**ENTERPRISE RELEASE MANAGEMENT**

*A holistic approach to the challenges of complex release management*

<table>
<thead>
<tr>
<th>Author</th>
<th>ROBERT SPENCER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reviewers</td>
<td>STEVE GREEN – VERALUM CONSULTING</td>
</tr>
<tr>
<td></td>
<td>ITSMF UK SERVICE TRANSITION SIG</td>
</tr>
<tr>
<td>Date</td>
<td>30TH JULY 2014</td>
</tr>
</tbody>
</table>
ENTERPRISE RELEASE MANAGEMENT

A holistic approach to the challenges of complex release management

Contents
Executive Summary ........................................................................................................................................... 2
Introduction .................................................................................................................................................. 2
Section 1: What is Enterprise Release Management? ........................................................................... 3
Section 2: Understanding multi-stream inputs ...................................................................................... 5
  Horizontal .................................................................................................................................................. 5
  Vertical: Process & Governance for each stream .................................................................................. 6
    The Last Mile ......................................................................................................................................... 7
Possible Additions ....................................................................................................................................... 7
Who benefits? ............................................................................................................................................... 7
Section 3: Planning releases across all streams ..................................................................................... 9
  Constructing the HLRS ........................................................................................................................... 11
Section 4: Planning complex Deployments ............................................................................................ 12
  Don’t over-complicate it if you don’t have to ......................................................................................... 12
  Complex Deployments .......................................................................................................................... 12
Section 5: ELS Considerations ................................................................................................................ 14
Summary ..................................................................................................................................................... 16
About the author ......................................................................................................................................... 16

Appendix A.1 – Example Enterprise Transition Map ............................................................................. 18
Appendix A.2 – Example high level release schedules ......................................................................... 19
Appendix A.3 – Example complex cutover plan ..................................................................................... 20
EXECUTIVE SUMMARY

This whitepaper introduces the concept of Enterprise Release Management for CIOs, IT Operations Leads and senior Transition Managers. It provides an easily accessible approach and structure for those who regularly experience the effects of frequent, concurrent releases across several streams (whether project delivery, infrastructure, patching, defect fix etc.)

The problems addressed in this whitepaper are:

- Understanding complex, multi-originator IT change landscapes and their respective governance processes
- Visibility of all Production releases in one place (since programme plans will be scope limited and operational change schedules are often neither covering non-prod, nor engaged early enough)
- Managing contention of IT resources for overlapping releases through build and test
- Managing release collisions either through avoidance (long range visibility) or dealing with complex cutover windows

Solutions to these problems are often not trivial, but this paper includes some example plan formats to help your organisation achieve quick wins as well as discussing the concepts which underpin them. Some references to ITIL ® are made, however knowledge of ITIL is not a prerequisite.

Organisations which adopt Enterprise Release Management principles can expect to benefit from increased confidence in managing & articulating the delivery roadmap to stakeholders, increase their efficiency in managing the path to production and ultimately making better-informed decisions when balancing resources and service stability with responsiveness to changing business demands.

INTRODUCTION

Enterprise Release Management (ERM) isn’t a new concept, or even a new term. However, it is becoming more prominent, and it is starting to enter common usage amongst senior IT managers. Enterprise Release Management is also not an ITIL term, but it spans many of the Service Transition processes and capabilities, and is relevant for the context of complex and frequent project delivery, whilst supporting simpler technical and maintenance releases.

Many organisations already have Release Managers, and the job description of a ‘Release Manager’ can vary wildly from one organisation to the other, but very few have an ERM person or function.

This whitepaper offers a definition for Enterprise Release Management and describes some contextual activities, tools and techniques.
SECTION 1: WHAT IS ENTERPRISE RELEASE MANAGEMENT?

The Enterprise Release Management (ERM) function is characterised primarily by the complex range, wide scope, and input variety of releases under its control.

It is less concerned with specific technologies, business areas or projects, and more with giving your IT organisation the forward view of all releases, and highlighting their integration points and dependencies which then allows:

- **Better understanding of the overall effect** that releases in these areas have on service provision to the business
- **More accurate view of likely impact** on Service Operations and the business / customers
- **Improved resource planning**, particularly with shared assets in the operational and delivery sides of the organisation, such as:
  - People & Functions (eg. developers and testers, ServiceDesk, App Support)
  - Infrastructure (eg. shared environments, common platforms, tooling)
  - Service Warranty (Availability, Capacity, Continuity & Security)

The following table provides some characteristics of three types of Release Management, Enterprise, Technical and Delivery, and shows where they differ in remit.

<table>
<thead>
<tr>
<th>Type</th>
<th>Focus on</th>
<th>Role overlaps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical Release Management</td>
<td>Build and configuration of deliverables, predominantly code assets.</td>
<td>Build Manager&lt;br&gt;Configuration manager (code)&lt;br&gt;Lead Developer&lt;br&gt;Environment Manager&lt;br&gt;Test Manager&lt;br&gt;Application Packager</td>
</tr>
<tr>
<td></td>
<td>May specialise in a specific technology or functional area.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Related ITIL Processes:</strong>&lt;br&gt;Release &amp; Deployment Management&lt;br&gt;Service Asset &amp; Config. Management&lt;br&gt;Change Management</td>
<td></td>
</tr>
<tr>
<td>Delivery Release Management</td>
<td>Concerned with the delivery of entire releases, bundles of changes including software, hardware, infrastructure configuration changes as part of an overall project or programme stream.</td>
<td>Project Manager&lt;br&gt;Delivery Manager&lt;br&gt;Programme/Project Release Manager&lt;br&gt;Environment Manager</td>
</tr>
<tr>
<td></td>
<td>Loyalties usually lie with the delivery side of the organisation (rather than the receiving or operational side) and ensuring timely, complete closure of each delivery including ELS entry and a clean cutover (deployment).</td>
<td></td>
</tr>
</tbody>
</table>
Related ITIL Processes:
Transition Planning and Support (TP&S)
Strategy Management (Service Strategy)
Design Coordination (Service Design)
Release & Deployment Management
Change Management

Enterprise Release Management

Overall visibility of multiple release streams and sub-streams (programme delivery, maintenance and patching, infrastructure & platform etc.)

Management and facilitation of release contention issues, whether competing for resources or windows, and ensures release packages smoothly transition into operations as a whole.

Should be able to interact and lead engagements with delivery and operations to ensure the best coverage.

Related ITIL Processes:
Change Management
Transition Planning and Support (TP&S)
Release & Deployment Management
Service Validation and Testing

Release Manager (Operational)
Transition Manager
Environment Manager
Head of Transition
Change Manager (Senior/Strategic rather than operational/technical)

Whilst competition for resources is often felt to be a Project or Programme issue, its effects can be felt after go-live if not addressed early, and reviewed often.

Enterprise Release Management seeks to understand and manage clearly scoped deliveries from all streams targeting production. Its focus is:

- Long range multi-stream planning
- Managing the usage of environments (including shared non-production environments)
- Planning and execution of complex, interdependent deployments
- Non-functional testing and release-level regression testing
- Gate Management in support of Operational Readiness and Service Acceptance

Enterprise Release Management therefore fulfils an important role: helping Service Operations to identify and manage the consequences of high volume, high complexity & highly integrated IT change.

This will deliver real benefits to both the delivery and operational sides of IT – allowing both to be confident about the quality of releases through early visibility, integrated impact assessments, efficient resource utilisation and regular bi-directional engagement.
SECTION 2: UNDERSTANDING MULTI-STREAM INPUTS

ERM is concerned with managing multi-stream releases from disparate sources. It is important for each organisation to understand what these sources will be, the nature of changes / releases expected from them, and the route that each change / release type will take through your release process. This can be shown in an Enterprise Transition Map which is a governance overview that shows all routes to production, from start to finish (going top to bottom) and from low complexity to high (going left to right).

It often begins as a simplified view, and over time, it evolves to become more comprehensive. Once built, it may not need to change significantly, but it should be periodically reviewed for accuracy (as should all your governance / process models).

The transition map is based around 2 simple axes:

Horizontal

This is the ‘spectrum’ of change in your organisation. It divides the map into process streams, each representing one or more change types split by complexity / impact, functional area or even technology as shown in the full example in Appendix A.1.

Begin by adding the basic change streams as shown in the excerpt below; a new stream typically means a different way of processing that type of change.

![Adding streams to your map](image-url)
Simple, low risk and repeatable change, such as Standard & Pre-defined Changes, can be situated at the far left of the map. Emergency change is located at the far right (due to the inherent risk and breadth of scope possible within an emergency change).

Across the centre two thirds is ‘Normal’ change. As mentioned, this is sub-divided into streams showing different complexities, input streams (eg. Simple Technical through to Major Project Delivery) or even functional/business area.

Note: In a SIAM (Service Integration and Management) organisation, each service tower can also be represented here by one or more transition input streams - assuming that the provider delivers change into the host organisation via an alternative process.

**Vertical: Process & Governance for each stream**

With the basic structure and input streams identified, detail can be added to each one to show the (linear) process and governance gateways.

The excerpt on the right shows two streams of change:

A stream for **Vendor patching** (eg. Oracle or SAP patching) which may need validation from a technical design function (aka Design Board). This will require a tactical Change Advisory Board (CAB), or an alternative dev / build approval forum, to give the go ahead to load the patches into a non-production environment to build and test the release.

A stream for **minor functional fix** (ie. simple bugfixes). These may need to be reviewed at a Design Board, but they can typically be built and tested straight away.

Other streams on the example map in A.1 show both simpler and more complex exchange streams, including governance activities such as Financial approval, steering committees and strategic approval boards.
The Last Mile
Near the bottom of the map lies the final governance gateway (typically a Production CAB). In some circumstances, this can be bypassed when showing Standard Changes and Service Requests, and in the below diagram, it becomes an emergency CAB on the far right to support your emergency change stream.

There is often then a final area of governance covering Early Life Support, Operations Acceptance / Handover and (for some releases) Benefit Realisation.

![Diagram](image)

Possible Additions

With processes added more details can be added to your streams, such as:

- Which tools are used to track requests at various stages
- When (and on which streams) additional processes (such as Transition Planning & Support, Release & Deployment Management, Service Validation and Testing) engage
- Who key stakeholders are, for example: the service owner, service level manager, business process/service owner.

These details haven’t been added to the example here, but a personalised version should be enhanced with any information that will add value and avoid confusion… but make sure it does not become too bloated.

Who benefits?

Initially, the transition map is a useful tool for the Enterprise Release Manager to understand the full spectrum of change that he or she will need to deal with. It provides clarity not only on the routes releases are utilising, but also the key governance steps towards production.

Business Relationship Managers can use it to articulate transition governance processes to their customers. The Operations and Delivery functions within your organisation can use it to understand the terms of reference and engagement between each other.

As service management matures within your organisation, the map gives you a framework for the scope of your Service Transition policy. It can be evolved to show all relevant processes within Service
Transition, either in detail (which could lead to clutter on the map) or at specific key engagement / decision points.

Additional benefits include:

- Quick recognition of input streams which have either too much or too little governance
- Identification of opportunities to unify and harmonise disparate change management processes across several streams by spotting existing similarities between them
- Highlighting points of integration that need strengthening.
SECTION 3: PLANNING RELEASES ACROSS ALL STREAMS

Once you have an understanding of the various streams delivering change, releases from within each stream can be plotted against a longer range timeline to give you a High Level Release Schedule (HLRS).

The HLRS is an overview of your release roadmap and will enable you to give answers to questions on release windows, environment and resource utilisation. It should be simple enough to give senior stakeholders an easy view of upcoming releases, but with enough detail to provide value for project managers, environment managers and technical/development/application managers to inform their work.

The HLRS is straightforward to construct, following a layout which can be easily realised in either a spreadsheet or dedicated project tooling with the following key characteristics:

- Timescale is typically weekly rather than daily
- Timeline is normally from 6-12 months, but can be extended with lower levels of detail
- Each release only takes up one line
- Releases are grouped into their streams which may relate to the streams on the transition map
- The release line can show the governance phases and selected milestones (see example above)
- It should also show critical IT and business events which could impact the release schedule such as change freezes, periodic reporting, external/external audits and periods where there are expected to be high levels of business activity or low levels of staffing
One of the biggest benefits of having such a high level release schedule is being able to clearly identify upcoming time periods with multiple deployment events. These may impact one another and cause collisions, or at the very least, make for an interesting cutover - something covered in the next section.

In the example below, two project releases, some system patching and a network hardware refresh are all planned for the same week. This may or may not cause problems for an organisation, but the key is being able to spot this far enough in advance so that the right questions are asked and serious consequences of conflicting releases are avoided.

This is arguably a role which the Change Management function could perform via the Change Schedule, but it’s worth highlighting that many Change Schedules show only Production activity, and without enough notice for strategic interventions. Change Management won’t help you identify contention in some areas of shared resources, such as test teams and shared test environments, and will often fail to support anything other than tactical mitigation for late identified issues. In addition to this, your Project Managers or Project Management Office (PMO) will often hold the delivery side plans, without showing all indirect operational activity that can affect them or their cutovers.

A Planning Change Request process (and Advisory Board) can assist with governance of the HLRS, and ensure it is reviewed regularly by IT (and business) stakeholders. This can help inform, or even prompt, the formation of a more strategic board which involves senior management and takes decisions affecting the entire IT roadmap, and reviews the issues highlighted by the HLRS that could not be resolved without escalation.
Constructing the HLRS

The example at the starts of this section shows a two-dimensional model for the plans (a full example of which is in Appendix A.2), and as mentioned can easily be constructed in a spreadsheet or project Gantt chart application. However, dedicated tooling which supports pivot data (ie. relational tables) should be considered when managing multiple releases by multiple release managers. This enables custom views, such as an environment-centric view of release activities (see below):

<table>
<thead>
<tr>
<th>Environment</th>
<th>Release</th>
<th>06 Jan</th>
<th>13 Jan</th>
<th>20 Jan</th>
<th>27 Jan</th>
<th>03 Feb</th>
<th>10 Feb</th>
<th>17 Feb</th>
<th>24 Feb</th>
<th>03 Mar</th>
<th>10 Mar</th>
<th>17 Mar</th>
<th>24 Mar</th>
<th>31 Mar</th>
<th>07 Apr</th>
<th>14 Apr</th>
<th>21 Apr</th>
<th>28 Apr</th>
</tr>
</thead>
<tbody>
<tr>
<td>SysInt 1</td>
<td>Vendor Patching</td>
<td>Q1 nonProd</td>
<td>Build for 1</td>
<td>Build for R1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Payroll Project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ERP Project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PerfTest 1</td>
<td>Vendor Patching</td>
<td>Q1 nonProd</td>
<td>PSV</td>
<td>PSV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Payroll Project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ERP Project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UAT 1</td>
<td>Vendor Patching</td>
<td>Q1 nonProd</td>
<td>UAT</td>
<td>UAT</td>
<td>CAT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Payroll Project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ERP Project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Pivoted view of the High Level Release Schedule showing releases grouped by environment

This dedicated tooling can be developed internally, if skills exist or the organisation can go to market to use existing applications which allow this level of relational data in a release management context, such as Service Now’s Release module, Plutora Release Manager, Xebia Labs XL-Release, Serena Release Manager and Aha!, among others.
SECTION 4: PLANNING COMPLEX DEPLOYMENTS

Eventually, every release needs to be deployed to a production environment. Good practice (and common sense) dictates that this should be planned. Ideally, the end to end deployment should have been tested (or rehearsed) in a clean environment first (such as a Pre-Production environment).

Don’t over-complicate it if you don’t have to

As most IT departments know, simple releases to single systems can be planned using a simple deployment plan. This should detail who is doing to what, to where and in which order.

<table>
<thead>
<tr>
<th>Step</th>
<th>Task</th>
<th>Team</th>
<th>Duration</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Check monitoring for issues</td>
<td>RelMan</td>
<td>5 min</td>
<td>Not started</td>
</tr>
<tr>
<td>2</td>
<td>Enable Maintenance page</td>
<td>SysEng</td>
<td>5 min</td>
<td>Not started</td>
</tr>
<tr>
<td>3</td>
<td>Deploy DB Schema changes</td>
<td>DBA</td>
<td>20 min</td>
<td>Not started</td>
</tr>
<tr>
<td>4</td>
<td>Deploy Application Binaries</td>
<td>SysEng</td>
<td>15 min</td>
<td>Not started</td>
</tr>
<tr>
<td>5</td>
<td>Commission testing</td>
<td>QA Team</td>
<td>30 min</td>
<td>Not started</td>
</tr>
<tr>
<td>6</td>
<td>Deactivate maintenance page</td>
<td>SysEng</td>
<td>5 min</td>
<td>Not started</td>
</tr>
<tr>
<td>7</td>
<td>Check monitoring</td>
<td>RelMan</td>
<td>5 min</td>
<td>Not started</td>
</tr>
<tr>
<td>8</td>
<td>Hand back to Ops</td>
<td>RelMan</td>
<td>5 min</td>
<td>Not started</td>
</tr>
</tbody>
</table>

Simpler deployment/cutover plan

Complex Deployments

Complex releases often require both a detailed deployment plan showing the individual steps to be performed, and a higher level overview plan showing the overall release(s) at a glance. This is relevant where a release spans multiple systems, thousands of component changes, or is expected to last longer than a single day, and will sometimes involve more than one line from the HLRS coming together in a single release window.

The high level overview plan should show:

- System Availability for each system impacted
- Release Managers, on-call duty managers and technical / functional subject matter experts
- Deployment activity by team or by technology area.
- Milestones (checkpoints, remediation points-of-no-return, conference calls, comms to business or vendors etc.)
- Remediation profile: how you will ensure you can recover at given points in time (eg. revert to snapshot)

The example in Appendix A.3 shows an example deployment plan for two unrelated releases (Release 1 and Release 2) which impact two unrelated systems (System 1 and System 2). These releases are being managed together in the same release window; in this case, a weekend.
This high level snapshot of the entire weekend is a useful at-a-glance reference for anyone coming into the release cold, as it allows them to quickly understand the timeline. It doesn’t replace a detailed, line by line plan with instructions on who is doing what to which system and in which order.
SECTION 5: ELS CONSIDERATIONS

Early Life Support (ELS) is the period when the people who delivered the change / release are responsible or accountable for its support and operation. Some organisations also refer to this period as ‘HyperCare’ although this term can also mean the first part of ELS whilst stabilisation occurs. Early Life Support replaces the ITIL v2 terminology, ‘Release Warranty’.

This paper is not focused on Early Life Support, but the Enterprise Release Manager must be involved in the planning and running of this key phase, to ensure the impact of any delay or additional instability is made aware across the rest of the HLRS. They will also be involved in the scheduling of additional releases required during the ELS period.

Typically, the responsible people for ELS will be a project or programme function, although day to day accountability can reside elsewhere. The ELS period is usually the time when the project completes its handover to operation, and may occur in a staged manner, eg:

<table>
<thead>
<tr>
<th>Week</th>
<th>1st Line</th>
<th>2nd Line</th>
<th>3rd Line</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ServiceDesk + Project Temps</td>
<td>Project FloorWalkers</td>
<td>Project &amp; Vendor</td>
</tr>
<tr>
<td>2</td>
<td>ServiceDesk + Project Temps</td>
<td>OnSite Support +FW</td>
<td>Project &amp; Vendor</td>
</tr>
<tr>
<td>3</td>
<td>ServiceDesk</td>
<td>OnSite Support +FW</td>
<td>App Support &amp; Vendor</td>
</tr>
</tbody>
</table>

Table showing staged handover during ELS where the project provides additional ServiceDesk staff and local floor-walkers to supplement BAU support/operations

ELS activities that the Enterprise Release Manager can assist with include:

- Work-Off Plan scheduling and its cutover planning
- Entry and Exit criteria, including acceptance criteria
- Route to production for issues identified during ELS (ie. will they all be Emergency Changes? Will you have a short-burst daily release cycle for the first week?)
- Expected duration, and what can trigger an extension
- Stakeholders and decision makers (the business must be involved)
- Meeting structure and attendance (eg Sunrise/ Sunset daily meeting)
- Resourcing contention between other activities and the project / programme under ELS

The challenges for the Enterprise Release Manager in dealing with multiple ELS periods for diverse releases are:
• Understanding clearly the accountability for support and maintenance and ensuring that the receiving support / operations teams are well supported in the new or changed services
• Dealing with a higher-than normal rate of urgent or emergency changes post-release and being able to accurately and quickly identify their impacts on both the business and operations
• Ensuring that functionally-integrated changes are properly assessed and managed

The ERM does not drive ELS, they are not accountable or responsible for it, but if you want to understand how issues during ELS impact other activities in your organisation, they are critical.
SUMMARY

Enterprise Release Management is not new. Most of the tools and techniques discussed are already used within many organisations and a lot of activities are recognisable as project manager or change manager activities. But the Enterprise Release Manager has the breadth of visibility (across all delivery and operational streams) of the change manager, with the depth of visibility required by delivery / project managers in tracking each release from start to finish.

This merging of perspectives gives the Enterprise Release Manager a unique platform to influence the quality of deliveries across the environments and ultimately help IT Management to make informed decisions about stability of service and responsiveness to changing business demands, without sacrificing either.

In summary, key benefits of the Enterprise Release Manager are:

- Better awareness of releases from all areas of IT, early enough to influence scheduling and resolve conflicts
- Better understanding of the overall effect of releases on each other
- More holistic view of likely impact
- Improved input to shared resource planning
- Tighter links between delivery and operational planning

All of this can be achieved by knitting together manually the various teams across your organisation, via integrated processes and numerous touch-points, but by giving cross-platform, cross-technology, cross-stream responsibility to a single individual or function you will ensure those communication channels stay open. Your organisation will then have a single version of the truth when it comes to releases, something that can only become more important as IT solutions become more integrated whilst their delivery channels become more diversified.

ABOUT THE AUTHOR

Robert Spencer has been working in Service Transition since 2001. He has managed change & release teams delivering high volume and high complexity change whilst helping public and private organisations (including FTSE100 & Fortune 500 companies) understand and improve their Service Transition practices.

You can follow him on twitter via @changerelease.
APPENDIX A.1 – EXAMPLE ENTERPRISE TRANSITION MAP

- Standard/pre-defined changes
  - Create RFC
  - Design Board
  - Tactical CAB
  - Build
  - Test
  - Perf. Test
  - UAT/OAT
  - Rehearsal
- Minor technical changes
  - Create RFC
  - Design Board
  - Build
  - Test
- Vendor patching
  - Create RFC
  - Design Board
  - Build
  - Test
- Minor functional fixes
  - Create RFC
  - Design Board
  - Build
  - Test
- Major technical changes
  - Create RFC
  - Strategy Board
  - Design Board
  - Tactical CAB
  - Build
  - Test
  - Perf. Test
  - UAT/OAT
  - Rehearsal
- Small project deliveries
  - Create RFC
  - Strategy Board
  - Design Board
  - Build
  - Test
  - Perf. Test
  - UAT/OAT
  - Rehearsal
- Strategic/major project deliveries
  - Create RFC
  - Strategy Board
  - Steering Committee
  - Finance Approval
- Emergency Changes
  - Create RFC
  - Build
  - Test

No CAB Required
Production CAB
Emergency CAB

Early Life Support, Operations Handover, Benefit Realisation
## Appendix A.2 – Example High Level Release Schedules

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>External events</td>
<td>Freeses</td>
<td>H1 compliance audit</td>
<td>EU Holidays</td>
<td>H2 compliance audit</td>
<td>Xmas Freeze</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audits</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Business</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Payroll Delivery 1</td>
<td>Accept</td>
<td>Build</td>
<td>Accept</td>
<td>Deploy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Payroll Delivery 2</td>
<td>Accept</td>
<td>Build</td>
<td>Accept</td>
<td>Deploy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERP R1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERP R2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERP R3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERP R4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tech &amp; Infra</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q1 Patching</td>
<td>Non-Prod</td>
<td>Prod</td>
<td>Non-Prod</td>
<td>Prod</td>
<td>Non-Prod</td>
<td>Prod</td>
<td>Non-Prod</td>
<td>Prod</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q2 Patching</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q3 Patching</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q4 Patching</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network refresh</td>
<td>Core Switch</td>
<td>Firewall</td>
<td>Heartbeat NW</td>
<td>Pro</td>
<td>Pro</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### HLRS view by stream & release

- **Stream**: External events, Audits, Business
- **Project**: Payroll Delivery 1, Payroll Delivery 2, ERP R1, ERP R2, ERP R3, ERP R4
- **Tech & Infra**: Q1 Patching, Q2 Patching, Q3 Patching, Q4 Patching, Network refresh

### Environment

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SysInt 1</td>
<td>Vendor Patching</td>
<td>Q1 nonProd</td>
<td>Build for 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Payroll Project</td>
<td>ERP Project</td>
<td>Build for R1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PerfTest 1</td>
<td>Vendor Patching</td>
<td>Q1 nonProd</td>
<td>PSV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Payroll Project</td>
<td>ERP Project</td>
<td>PSV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ERP Project</td>
<td>ERP Project</td>
<td>PSV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UAT 1</td>
<td>Vendor Patching</td>
<td>Q1 nonProd</td>
<td>UAT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Payroll Project</td>
<td>ERP Project</td>
<td>UAT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### HLRS view by Environment & Stream

- **Environment**: SysInt 1, PerfTest 1, UAT 1
- **Release**: Vendor Patching, Payroll Project, ERP Project

---

Enterprise Release Management – an ITSMF Whitepaper
APPENDIX A.3 – EXAMPLE COMPLEX CUTOVER PLAN

This example high level cutover plan for 2 concurrent releases doesn’t replace the detailed, line-by-line implementation plan, but it does help you make sense of busy cutover windows at a glance.