up to. But the effort was worth it because it gave me a solid foundation to be able to eliminate more Problems from.
You can use the following techniques to give you more of an idea about the types of improvements you may wish to make.

I found that with each Vendor there were three distinct stages of what I called “Vendor enlightenment”. They are: -

### Three Stages of Vendor Enlightenment

**STAGE 1:** Every Problem is tackled separately and distinctly, there is a limited view of trends, limited proactivity. This is costly!

**STAGE 2:** Aha! These Problems are related to a particular root cause – we can tackle one root cause and eliminate multiple problems!

**STAGE 3:** Eureka! As well as tackle the root cause, I can go ‘upstream’ and tackle the underlying reasons of the source of the actual problem, using this model, and drive down the potential for new problems to occur.

So by carefully plotting the enlightenment stage I was at with each and every Vendor, I found I could prioritize which sources of Problems to eliminate in the best order and so I was able to proactively close-down and prevent the majority of problems from occurring over time.

The key words I made sure were included in all contracts were: -

“The Vendor will work proactively with the organization to identify and eliminate both impacting and potentially impacting Problems to protect the defined Service Levels within this agreement.”

Twenty seven words in total. These proved a significant benefit to my efforts to eliminate Problems over time.

**SLA’s and Vendor Relationships are the Major Differentiating Factor**

There is an old saying, “you only get what you pay for”, but with Vendors you should not have to ‘pay’ (and by that I mean suffer) for what you get.

Therefore you need to proactively ‘get’ the flexibility, co-operation and buy-in to be able to work with your Vendor and make a real difference.

Something else that really worked well was the following approach: -
The S7VEN Sources of Problems

- Review all your contracts and SLA’s for relevancy
- Go out to tender to secure a better deal if you find a lack of co-operation or the results are not being delivered

There is nothing that sharpens up a Vendor’s responsiveness more than the possibility that they might lose your valuable business unless they begin to support your elimination efforts.

Going out to tender costs the Vendor time, effort and money. Such a re-investment is a real hassle which most Vendors can well do without. They also do not like the ‘risk’ involved with potentially losing your business to a competitor.

Look for other opportunities to influence your Vendors to work more like you do. Invite their key players to your internal briefings, offer to provide some one to one education session on new techniques for their people, share initiatives (such as excellence in Change Management), show the positive results these can have on minimizing downtime and increasing people’s active hours on projects and reducing costs of overtime. The closer you work together – the more infectious the whole Problem elimination programme becomes.

Implement a Vendor Scorecard Programme - Measure Vendor Performance based on Factual Evidence

Scorecards sound like a real load of time and hassle but a simple and effective scorecard bolted onto your weekly or monthly Vendor Service review meeting can make a BIG difference.

I can quickly show you how to create a simple but effective scorecard and achieve great results with it.

It’s based on some key principles: -

- Keep it simple and focused on what really matters
- Keep it factually based – use real life evidence
- Ensure the Vendor scores their own performance first
- Objectively score the Vendors performance
- Compare the scores and agree the final ratings

Here is a quick summary of how to get your scorecard up and running: -
How to create a simple, low over-head scorecard that works

a. Make a list of, say, six key topics that you wish to review with your vendor.

b. For each of the six topics, list up to 10 key questions that you want answering each week/month.

c. Prior to your review meeting - ask the Vendor to rate themselves and provide background explanations/evidence as to why they answered like they did, and bring this information along to the review meeting.

d. During your Vendor Services review meeting walk through the answers together and discuss the evidence and provide immediate feedback – rating or re-rating the Vendor as you go - using a scale of "0 - no evidence" through to "5 - World Class" and the varying degrees in between!

e. By the time the meeting’s over – you will have a more comprehensive, factual based set of results to be able to judge actual Vendor performance

f. You can then summarize your ratings in a formal e-mail to record performance. Use these scores and the average over the previous few meetings to form the foundation for the next meetings discussion. Remember - you are trying to drive improvements backed by sound evidence.

Note: - Ensure that one of the key topics is Problem Prevention and Elimination (not Problem Management – anyone can ‘manage’ problems, the goal is to prevent them!)

I rolled out an extremely successful programme with all of my Vendors over a twelve month period. We reached the stage where each Vendor was formally rated once a month.

Here is where it gets really interesting!

I compiled a kind of league table of Vendor performance scores with the best performing Vendor at the top of the table and the worst performing Vendor at the bottom. There were two league tables actually: one for high value Vendors and one for all the rest. Kind of a premiership and then a first division!

Every month I asked my boss to personally e-mail the highest performing and the most improved Vendor for that month with a thank-you note, explaining how their contribution was making a difference to our business and their future business.

Also, I got my boss to e-mail the worst performing Vendor and the Vendor that had dropped the most, to advise them of his concerns with their performance and what should be done next month to improve.
Any Vendor that got stayed at the top for three consecutive months would receive a phone call from our CEO to their CEO personally congratulating their team.

Any Vendor that stayed at the bottom for three consecutive months also received a phone call from our CEO.

**It was an absolute revelation!**

Vendors would work *really* hard to improve, stay at the top or get off the bottom.

This one technique alone was singularly responsible for a 120% jump in Vendor performance over 12 months – all at exactly the same annual Contract fees. It did not cost us one extra penny in revenue.

**Integrate Deeper with Vendor’s Processes**

For the more *astute* Service Provider - how about integrating your processes end-to-end with your vendors so that there is a smooth flow through of information and activity?

Roles and responsibilities are shared and known across the process flow. Teams work closely together to avoid wasting effort, reduce time to respond and reduce costs.

Additional “target times” to perform other related activities and the measurement of important key performance indicators are mutually agreed. These additions also underpin the Contractual agreement.

Such levels of deeper integration are only possible with the right relationship, contract and culture – however they are very effective in helping your organization prevent and/or eliminate problems faster.

**Involve Vendors More**

A final point on Vendors - but a key one. The more Vendors are embraced and made to feel like an active part of your organization – then the more they will ‘feel’ the responsibility of problem prevention. For example, do your Vendors attend your daily/weekly Incident Post-Mortem meetings? Do they attend your Problem Elimination meetings? Do they actively suggest new approaches to preventing problems? A significant culture shift is often required to enable such a participative approach but the benefits are huge.
Source #5 – User Error

The very people who use your IT Systems to deliver Service to your true end-customers are also a key source of your Problems.

I will use the label ‘User Error’ – you may prefer a different term for your own organization.

Your business folks are often under tremendous pressure to deliver ongoing results within their business lines. They have no time, little patience and absolutely no sympathy for technology or IT Systems.

Therefore when things go wrong (as they sometimes do!) they like to circumvent the usual support processes – like calling the Service Desk to report a fault, waiting for technical assistance and providing valuable information to assist your support services people.

They prefer to call their ex-colleagues from Support (who perhaps now work in the business themselves), try to fix the Problem themselves or worst of all – ignore it until it becomes absolutely critical!

I admit, I have painted a rather gloomy picture here and perhaps that’s a little unfair. But from my experience of business lines, where people spend tens of hours servicing customers, end-users are only too happy when their systems fail – they get to do other work or take an early break!

Remember – their business is effectively suffering from downtime and I’m pretty sure their boss won’t be happy with the situation.

So, against this background what are the typical reasons why Problems can occur in the “User Error” source?
The **S7VEN** Sources of **Problems**

**User Education** – or rather the lack of it!

Often within the project delivery lifecycle, testing and education are some of the first activities to be cut back. This means that due to over-runs in the design, build and (sometimes) testing stages – there’s little time to actually education end-users on their forthcoming new IT system.

As well as reducing the businesses ability to actually realize the benefits of the system – it leads to users making mistakes, “tinkering” and finding their own amazing “shortcuts” to get the end results that they require.

**There is really no substitute for quality education.**

It helps to prevent end-users circumventing how they are supposed to use their systems.

Positive things you can do here include: -

- Maintain strong linkages with the IT education team
- Gauge just how much education end users have already received
- Make education a mandatory requirement on the business side of your SLA’s
- Ensure a local ‘escalation’ procedure (where education gaps are apparent) exists to let end users raise questions and receive answers for new queries is known and in place

**Purposeful Documentation**

Users guides, cheat sheets, ‘what to do if this happens’, ‘where to get help’, ‘what to expect from IT Support’ – are all examples of local documentation that can be carefully placed on or near the end-users desks to prevent “tinkering” and ingenious workarounds being applied.

For example one guy I knew in a customer service function actually created his own set of highly complex report writing tools within Excel, because he thought that access to the correct report writing suite was only limited to his manager!

One day he told his manager about his new ‘system’ and to her complete amazement she didn’t ever stop to appreciate exactly HOW he was producing these reports. She immediately granted him access to the Report Suite.

*(Ironically, the Excel version did everything the team needed – so it stayed!)*
The Delivered System “Fails to Meet Specified Requirements”

This is more common than you perhaps realize. The system, as delivered manages to get through both testing as well as user approval and is accepted into production. However for some reason the system doesn’t do something that it’s supposed to do.

Some of the end-user community will always find an ingenious way around this. In my experience this is usually something relatively complex involving importing data and processing it against key dates or maybe something like exporting files between systems - because the overnight automated routine keeps failing and the system cannot automatically recover.

Regardless, you need to come to terms with the fact that end-users finding their own workarounds maybe perfectly fine in the very short term (sometimes as little as lasting for that particular working day) however this should be strictly a temporary situation. End-user local ‘workarounds’ like this are often “triggers” for new Incidents just lying there… waiting to impact production.

To try to overcome this common situation, you must try to determine where the system fails to meet requirements and drive activity to ensure this will be remedied. Once you know where – you can use many of the other approaches I have outlined so far – to begin to eliminate them.

Validation, Robustness and ‘Self-Healing’

It makes sense that the ‘stronger’ a system is (in terms of handling weird occurrences) then the less likely that you will see Problems with it further downstream. With the advent of object orientated coding methods, the ability of modules to self-correct and maximize the availability of the system is excellent. But what is object orientation is not a core capability of your organization?

You should consider taking direct action during the design stages of your new systems to ensure that the levels of robustness are where you would expect them to be. Balance this level against the required level of systems availability demanded by your business or SLA.

Here are some of the other actions, activities and approaches that got me some great results in this area:

1. Ensure there is really a problem to eliminate in the first place! – be careful not to expend large amounts of resources on ‘red herrings’!

2. Ensure education is not ‘axed’ – Stand up for quality education. Not only does it help to maximize the benefits of a system – but it prevents Problems too.
3. **Ensure systems are ‘fit for purpose’** – Check out the test results, work with user QA groups and also get a good ‘feel’ for the forthcoming system before it enters production. Influence the quality of the final solution BEFORE it enters production.

4. **Testing should include destructive tests that satisfy curious minds** – It’s guaranteed that once a system goes live it will be absolutely hammered with every possible combination of data/keyboard presses and all sorts of eventualities that were never thought of during testing. To this end, some up-front destructive testing should help to find some of the more serious conditions that might occur after go-live. You do destructive testing, right?

5. **Implement a ‘super-user’ programme** – best of all – maintain an excellent relationship with the business user community. Form partnerships with key folks who act as ‘super users’ for different systems. They will help you to differentiate between real problems and ‘red herrings’ as well as help you get access to the right people and information should a real problem arise.

*Next we will examine the penultimate source of problems #6 – How Production is executed.*

*In this source we explore how you run your production environment, look more closely at internal controls, and finish up with examining the overall effectiveness of your Problem Management process.*
Source #6 – How Production is Executed

This source looks directly at the actual way that Operations, Production Support and other maintenance and support function delivers the service you need to help ensure non-stop continuous operations.

The way we actually manage, control, change and improve our production environment directly correlates to the likely volume and frequency of new problems.

I will highlight many of the key areas that I worked hard, with my support teams, to raise the quality of the function (or documentation) in order to help identify and eliminate the sources of Problems stemming from our own production environment.

Stop/Start Recovery Procedures.

Do you know how to quickly and effectively ‘stop’ any of your systems or services?

Do you know how to quickly and effectively ‘start’ then again?

Failing to have adequate stop/start procedures for your core Infrastructure, job scheduling, databases and other application support mechanisms will ultimately lead to new problems. It’s all about being in control. Can your IT support teams effectively take down and then, after some remedial action has been taken, start service again?

It’s inevitable that you will need to be able to ‘stop’ and ‘start’ your systems or Infrastructure in an unplanned fashion from time to time. This could arise after a fault or other service outage. The ability to cleanly close down and then cleanly start up such components (or groups of components in the right sequence) will prevent Problems.

How do I know this? Well, after determining the root cause of thousands of Problems over five years, we identified that a significant number of them were actually related to how we ‘stopped’ and/or ‘started’ services incorrectly – particularly when trying to recover from another incident and restore service.

Support teams, probably due to lack of practice or inadequate documentation, would regularly and inadvertently ‘stop’ or ‘start’ a service incorrectly. This could have been an incorrect startup sequence, executing job that should have run in the incorrect order, or missing important data feeds since the service had been down for an hour.

The overall goal is minimum impact and minimum disruption. It is important to note that a significant proportion of new problems often stem from the way that service was taken down and/or restarted. We have counted numerous
times that either for routine (non service impacting) maintenance or during the life of a run of the mill incident, incorrectly executing ‘stop’ procedures or inadvertently executing the wrong start script without due consideration to the time of day and the end state of the systems, can lead to far greater problems than necessary.

Some helpful questions to get the answers to: -

- Have you mapped all your underpinning Infrastructure to a Service Map – so you know what needs to be available to drive your service?
- Have you clear and unambiguous stop/start procedures that are known and understood?
- Do your people know how and when to execute these procedures?
- Once service has been restored, are all data feeds and file transfers as you would expect them? Are standard health checks in place?

**Automated Tools and the Ability to Handle “Out of Line” Situations.**

Tools are great – they cut down on human intervention and help us to manage more effectively. However, the sheer plethora of tools and their in-built complexity often means that they “sprawl” out on their own and no-one really knows what’s happening with them, in terms of when they execute their own housekeeping, tidy-up routines and perform their core processing.

To this point, when problems occur and the root cause requires rapid identification and elimination – you really need to use the full power of your diagnostic tools to assist.

Some helpful questions to get the answers to: -

- Have you got the right tools, measuring and capturing the right things?
- Does your tool configuration proactively support your Problem Management process?
- Are your tools paying off? Do you spend more time ‘tuning’ them – than they do providing benefit for your environment?

**Impact of Changes on Production Schedules.**

The key point here is really simple. When aspects of service change (for example: process start time, the number of end-users, the introduction of new interfaces) does your underlying processing also change?
The **S7VEN** Sources of **Problems**

There are many occurrences where the business aspect changes, but someone back in Support or Operations forgets to (or is not informed) to re-schedule or re-design the overnight processing flows to accommodate these changes – resulting in more new Problems.

Some helpful questions to get to the bottom of:

- Are tools and scheduling products automatically linked to your change management process?
- Have you got your processing schedules really ‘nailed down’ and under control?
- If an overnight job fails, are clear re-start instructions in place for your teams to continue processing without the need to call for support?

**Human Error.**

The final area to explore in this source of problems – but perhaps the most important – is human error.

Human intervention by support teams, service teams and the business admin people can lead to procedures being executed incorrectly and errors made that cause new problems. It is critical that this source of problem is identified and the relevant person understands and knows ‘the error of their ways’. The biggest reason why people do not own up to human error is because their working environment suffers from a ‘blame culture’. Our advice is simple, eradicate any ‘blame culture’ and focus on a building a supportive culture with education to help people fully understand how to do things correctly first time.

Next we come to the seventh and final source of Problems. Here we explore **source #7 - "Failures"**, where I recommend some key actions to help minimize how this source can impact your production environment.
Source #7 – Failures

Now it’s time to examine the seventh and final source of Problems – Failures.

This is the source that you cannot ultimately prevent but you can certainly go a long way to mitigate it when a Problem does occur. Here’s the arrow:

In this context ‘failure’ means everything that could fail and have an impact on service (or your capability to provide service) that is not caused by any of the other sources of Problems, such as Changes, Upgrades, or Suppliers.

"Sometimes Things Just Fail"

That's why we have maintenance contracts. That's why we have health insurance come to think of it!

There are a number of areas where failures can directly impact service provision and reduce availability:

- Mechanical or hardware failure
- Data transfer or data formatting failures
- Logic failures
- Processing schedule failure
- Power failures

Now, there will often be a 'cause and effect' challenge; for example - the air conditioning system would not have failed if we had realized that there was an
extra load on it due to 100 new business users starting in that building last week.

So what can you really do about failures?

There are several working practices that can be adopted to help relieve the pressure of new problems coming from this particular source.

**Preventative Maintenance Schedules.**

In today’s 7 X 24 non-stop service environments it is extremely difficult to request any unplanned downtime or be able to proactively switch over to stand-by systems. The risk of some kind of failure is just too great. Plus, probably the last time you tried it – there was some kind of Incident as a result. This was probably due to lack of ‘stop / start’ procedures or a general lack of experience dealing with the complexities of taking down multiple services and re-starting them later. So getting any kind of ‘air gap’ to perform preventative maintenance is a challenge.

For every component within your Infrastructure you should always be able to determine its ‘health’, its current status and when it might be due a check-up from the maintenance teams. Most Infrastructure components already come delivered with a pre-determined maintenance schedule but it’s up to you to ensure that this gets performed.

My advice in this area is to create a master preventative maintenance schedule on a calendar that’s tightly controlled and managed. Then try to group sets of components together, based on how available you require them to be and when the best time to perform maintenance actually is. This will lead you to well defined time slots when you would like your maintenance to be performed. Whoever controls the master schedule MUST have a close working relationship with the Change Management team. Remember, the Change Manager will closely guard and protect his “One Version of the Truth” Forward Schedule of Change and so they will be able to advise and help you protect your pre-defined maintenance windows.

Also, you should talk regularly to your facilities (I’m thinking power and building site services here) team about their schedules too. Why not combine schedules to work effectively together to minimize overall downtime to the business?

**Manufacturers root cause analysis, including faulty batch validation.**

It's amazing how many times you have experience an Incident - say due to a simple power supply failure in a Router – and you quickly find out that you’re not the first, and you’re not alone!
Manufacturers and systems integrators occasionally suffer from potential mass failures - and sometimes need to perform an emergency field upgrade to prevent the further failures or replace suspect components.

It is important when these occur to check your full inventory for any other similar / related items and also ask the Vendor / Manufacturer which ‘batch’ did those failing items come from, what is the root cause of the problem - and most importantly - will the new item work? How do they know? Has it been bench tested first? What if the same symptoms occur again?

Ask tough questions - because it is your production environment and the availability of your systems that’s at stake.

“Where Else Could This Failure Occur?”

A very simple question - but it is often effective at obtaining an answer that helps to prevent further failures.

When you are impacted by a failure - ask yourself this question. Make sure that you take positive action and follow through correctly to prevent similar failures elsewhere.

For example if you have four highly redundant, all singing, all dancing super powerful Routers and one suffers a power supply failure and shorts the entire backplane - taking the whole service down – ask yourself, “What ensures that the other three won’t suffer the same sometime soon?”

Please do not lose this important point as everyone is scrambling around trying to restore normal service. Learn to think ‘wider’ and logically about what’s happened. Keep asking those tough questions too!

After such severe failures, Vendors may try to give you a ‘hundred reasons’ why it was just a ”one in a million” - but was it really?

Is it time to ensure that they safely replace the other three power supplies or maybe bring additional spares on site – just in case?

Has the Vendor run checks in their fault database to see how many other power supplies of this type have failed?

**Proactive Failure Prevention Programme**

You will be pleasantly surprised if you just call in your top support people, along with the facilities team and some of your key vendors and just "brainstorm" this topic for a couple of hours.
With the right facilitation and ‘tone’ for the meeting (I.E. no Blame culture!) I suspect they will tell you a few home truths about cable infrastructure, labeling accuracy, items that failed maintenance and have not yet been replaced or re-checked. Its real-life and these things happen.

What you hear can all sound quite scary – but it’s important to have the concerns and ‘under the carpet’ issues out in the open and clearly discussed.

Getting everyone together and creating the right atmosphere where people can contribute and help to identify the most likely items to fail - is revealing but ultimately a key part of your overall elimination programme.

**Know Your S.P.O.F’s and mitigate or eliminate them**

I have already covered a lot about SPOF’s earlier. You must identify, hunt down and eliminate (or mitigate) single points of failure, where you can afford to do so - or – formally accept the risk of not eliminating them.

Single points of failure can also be at a logical, rather than a physical level, especially with network configurations and databases. So, keep your perspective pretty broad on this topic.
How To Execute Your Problem Elimination Program

As we reach the end of Part 1 of my journey across the Seven Sources model you may have noticed throughout this book that I have paid particular attention to the “human” elements of Problem elimination. Amongst other observations, I have outlined ‘blame’ culture, the importance of education as well as the fact that Vendors are only human too.

This is for very good reason because the absolute foundation of any Problem elimination program is “people”.

I have always found that it is People that make the fundamental difference between mediocrity and success.

As I continued my journey of Problem elimination over my five years, I have always paid particular attention to the humanistic side of success, by leveraging the many different facets of how human beings work, behave and achieve results.

I am certainly no master coach nor do I claim to have any magical power over people – far from it – but I have built up a useful portfolio of what works well, in the real world, to eliminate Problems.

I can break this down for you much further and share some of the many real world tactics that worked for me. I hope the appropriate ones work for you.

How to get the most out of your people

➔ People enjoy feeling motivated – so set aside some quality time each week with your team to share success and failures, learn lessons and reach agreement on how things will work differently next time. Ensure you maintain a stead incoming pipeline of new ‘challenges’ – so as soon as one Problem is eliminated you have other (slightly lower priority one’s) to tackle. Ensure that you reward success with anything from a straightforward ‘pat on the back’ in front of the whole team, through to a small bonus to let someone be able to take their partner out for dinner. Sometimes, it’s these small things that carry most weight and are appreciated!

➔ Ensure that people understand that you are investing in them. Not only your time, but company funds too. By this I am referring to education and development time in particular. IT and Technology people have a much higher desire to continue their development than many other professions and looking forward to a new training course, certification (like ITIL Foundation) or other development activity also builds and maintains motivation.

➔ Deliberately demonstrate to people that they have your ongoing management support. There are many opportunities across a week to show your support and appreciation. When people feel supported they also find it
The S7VEN Sources of Problems

easier to move into their own 'empowerment zone' and achieve higher results, take the initiative more and put their talents to best use. Remember – over a sustained period of eliminating Problems – you are going to need the maximum amount of effort from everyone involved.

➤ Make sure that results and achievements are rewarded fairly when it comes to annual appraisal time. This is an ideal opportunity to discuss performance, improvement areas and new opportunities to expand and grow. You should always be looking for your elimination team members to be developing further. The secret to effective appraisal delivery is always openness, honesty and factual based feedback. There should also NEVER be any surprises in there! If you have held back on telling someone about their poor performance – then you are failing as a leader. Poor performance should always be addressed as soon as it is identified – in the correct way - one to one, face to face, and straight to the point – always focusing on the facts.

➤ It’s paramount that you only hire and bring on board quality people that have the right attitude, the right aptitude towards continuous improvement and that they are committed to making a real difference. The most important thing you do as a leader is hire the right people. Take quality time to identify, interview and properly assess potential new team members. Only hire if you feel that special inner feeling that this person is going to make a difference to your overall efforts. If you hire mediocrity – you get less than average results. It’s as simple as that.

➤ You should avoid people burnout and stress at all times to protect performance in the long run – as well as that person’s health. A range of actions can prove beneficial here: from ensuring your people all take vacation at regular intervals (to re-charge their inner battery’s and mental wellbeing) through to planning job rotations in advance to give people a rest from the pressures on working on the front line. In the real world – a change really is as good as a rest. From example, why not rotate your best Help Desk analyst for two weeks into the Change team. The break from taking incoming calls and processing e-mails all day will be refreshing. The challenge of working in a new way, with new people and taking on new responsibilities will also broaden that person’s horizons more. If you have never tried rotating people – find a suitable candidate and run a pilot. I believe you will be pleased with the results.

➤ When resources are scarce and everyone’s really busy with their day job – but you need a team to urgently eliminate a Problem – then consider forming a temporary “taskforce”. Special cross-functional taskforces, consisting of people from different teams, if managed correctly with the right terms of reference and management support, can deliver amazing results. I have consistently used taskforces over the years and the achievements delivered have never ceased to amaze me. There is an in-built power in breaking down silo’s, communication barriers and enabling like minded individuals to work together with a special ‘goal’. Taskforces are always empowered to go wherever they need to go, and talk to whoever they
need to talk to – to obtain the answers to the tough questions in order to systematically eliminate Problems. Again, if this concept is unusual in your organization, try a pilot. Deliberately select a Problem that you know the team can tackle well and is within their comfort zone. Run this taskforce yourself as a mini-project with an agreed set of deliverables set up front with senior stakeholders. Maintain regular, daily contact and ensure that the team is fully supported. Plan right the way through to benefits realization – which in this case means the Problem is eliminated and you have saved money, prevented impacts or enabled someone to be freed up to work more pro-actively. Celebrate the results and go slightly overboard with how many people you share the success with. Create a ‘buzz’ and let it be seen that working on temporary taskforces brings positive exposure and is an opportunity not to be missed.

Some final things that work...

Before I close out on part one, I just wanted to share with you some miscellaneous tactics and thoughts that never made it into this eBook in their own right but are worthy of a special mention.

- Providing education on “How to eliminate Problems” is often overlooked. Do your people really know HOW to go about eliminating Problems correctly? What previous experiences (and bad habits) have they come armed with? Can you run a one day course on the techniques, approaches and concepts of effectively eliminating Problems? This should not only include the traditional “fish bone” diagrams and “the five whys” but also the concepts that I have outlined in this eBook that are appropriate for your environment. I know this sounds elementary – but you run a quick straw poll about how many folks HAVE ever received rudimentary elimination education – I think you will be surprised.

- Do not under-estimate the power in “internal” marketing. By this, I mean how useful it is to present back the results that the teams are achieving to senior managers and executives. How else will they really know what’s go on – and how well the team have delivered – if you don’t sit down and provide feedback? I have always ensured that the leader of any taskforces I ever sponsored (or run directly) was given the opportunity to present the success stories to executives directly. Although I was there to support them and handle any tough questions – they really gained an advantage from the exposure and learned from the whole process. Additionally, the executive would be extremely pleased that we had (at last!) eliminated some long standing Problem from production. Think about it – they are being provided with GREAT news and information to share with their executive peers – why wouldn’t they!

- Sometimes you have to say “no”… sorry, I mean “Yes, However…” when asked to form a new task force to tackle a Problem. This is usually because the teams are at maximum capacity and you are totally focused on
higher priority challenges. It’s only fair to everyone that they are allowed quality time to achieve results. Sometimes, some taskforces became so successful that executives would stop one of the taskforce leaders in the office and essentially ‘kidnap’ them onto their own ‘pet’ Problems. If those Problems carry more weight or have a greater negative impact on the business should a recurrence happen – fine.

But nine times out of ten, these ‘pet’ projects do not. You need to advise your teams of this. Don’t allow yourself to become a ‘victim of your own success’.

**One last thing…for now**

As a final point on the Seven Sources model, you should consider the initiatives, tactics and approaches I have explained **holistically**, rather than in silos across the individual sources or indeed across individual teams.

It is extremely rare for a Problem to only ever involved just one team, one piece of Infrastructure or one location. Today’s services are end-to-end and involve many many different components, locations, teams and customers.

Therefore an integrated programme approach, where you keep your costs low by improving the way that your support people currently work, is highly desirable and to be recommended.

Everything I have explained in this eBook is obviously not ‘rocket science’ and I can hear the cries of, “oh it’s only common sense”, right now!

But unfortunately with time and resource constraints, lack of support and a distinct lack of vision – many of these are actually quite uncommon – in the real world.

If you don’t think it’s necessary to eliminate your Problems – just ask yourself: “What happens if we don’t eliminate our Problems?”

**Problems are not like wine…**

**…they don’t get better with age!**